

Table of Contents

1 DBDOC Introduction.....	1
1.1 DBDOC 11.6.....	1
1.2 What is DBDOC?.....	1
1.3 Who Can Benefit From DBDOC?.....	1
1.4 How Does DBDOC Work?.....	1
1.5 System Overview.....	2
1.6 How is DBDOC Used?.....	2
1.6.1 DBDOC for Troubleshooting.....	3
1.6.2 DBDOC as Documentation.....	3
1.6.2.1 Uses of DBDOC for Documentation.....	3
1.6.2.2 Optional Documentation Extensions to DBDOC.....	4
1.6.3 DBDOC During Development.....	4
1.6.4 DBDOC for Commissioning and Checkout.....	4
1.6.4.1 Experiences with DBDOC for Commissioning and Checkout.....	4
1.6.5 Using DBDOC with a Simulator.....	4
1.6.6 DBDOC as a Training Tool.....	5
1.6.7 DBDOC for Error Detection.....	5
1.6.8 DBDOC as a Portable System.....	5
1.6.9 DBDOC for NERC Compliant Live Data on Business LAN.....	6
1.6.10 DBDOC as a Collaboration Tool.....	6
1.6.11 DBDOC for Console Upgrades.....	6
1.6.12 DBDOC for System Audits.....	6
2 Hyperview Browser.....	7
2.1 How to run Hyperview, load a project file, and get help.....	7
2.1.1 How to Run Hyperview.....	7
2.1.1.1 Starting Hyperview from a desktop shortcut.....	7
2.1.1.2 Starting Hyperview from the Start Menu.....	7
2.1.1.3 Launching an additional Hyperview window.....	7
2.1.1.4 Advanced: Running Hyperview from the command line.....	7
2.1.2 How to load a project file.....	9
2.1.2.1 Opening a specific project file.....	9
2.1.2.2 Opening a recent project file.....	9
2.1.2.3 Advanced: Opening the most recently built project file.....	9
2.1.2.4 Advanced: Associating project files with Hyperview in Windows.....	10
2.1.3 Help and Documentation.....	10
2.1.3.1 How to access the built-in help documentation.....	10
2.1.3.2 Help and tutorials on the GMCL website.....	10
2.1.3.3 Finding the version of Hyperlink that built the current project file.....	10
2.1.3.4 Figuring out what version of Hyperview is running.....	11
2.1.3.5 Other ways of getting help.....	11
2.2 Guide to basic navigation in Hyperview.....	12

Table of Contents

2 Hyperview Browser

2.2.1 Panes in Hyperview.....	12
2.2.1.1 Specifying the pane orientation.....	12
2.2.2 Index pane.....	13
2.2.2.1 Displaying the Table of Contents.....	13
2.2.2.2 Switching between Classic and Treeview Table of Contents.....	13
2.2.2.3 Clicking on hotspots in the index pane.....	14
2.2.2.4 Index navigation.....	14
2.2.3 Content pane.....	14
2.2.3.1 Displaying hotspots.....	15
2.2.3.2 Clicking on a hotspot to see all the uses of a value.....	15
2.2.3.3 Double clicking on a hotspot to go to the source of a value.....	15
2.2.4 Focus for the active window.....	15
2.2.5 Setting a "Home Page" for your index and content panes.....	16
2.2.5.1 Setting your "Home Page" to the current document.....	16
2.2.6 Scrolling.....	16
2.2.6.1 Default mode scrolling.....	17
2.2.6.2 Legacy scroll & zoom mode scrolling.....	17
2.2.7 Zooming.....	17
2.2.7.1 Default mode zooming with the mouse.....	17
2.2.7.2 Legacy scroll & zoom mode zooming with the mouse.....	17
2.2.7.3 Zooming with toolbar buttons or the keyboard.....	18
2.2.7.4 Manually setting the magnification factor.....	18
2.2.8 Using browser history to return to previously viewed documents.....	19
2.2.9 Using bookmarks for quick access to pages of interest.....	19
2.2.9.1 System bookmarks.....	20
2.3 Techniques for tracing inputs and outputs throughout your system.....	21
2.3.1 Overview of signal tracing in Hyperview.....	21
2.3.2 Tracing signals using hotspots and the index pane.....	21
2.3.2.1 Tracing back to the source of a value.....	22
2.3.2.2 Finding and displaying the uses of a value.....	22
2.3.3 Tracing signals using Go To Use, Go To Source, and double-clicking on hotspots.....	22
2.3.3.1 Jumping directly to the source of a value with Go To Source or double-click.....	22
2.3.3.2 Jumping directly to a use of a value with "Go To Use" or double-click.....	23
2.4 Full text, database, or title searches and search result navigation, and finding coordinates.....	25
2.4.1 Searching.....	25
2.4.2 Text and database searches.....	25
2.4.2.1 Using Full Text Search to search your system for a word or phrase.....	25
2.4.2.2 Using Database Search to find particular blocks or tags.....	27
2.4.2.3 Setting search options for text and database searches.....	28
2.4.2.4 Viewing search results for text and database searches.....	29
2.4.2.5 Displaying and stepping through search results using the Hyperview toolbar.....	29

Table of Contents

2 Hyperview Browser

2.4.3 Using the Topic Title Search to jump directly to a topic.....	30
2.4.4 Advanced: Finding a Coordinate.....	32
2.5 Guide to Live Data on Documents.....	34
2.5.1 Live Data Monitoring.....	34
2.5.1.1 Using Slow Mode live data.....	34
2.5.1.2 Getting live data in Continuous mode.....	34
2.5.1.3 Managing the Slow Mode Data Load.....	35
2.5.2 Live Loop Annotations.....	35
2.5.3 Creating a Live Loop Annotation.....	38
2.5.3.1 Block Options (for either Live Loop Annotations or Watch Window Blocks).....	40
2.5.3.1.1 Basic Options.....	41
2.5.3.1.2 Annotation Options.....	41
2.5.3.1.3 Watch Window Options.....	41
2.5.3.1.4 Plot or Dynamic Bar Range.....	41
2.5.3.2 Reading Live Loop Annotations.....	41
2.5.3.3 Managing Live Loop Annotations.....	42
2.5.3.4 Moving Live Loop Annotations.....	43
2.5.3.5 Deleting Live Loop Annotations.....	44
2.5.4 Live Specs.....	44
2.6 Trending Live Data in the Watch Window.....	46
2.6.1 Watch Window Overview.....	46
2.6.1.1 Using the Watch Window: quick summary.....	46
2.6.2 Opening the Watch Window.....	47
2.6.3 Adding blocks to the Watch Window so they can be plotted.....	47
2.6.4 Defining and editing block plots in the Watch Window.....	47
2.6.4.1 Specific Watch Window Options.....	49
2.6.5 Creating groups of blocks in the Watch Window.....	49
2.6.5.1 Importing a list of blocks from CIULink.....	49
2.6.5.2 Managing groups of blocks.....	50
2.6.6 Watch Window Highlights.....	50
2.6.7 Timestamp Navigation.....	51
2.6.8 Data Marker Navigation.....	51
2.6.9 Managing Watch Window related bandwidth, and disk usage.....	52
2.6.9.1 Bandwidth.....	52
2.6.9.2 Memory.....	53
2.6.9.3 Disk Usage.....	53
2.6.10 Importing data into other applications.....	53
2.7 Viewing and Managing System Errors with the Error Browser.....	54
2.7.1 Error Browser Overview.....	54
2.7.2 Opening the Error Browser.....	54
2.7.3 Marking errors as Starred, Reviewed or Hidden.....	55

Table of Contents

2 Hyperview Browser

2.7.3.1 Marking individual errors.....	55
2.7.3.2 Marking selected errors.....	55
2.7.4 Filtering errors.....	56
2.7.5 Grouping errors.....	56
2.7.6 New errors.....	57
2.7.7 Sharing error information with other users.....	57
2.8 Customizing the Browser Display.....	59
2.8.1 Showing and hiding specs and attributes.....	59
2.8.1.1 Viewing specs on CAD/CLD diagrams.....	59
2.8.1.2 Viewing attributes on CAD/CLD diagrams.....	61
2.8.1.3 Showing live specs on CAD/CLD diagrams.....	62
2.8.2 Highlighting options in Hyperview.....	62
2.8.2.1 Highlighting search hits.....	62
2.8.2.2 Highlighting hotspots.....	63
2.8.2.3 Showing "Vegas Lights".....	63
2.8.3 Options Menu: Properties for Hyperview.....	63
2.8.3.1 View options.....	63
2.8.3.2 Color options.....	65
2.9 Making Notes on Documents.....	67
2.9.1 Overview of Annotations.....	67
2.9.2 Text Annotations.....	67
2.9.3 Managing Annotations.....	67
2.10 Printing Documents and Groups of Documents.....	71
2.10.1 Printing a single document.....	71
2.10.1.1 Printing options.....	71
2.10.1.2 Fast Print to use your previously defined print settings.....	73
2.10.1.3 Printing selected text.....	73
2.10.2 Marking the current topic for later printing.....	73
2.10.2.1 Marking the current topic for printing.....	73
2.10.2.2 Marking a linked topic for printing.....	74
2.10.3 Using the Mark for Print dialog to print groups of topics.....	74
2.10.3.1 Marking a group of topics for printing.....	74
2.10.3.2 Removing topics or groups of topics from the marked topics list.....	75
2.10.3.3 Other options.....	75
2.11 Sharing Data with Other Hyperview Users.....	76
2.11.1 Overview of User Data Sharing.....	76
2.11.2 Specifying the Shared Data Folder.....	76
2.11.3 Updating your user information.....	76
2.11.4 Specifying what sorts of user data to share with others.....	77
2.11.4.1 Specifying whose data to include by default.....	77
2.12 Tools, Reports and Indexes.....	78

Table of Contents

2 Hyperview Browser

2.12.1 Viewing the resources used in a drawing with the Audit Window.....	78
2.12.2 Using PCU Map to understand interactions among PCUs.....	79
2.12.3 Graphical Block Map: a graphical overview of block usage.....	80
2.12.4 Tag Index: an alphabetical index of all the tags in your system.....	82
2.12.4.1 Using the Tag Index.....	83
2.12.5 Graphics Cross-Reference Window: index of all the graphics that link to and from the graphic, plus included symbols.....	84
2.12.6 Function Code Descriptions and Uses: an index of all the function codes in your system.....	85
2.12.7 Function Block Index: a list of all the function blocks used on a CAD/CLD.....	86
2.12.8 Graph of F(x) (Function Code 1).....	86
2.12.9 Adapt Block.....	87
2.12.10 Rung Block Display.....	88
2.12.11 AutoCAD and MicroStation.....	89
2.13 Managing Live Loop Connections.....	90
2.13.1 About live data settings.....	90
2.13.2 About the Live Loop Setup dialog.....	90
2.13.3 Choosing a settings list.....	91
2.13.4 Understanding your current live loop settings.....	92
2.13.5 Adding or modifying live loop connections.....	93
2.13.5.1 Adding a connection.....	93
2.13.5.2 Editing a connection.....	94
2.13.5.3 Deleting a connection.....	94
2.13.5.4 Ignoring a connection.....	94
2.13.6 Specifying a non-default local port range.....	94
2.13.7 Adding routing preferences.....	95
2.13.8 Unavailable data.....	96
2.13.9 Status of live data connections.....	97
2.13.10 Compression of Hyperview data requests.....	97
2.14 Managing Live Data Load.....	98
2.14.1 Overview of live data bandwidth limitations when fetching live system data.....	98
2.14.1.1 Constraints on live data fetch rate.....	98
2.14.2 Managing live data load.....	99
2.14.2.1 Managing data load for live data on documents.....	99
2.14.2.2 Managing live data load for blocks in the Watch Window.....	100
2.14.3 How to tell how much data is being requested (Live Data Status Bar).....	100
2.15 Hyperview Keyboard Shortcuts.....	102

3 Guide to Hyperview in a Browser.....104

3.1 Hyperview in a Browser.....	104
3.1.1 Overview.....	104

Table of Contents

3 Guide to Hyperview in a Browser

3.1.2 Platform/Browser compatibility.....	104
3.1.3 Setting up a Hyperview Service.....	105
3.1.4 Managing user accounts.....	105
3.2 Setting up a Hyperview Service.....	105
3.2.1 Overview.....	105
3.2.2 Running the Hyperview Service Controller.....	106
3.2.3 Configuring your Hyperview Services setup.....	108
3.2.3.1 General setup for all Hyperview Services.....	108
3.2.3.1.1 Hyperview server setup.....	109
3.2.3.1.2 HTTPS Settings.....	109
3.2.3.1.3 Other setup information.....	110
3.2.4 Adding or modifying a Hyperview Service.....	110
3.2.5 Starting and stopping Hyperview Services from the command line.....	112
3.2.5.1 Command Line.....	112
3.2.5.2 Example.....	112
3.2.6 Accessing Hyperview Services through a firewall.....	113
3.2.7 Managing Hyperview Service log files.....	113
3.2.7.1 Log file location.....	113
3.2.7.2 Log file names.....	113
3.2.7.3 Logging options.....	114
3.2.7.3.1 Hyperview Service Controller logs.....	114
3.2.7.3.2 Hyperview Service logs.....	114
3.2.8 Finding Hyperview Service user data files.....	115
3.3 Using Hyperview in a Browser.....	116
3.3.1 Hyperview website location.....	116
3.3.2 Hyperview in a Browser functionality.....	116
3.3.3 Basic functionality.....	117
3.3.3.1 Displaying the main Table of contents.....	117
3.3.3.2 Synchronizing the index pane to the pane with focus.....	117
3.3.3.3 Opening the menu.....	117
3.3.3.4 Creating and managing bookmarks.....	118
3.3.3.5 Getting online help.....	119
3.3.4 Navigation and signal tracing.....	119
3.3.4.1 Browsing to the previous topic in a topic group.....	119
3.3.4.2 Browsing to the next topic in a topic group.....	119
3.3.4.3 Browsing to the previous entry in an index.....	120
3.3.4.4 Browsing to the next entry in an index.....	120
3.3.5 Zooming and scrolling.....	120
3.3.5.1 Zooming on a graphical topic.....	120
3.3.5.2 Scrolling and panning.....	121
3.3.6 Searching.....	121

Table of Contents

3 Guide to Hyperview in a Browser

3.3.6.1 Searching for a word or phrase.....	121
3.3.6.1.1 Overview of Full Text Search.....	121
3.3.6.1.2 Search options.....	122
3.3.6.1.3 Search results.....	122
3.3.6.2 Searching topic titles.....	122
3.3.6.3 Clearing search highlights.....	123
3.3.7 Showing live data.....	124
3.3.7.1 Overview.....	124
3.3.7.2 Live loop settings.....	124
3.3.7.3 Updating live data settings.....	124
3.3.7.4 Displaying live data on a document (updated periodically).....	125
3.3.7.4.1 Behavior.....	125
3.3.7.5 Displaying live data on a document (updated when needed).....	125
3.3.7.5.1 Behavior.....	125
3.3.8 Customizing the display.....	126
3.3.8.1 Showing or hiding function code specs.....	127
3.3.8.1.1 Spec Display Modes.....	127
3.3.8.2 Showing or hiding tag names and attributes.....	128
3.3.8.2.1 Attribute Display Modes.....	129
3.3.9 Finding version information.....	130
3.3.9.1 Showing DBDOC file build information.....	130
3.3.9.2 Showing Hyperview Server information.....	130
3.3.10 User accounts.....	131
3.3.10.1 Logging in to your Hyperview in a Browser account.....	131
3.3.10.2 Changing your password.....	132
3.3.10.3 Signing out of your Hyperview in a Browser account.....	132
3.4 Managing user accounts.....	132
3.4.1 User accounts.....	132
3.4.1.1 User actions.....	133
3.4.1.2 Administrator actions.....	133
3.4.2 Creating, modifying, and deleting user accounts.....	133
3.4.3 Resetting the Administrator password.....	134

4 Overview of BuildPlus.....135

4.1 Getting Started with BuildPlus.....	135
4.1.1 DBDOC dongle.....	135
4.1.2 About your DBDOC license and computer key.....	136
4.2 Adding or changing license information.....	136
4.2.1 Opening BuildPlus.....	136
4.2.2 About BuildRoots.....	137
4.2.3 Getting Help with BuildPlus.....	139

Table of Contents

4 Overview of BuildPlus

4.2.4 About your DBDOC version.....	140
4.3 Doing Your First Build.....	141
4.3.1 Naming Your Project and Choosing Database and 800xA Files to Build In.....	141
4.3.2 Specifying Configuration, Graphics and Other Files with Configuration Wizard.....	144
4.3.3 Building in Settings for Live Data.....	148
4.3.4 Resolving Other BuildPlus Dialogs.....	151
4.3.4.1 Duplicate File Choices.....	151
4.3.4.2 Conductor NT or PPB Console associations.....	152
4.3.4.3 Symbols-Graphics Association.....	155
4.3.4.4 Database-Graphics Association.....	157
4.3.4.5 Database Selections - Changing included databases.....	159
4.4 After Your First Build.....	160
4.4.1 About Your DBDOC File.....	160
4.4.1.1 Changing how the Snapshot file will be named (BuildRoot option).....	160
4.4.1.2 Choosing locations for copying the Snapshot after the build.....	161
4.4.1.3 Renaming a Snapshot from BuildPlus.....	161
4.4.1.4 Building bookmarks and annotations into your project file.....	162
4.4.1.4.1 "User" vs "System" Hyperview.....	162
4.4.1.4.2 Types of "System" data.....	163
4.4.2 Launching Windows Explorer to Your Project Files.....	164
4.4.3 Help with Error Files.....	164
4.4.4 Sending an Error.7z File to GMCL.....	164
4.4.5 Exiting BuildPlus.....	165
4.5 Rebuilding.....	165
4.5.1 When your files have been modified.....	165
4.5.2 Changing Your Data File Sources.....	168
4.5.3 After a DBDOC Update.....	169
4.5.4 Changing Options.....	169
4.5.5 Scheduled Builds.....	169
4.5.6 Command line builds.....	169
4.5.7 Comparing Build Results with Winmerge.....	169
4.6 More About BuildPlus.....	170
4.6.1 Opening a recent project.....	170
4.6.2 Renaming a project.....	170
4.6.3 Moving a project.....	171
4.6.4 Cloning a project.....	172
4.6.5 Deleting a project.....	172
4.6.6 Exporting a project.....	172
4.6.7 Importing a project.....	173
4.6.8 Reverting to a previous build.....	173
4.6.9 Cleaning up temporary/intermediate files.....	174

Table of Contents

4 Overview of BuildPlus

4.6.10 Sibling Projects.....	175
4.6.11 BuildPlus User Interface.....	175

5 Basics of CIUMon.....178

5.1 Overview.....	178
5.1.1 Technical requirements.....	179
5.2 Methods of connecting CIUMon to a CIU.....	179
5.3 Connecting to a CIU.....	180
5.3.1 Choosing a port to get data from CIUMon.....	181
5.3.2 Configuring CIUMon for your CIU type.....	182
5.3.3 Connecting to a TRAX emulator.....	182
5.3.4 Making sure you can get data from CIUMon.....	183
5.3.5 Running CIUMon manually.....	183
5.4 Connecting to RoviSys Turbo.....	184
5.4.1 Installing RoviSys Turbo.....	185
5.4.2 Connecting RoviSys Turbo to a CIU.....	186
5.4.3 Methods of connecting CIUMon to RoviSys Turbo.....	187
5.4.4 Connecting to RoviSys Turbo with virtual ports.....	188
5.4.5 Connecting to RoviSys Turbo with serial ports.....	190
5.4.6 Adding serial ports to your computer.....	193
5.4.7 Connecting other programs to RoviSys Turbo.....	196
5.5 Connecting to Harmony OPC Server.....	197
5.6 Managing CIUMon configurations.....	199
5.6.1 Loading, saving, and deleting configurations.....	200
5.6.2 Keeping your configurations organized.....	201
5.7 Getting live data across a network.....	202
5.7.1 Getting live data through a firewall.....	203
5.7.2 Getting live data across an intermediate LAN with relay mode.....	204
5.7.3 Getting live data directly from a business LAN.....	204
5.7.4 Running CIUMon in relay mode.....	204
5.7.5 Running CIUMon in collector mode.....	205
5.7.6 Is it safe to let everyone on the network access live data?.....	206
5.8 Running CIUMon automatically (as a service).....	206
5.8.1 Setting up a CIUMon service.....	206
5.8.2 Changing a CIUMon service.....	207
5.8.3 Error recovery when running as a service.....	209
5.8.4 How to tell if CIUMon is running as a service.....	209
5.8.5 Running RoviSys Turbo automatically.....	209
5.8.6 Starting and stopping a CIUMon service with CIUWatcher.....	210
5.8.7 Removing a CIUMon service.....	211
5.9 Improving CIUMon's performance.....	211

Table of Contents

5 Basics of CIUMon

5.9.1	Limitations on how much live data you can get.....	211
5.9.2	Best practices for using DBDOC live data.....	211
5.9.3	Getting more live data.....	212
5.9.4	Reducing the impact of CIUMon on computer performance.....	213
5.10	Managing access to live data.....	214
5.10.1	Restricting access to live data.....	214
5.10.2	Tracking who is using live data.....	217
5.10.3	Allowing some users to get more live data than others.....	217

6 Build Reports and Analysis.....220

6.1	Overview of the DBDOC build reports.....	220
6.2	Build reports.....	220
6.3	General build reports.....	221
6.4	Other build report files.....	222
6.4.1	FCnnn.DBF - Extracted specifications from CAD and CLD files.....	223
6.4.2	SPEC.MDB - Used to link the files in an Access database.....	224
6.4.3	F(X) Specs.txt - Diagnostic data file FC 1 spec messages.....	225
6.4.4	FCList.txt - List of the number of instances of each function code.....	225
6.4.5	TagUsage.csv - Summary of tags used in the system.....	226
6.4.6	Emptyfiles.txt - Shows source files that were empty.....	227
6.4.7	Blank_CAD.txt - Shows CAD files and CLD sheets that were empty.....	227
6.4.8	New_tags.txt - A list of all new tags that have been added since the last build.....	227
6.4.9	Tags_in_graphics.dbf - A database with a record for each tag in each graphic.....	228
6.4.10	Unused Tag Lists - List of tags that are not in any graphic.....	228
6.4.10.1	Unused-unconfigured_tags.txt - Not on any graphic but not on any CLD or CAD sheet.....	229
6.4.10.2	Unused-Alarm_tags.txt - Not on any graphic but should alarm.....	229
6.4.10.3	Unused_Tags.txt - All tags that are not on any graphic and not in the specialized lists.....	230
6.4.11	Extracted and synthesized "flattened" databases.....	231
6.4.12	Global taglists - A master tag list with all the tags in the system.....	231
6.4.13	Unused_symbols.txt - This file lists symbols that are unused by the graphics.....	232
6.4.14	Significant Change Report - This report lists significant change of exception reported values in units and percent change.....	232
6.4.15	Undefined tags - lists tags found on graphics but not defined in a database.....	233
6.4.16	Undefined graphics - lists any hotspots that are supposed to link to other documents, but don't.....	233
6.4.17	Undefined symbols - lists missing symbols or sub-models.....	233
6.5	Management Of Change (MOC).....	234
6.5.1	Understanding Management of Change.....	234
6.5.2	Using Management of Change with Scheduled Builds.....	235

Table of Contents

6 Build Reports and Analysis

6.6 Build reports.....	235
6.6.1 Error summary reports.....	236
6.6.1.1 DBDOC.ERR - A log of the entire build.....	236
6.6.1.2 DBDOC_RESIDUAL.ERR - Any messages that are not included in our documented error messages.....	236
6.6.1.3 DBDOC_SUMMARY.ERR - Composed of every file that had errors.....	237
6.6.2 Individual errors.....	237
6.6.2.1 800xA color not found [INTERNAL 201].....	237
6.6.2.2 Adapt block adapts previous block [CHECK 118].....	238
6.6.2.3 Adapt block does not exist [ERROR 014].....	238
6.6.2.4 Adapt block: Adapted block does not exist [ERROR 021].....	239
6.6.2.5 Adapt block: Not used [ERROR 019].....	240
6.6.2.6 Adapt block: Spec out of range [ERROR 020].....	240
6.6.2.7 Adapt type mismatch [ERROR 169].....	244
6.6.2.8 Adapted block on other page [CHECK 044].....	249
6.6.2.9 Adapted coordinates out of order [INFO 032].....	249
6.6.2.10 Adapted rung spec [INFO 061].....	250
6.6.2.11 AlarmVector tag has case mismatch - tag is in table [ERROR 138].....	250
6.6.2.12 AlarmVector tag has case mismatch - tag is not in table [ERROR 139].....	251
6.6.2.13 AlarmVector tag is not defined in any database associated with console [ERROR 140].....	251
6.6.2.14 Ambiguous wiring [ERROR 121].....	252
6.6.2.15 APID block illegal decrease flag [ERROR 086].....	253
6.6.2.16 APID block illegal increase flag [ERROR 085].....	254
6.6.2.17 APID old algorithm and inhibit inputs [ERROR 087].....	256
6.6.2.18 Assertion failed [INTERNAL 080].....	257
6.6.2.19 Attempted ADAPT block redefinition [ERROR 176].....	258
6.6.2.20 Bad CAD magic number [INTERNAL 002].....	258
6.6.2.21 Block adapts a block that is already adapted [ERROR 117].....	259
6.6.2.22 Block not wired from same function code [CHECK 186].....	260
6.6.2.23 Block number on unwired input [COSMETIC 098].....	261
6.6.2.24 Block number too high [ERROR 322].....	261
6.6.2.25 Block out of range [ERROR 167].....	262
6.6.2.26 Blocks have been overlaid [CHECK 225].....	262
6.6.2.27 Bogus source for block zero [BUILD 180].....	263
6.6.2.28 Both 0 and 1 have same descriptor [CHECK 064].....	264
6.6.2.29 Broken connection [ERROR 345].....	265
6.6.2.30 Building multiple 800xA consoles into one project file not yet supported [INTERNAL 200].....	265
6.6.2.31 CADNames cannot be opened [ERROR 10044].....	266
6.6.2.32 Can't find ProgID for GUID-named dependency [INTERNAL 203].....	266

Table of Contents

6 Build Reports and Analysis

6.6.2.33 Can't open DBF file RESOLVE.DBF [BUILD 10001].....	266
6.6.2.34 Cannot determine target [CHECK 347].....	267
6.6.2.35 Cannot import error database [INTERNAL 289].....	267
6.6.2.36 Choosing description for XRefObjID [CHECK 10029].....	267
6.6.2.37 Circular inhibit [CHECK 181].....	268
6.6.2.38 Circular reference [CHECK 184].....	269
6.6.2.39 Circular symbol inclusion [CHECK 182].....	269
6.6.2.40 Circular symbol inclusion of tag [CHECK 183].....	269
6.6.2.41 CLD file has not been recompiled. [INFO 10022].....	270
6.6.2.42 Completely unconnected wire [ERROR 340].....	270
6.6.2.43 Connection to unconfigured variable [ERROR 342].....	270
6.6.2.44 Connection wired in Composer only [INTERNAL 170].....	271
6.6.2.45 Connection wired in drawing only [INTERNAL 171].....	271
6.6.2.46 Continuous exception reports generated [ERROR 145].....	272
6.6.2.47 Control character in tag description [INFO 323].....	273
6.6.2.48 Could not locate batch include file [BUILD 252].....	273
6.6.2.49 Couldn't open font file [BUILD 192].....	273
6.6.2.50 DAANG alarm control not an integer [ERROR 090].....	274
6.6.2.51 DAANG both de-alarm and return alarm [ERROR 089].....	274
6.6.2.52 DAANG illegal alarm control [ERROR 088].....	275
6.6.2.53 DAANG suspicious alarm period [ERROR 091].....	276
6.6.2.54 Data type mismatch [ERROR 094].....	277
6.6.2.55 Database has no records [ERROR 290].....	278
6.6.2.56 Descriptor empty [CHECK 015].....	278
6.6.2.57 Digital in Alarm Group cannot alarm [CHECK 078].....	279
6.6.2.58 DIOFCSpec associated with unknown FC [INFO 10014].....	279
6.6.2.59 Discarding description for XRefObjID [CHECK 10030].....	280
6.6.2.60 Display not found [CHECK 197].....	280
6.6.2.61 Display with GUID not found [CHECK 198].....	281
6.6.2.62 Displayed tag has no source - module not built [ERROR 266].....	281
6.6.2.63 Document defined in another chapter [INFO 188].....	282
6.6.2.64 doVector called with non-string second argument [INTERNAL 244].....	283
6.6.2.65 doVector() called with numeric 2nd argument [INTERNAL 243].....	284
6.6.2.66 Duplicate global OREF [CHECK 190].....	284
6.6.2.67 Duplicate local OREF [CHECK 296].....	285
6.6.2.68 Duplicate source [ERROR 092].....	285
6.6.2.69 Duplicate station tag [INFO 189].....	286
6.6.2.70 Duplicate tag in database [ERROR 103].....	286
6.6.2.71 Dynamic definition undefined [INTERNAL 277].....	287
6.6.2.72 EbVector references unknown graphic [CHECK 216].....	287
6.6.2.73 Element outside configured bounds [INTERNAL 202].....	287

Table of Contents

6 Build Reports and Analysis

6.6.2.74 Enhanced trend has illegal S2 value [ERROR 024].....	288
6.6.2.75 Enhanced trend: S1 wired from wrong function code [ERROR 025].....	289
6.6.2.76 Error count exceeded [BUILD 60000].....	290
6.6.2.77 External output reference is unreachable [ERROR 10015].....	291
6.6.2.78 External output reference not found [CHECK 10008].....	293
6.6.2.79 F(x) discontinuity (step) [CHECK 031].....	293
6.6.2.80 F(x) graph is flat [INFO 035].....	294
6.6.2.81 F(x) graph is flat (unwired) [COSMETIC 036].....	295
6.6.2.82 F(x) uses large values and may not clamp output [ERROR 034].....	296
6.6.2.83 Failed to locate field indices -- not a database? [BUILD 220].....	297
6.6.2.84 Failed to locate field name [BUILD 221].....	297
6.6.2.85 Failed to open file [BUILD 006].....	298
6.6.2.86 Failed to read header [BUILD 001].....	298
6.6.2.87 Failed to read library directory [INTERNAL 013].....	298
6.6.2.88 Failed to read record [INTERNAL 004].....	299
6.6.2.89 Failed to reseek to start [INTERNAL 005].....	299
6.6.2.90 FC126 S3 Exceeds 4 Linked Blocks [ERROR 318].....	299
6.6.2.91 FC225 S5 must be 1 for FC228 [ERROR 311].....	300
6.6.2.92 Fewer fields than substitutions [BUILD 222].....	300
6.6.2.93 File contains a space in its name [COSMETIC 213].....	301
6.6.2.94 File version unknown [INTERNAL 271].....	301
6.6.2.95 Font file not found [BUILD 191].....	302
6.6.2.96 Frozen import block [ERROR 093].....	302
6.6.2.97 Function block chain loop [ERROR 325].....	304
6.6.2.98 Function block disabled [ERROR 297].....	304
6.6.2.99 Gain spec is adapted [INFO 119].....	305
6.6.2.100 Got lost parsing file [INTERNAL 10007].....	306
6.6.2.101 Graphic link target is null [INTERNAL 291].....	306
6.6.2.102 Harmony I/O block illegally wired [ERROR 095].....	307
6.6.2.103 History tag references tag not in database [ERROR 253].....	307
6.6.2.104 Ignoring adapted Block 0 [ERROR 082].....	309
6.6.2.105 Illegal adapt block, spec 3 is 0 [ERROR 116].....	309
6.6.2.106 Illegal import (not exception-reported) [ERROR 129].....	310
6.6.2.107 Illegal import (type mismatch) [ERROR 131].....	311
6.6.2.108 Illegal import (different loop) [ERROR 130].....	312
6.6.2.109 Illegal import (full status ignored) [CHECK 132].....	312
6.6.2.110 Illegal module value [INTERNAL 066].....	313
6.6.2.111 Illegal spec value: should be 0 or 1 [ERROR 149].....	313
6.6.2.112 Illegal spec value: Spec S7 from 0 - 100 [ERROR 146].....	314
6.6.2.113 Import from same PCU [ERROR 128].....	315
6.6.2.114 Import information [ERROR 255].....	316

Table of Contents

6 Build Reports and Analysis

6.6.2.115 Import Module Status is wired but non-functional [CHECK 174].....	317
6.6.2.116 Import out of range [ERROR 336].....	318
6.6.2.117 Imported block has no source - module not built [ERROR 265].....	319
6.6.2.118 Included image was not found [CHECK 294].....	320
6.6.2.119 Inconsistent LPM data [ERROR 10020].....	320
6.6.2.120 Increasing drawing height [COSMETIC 023].....	321
6.6.2.121 Increasing drawing width [COSMETIC 022].....	322
6.6.2.122 Inhibit tag not found [ERROR 348].....	322
6.6.2.123 Input list attempts to import nonexistent block 9999 [ERROR 334].....	323
6.6.2.124 Input list attempts to import out-of-range block 9999 [ERROR 335].....	323
6.6.2.125 Input list imports block 9999 [CHECK 333].....	323
6.6.2.126 Input wiring expected - Harmony [CHECK 187].....	324
6.6.2.127 Inputs have been overlaid [CHECK 226].....	324
6.6.2.128 Integrator disabled by spec value [ERROR 298].....	325
6.6.2.129 Internal error: get_block() non-block tag [INTERNAL 049].....	325
6.6.2.130 Internal error: Unknown hotspot [INTERNAL 016].....	326
6.6.2.131 Invalid block number on line [INTERNAL 246].....	326
6.6.2.132 Invalid CAD file: Too short [INTERNAL 010].....	326
6.6.2.133 Invalid CAD file: Zero length [INTERNAL 009].....	327
6.6.2.134 Invalid header or bad magic number [INTERNAL 10011].....	327
6.6.2.135 Invalid loop number for block [INTERNAL 247].....	327
6.6.2.136 Invalid loop: high byte used [INTERNAL 248].....	328
6.6.2.137 Invalid record length [INTERNAL 003].....	328
6.6.2.138 IREF does not have an input [ERROR 096].....	328
6.6.2.139 IREF GlobalRef import mismatch [INFO 10019].....	330
6.6.2.140 IRef not found in IRefRes table: set to blank [INTERNAL 10013].....	331
6.6.2.141 IRef not found in IRefRes table: skipping [INTERNAL 10012].....	332
6.6.2.142 Legacy err message [INTERNAL 10005].....	332
6.6.2.143 Legacy fatal message [INTERNAL 10006].....	333
6.6.2.144 Legacy warning message [INTERNAL 10004].....	333
6.6.2.145 Library missing [COSMETIC 026].....	333
6.6.2.146 Linked Block Already Linked [ERROR 317].....	334
6.6.2.147 Linked Block Connected To Self [ERROR 319].....	334
6.6.2.148 Linked Block Has Wrong FC [ERROR 316].....	334
6.6.2.149 Live wire not connected [CHECK 199].....	335
6.6.2.150 Loop out of range [ERROR 164].....	336
6.6.2.151 M1 display file not found (not configured) [CHECK 195].....	336
6.6.2.152 Macro is present in multiple libraries [INFO 10031].....	337
6.6.2.153 Mismatched tagname [CHECK 205].....	337
6.6.2.154 Mismatched tagname (not in database) [CHECK 204].....	337
6.6.2.155 Missing associated file [INTERNAL 285].....	338

Table of Contents

6 Build Reports and Analysis

6.6.2.156 Missing definitions file [INTERNAL 292].....	338
6.6.2.157 Missing dynamic target [INTERNAL 275].....	338
6.6.2.158 Missing graphic target [INFO 314].....	339
6.6.2.159 Missing LBR Report [INTERNAL 000].....	339
6.6.2.160 Missing M1 not defined [CHECK 267].....	339
6.6.2.161 Missing string introducer [INFO 10040].....	340
6.6.2.162 Missing symbol or sub-model [BUILD 099].....	340
6.6.2.163 Module mismatch: does not match module in file [ERROR 10021].....	342
6.6.2.164 Module mismatch: does not match Module in file -- skipping [ERROR 10038].....	343
6.6.2.165 Module out of range [ERROR 166].....	343
6.6.2.166 More than one input to connector [ERROR 106].....	344
6.6.2.167 Multilevel alarms disabled [ERROR 062].....	346
6.6.2.168 Multiple slave FCs [ERROR 153].....	347
6.6.2.169 Multiple slave FCs: Too many FC 132's [ERROR 155].....	348
6.6.2.170 Multiple slave FCs: Too many FC 149's [ERROR 154].....	348
6.6.2.171 Multiple targets [ERROR 328].....	349
6.6.2.172 Multiple TSTALM [ERROR 084].....	349
6.6.2.173 Multiple wires at source: assuming connector [COSMETIC 219].....	350
6.6.2.174 Negative block number [CHECK 037].....	351
6.6.2.175 Negative string length [INFO 10041].....	352
6.6.2.176 Nested inhibit [CHECK 069].....	352
6.6.2.177 Never found library file [COSMETIC 011].....	353
6.6.2.178 No SHAPE.LBR was found [BUILD 012].....	353
6.6.2.179 No source: Imported block not found [ERROR 110].....	353
6.6.2.180 No source: Other [ERROR 113].....	354
6.6.2.181 No source: Used in display [ERROR 111].....	355
6.6.2.182 No source: Verify [COSMETIC 112].....	355
6.6.2.183 No tag database specified for console [BUILD 108].....	356
6.6.2.184 No valid sections found in XDC file [BUILD 109].....	356
6.6.2.185 No wired input [COSMETIC 097].....	357
6.6.2.186 Node out of range [ERROR 165].....	358
6.6.2.187 Non-local exit called -- CLD resolution [INTERNAL 10043].....	358
6.6.2.188 Non-local exit called at offset [INTERNAL 10042].....	359
6.6.2.189 Non-local exit from file [INTERNAL 060].....	359
6.6.2.190 Non-orthogonal rotation [INTERNAL 038].....	359
6.6.2.191 Non-orthogonal shear rotation [ERROR 327].....	360
6.6.2.192 Not enough inputs [ERROR 133].....	360
6.6.2.193 Only handling first tag [CHECK 048].....	361
6.6.2.194 OREFs differ in case only [CHECK 185].....	362
6.6.2.195 Out of expected range [ERROR 10026].....	362
6.6.2.196 Outdated file format [INTERNAL 206].....	363

Table of Contents

6 Build Reports and Analysis

6.6.2.197 Override on input [INFO 122].....	363
6.6.2.198 Override on output: forced to a constant [INFO 151].....	364
6.6.2.199 Override on output: held [INFO 150].....	366
6.6.2.200 Override on output: invalid code [INFO 152].....	366
6.6.2.201 Parse error in file [INTERNAL 115].....	367
6.6.2.202 PDF security prevents embedding [BUILD 324].....	367
6.6.2.203 PID block uses derivative with PV from discontinuous input [ERROR 307].....	368
6.6.2.204 PID Controller has all active tuning constants zero [ERROR 173].....	368
6.6.2.205 PID Controller has primary gain spec zero [ERROR 172].....	369
6.6.2.206 PID lag spec zero [CHECK 177].....	370
6.6.2.207 Point has no description [COSMETIC 251].....	371
6.6.2.208 Possibly corrupt GPI mdb File [ERROR 315].....	372
6.6.2.209 Possibly corrupt HGS mdb File [CHECK 320].....	372
6.6.2.210 Possibly corrupt HPG mdb File [CHECK 321].....	372
6.6.2.211 Problem communicating with dongle [INTERNAL 045].....	373
6.6.2.212 QOR forced to 1 [INFO 135].....	373
6.6.2.213 QOR used as a NOR [INFO 134].....	373
6.6.2.214 Record failed to find matching format [INTERNAL 210].....	374
6.6.2.215 REDAI both inputs same block [CHECK 072].....	375
6.6.2.216 REDAI deadband exceeds rate limit [CHECK 075].....	375
6.6.2.217 REDAI invalid deviation S4 [CHECK 074].....	376
6.6.2.218 REDAI quality is never tested [CHECK 175].....	377
6.6.2.219 REDAI select input hardwired [CHECK 073].....	378
6.6.2.220 REDAI select input wired but ignored [CHECK 079].....	378
6.6.2.221 REDDI both inputs from same block [CHECK 076].....	379
6.6.2.222 REDDI select input hardwired [CHECK 077].....	379
6.6.2.223 Referenced M1 display file not found [CHECK 194].....	380
6.6.2.224 Regex compilation error [INTERNAL 039].....	381
6.6.2.225 Regex execution error [INTERNAL 040].....	381
6.6.2.226 REMSET limits out of order [ERROR 261].....	381
6.6.2.227 Requested color index out of range [INTERNAL 193].....	382
6.6.2.228 RESTR block order wrong [ERROR 065].....	382
6.6.2.229 RESTR block will never save [ERROR 309].....	383
6.6.2.230 RESTR block will update to NVRAM every cycle [INFO 308].....	384
6.6.2.231 Rung block did not use all inputs to compute output [ERROR 127].....	384
6.6.2.232 Rung block spec is unwired but used [ERROR 125].....	385
6.6.2.233 Rung block spec is wired but unused [ERROR 124].....	386
6.6.2.234 Rung block spec is wired from constant [ERROR 126].....	387
6.6.2.235 Rung stack not initialized [ERROR 306].....	388
6.6.2.236 Same-bus input cannot read value in other loop [ERROR 10016].....	388
6.6.2.237 Same-bus input cannot read value in other PCU [ERROR 10017].....	390

Table of Contents

6 Build Reports and Analysis

6.6.2.238 Same-loop input attempts to read value in other loop [ERROR 10018].....	392
6.6.2.239 Segment control block auto-sequencing enabled [INFO 302].....	394
6.6.2.240 Segment control block bad priority [ERROR 300].....	395
6.6.2.241 Segment control block check cycle time [ERROR 303].....	395
6.6.2.242 Segment control block checkpoint period is too small [ERROR 304].....	396
6.6.2.243 Segment control block duplicate priority [ERROR 301].....	396
6.6.2.244 Skipping file: Error parsing submodel [INTERNAL 055].....	397
6.6.2.245 Skipping file: File contains no records [INTERNAL 059].....	397
6.6.2.246 Skipping file: File too short [INTERNAL 058].....	397
6.6.2.247 Skipping file: OpenDesign Exception [INTERNAL 052].....	398
6.6.2.248 Skipping file: Possibly corrupt A/B PLC file [INTERNAL 057].....	398
6.6.2.249 Skipping file: Unable to read file [INTERNAL 054].....	398
6.6.2.250 Skipping file: Unknown M1/M2 file type [INTERNAL 056].....	399
6.6.2.251 SODG aliasing error [ERROR 160].....	399
6.6.2.252 Source in same module [ERROR 157].....	399
6.6.2.253 Source wired to another source [ERROR 158].....	400
6.6.2.254 Span is zero for dynamic bar [COSMETIC 231].....	401
6.6.2.255 Spare block used in downstream wiring [CHECK 168].....	401
6.6.2.256 Spec input index out of range [ERROR 018].....	402
6.6.2.257 Spec is illegally adapted [ERROR 083].....	402
6.6.2.258 Specification out of valid range [ERROR 299].....	403
6.6.2.259 SQLite bind error [INTERNAL 208].....	405
6.6.2.260 SQLite error [INTERNAL 207].....	405
6.6.2.261 SQLite error [INTERNAL 081].....	406
6.6.2.262 SQLite FTS Indexing Batch Failed [ERROR 313].....	406
6.6.2.263 SQLite FTS Indexing Failed [ERROR 312].....	406
6.6.2.264 ST parse error [INTERNAL 346].....	407
6.6.2.265 Station span * PID gain greater than 125.0 [INFO 179].....	407
6.6.2.266 Station span * PID gain less than 80.0 [INFO 178].....	408
6.6.2.267 Symbol flag detection failed [INTERNAL 287].....	409
6.6.2.268 System error total exceeded [BUILD 60001].....	410
6.6.2.269 Tag atom not upper case [CHECK 264].....	410
6.6.2.270 Tag failed to find matching format [INTERNAL 209].....	411
6.6.2.271 Tag index not found [ERROR 159].....	411
6.6.2.272 Tag index not found [ERROR 272].....	414
6.6.2.273 Tag index not found in database [ERROR 123].....	414
6.6.2.274 Tag mismatch [ERROR 101].....	415
6.6.2.275 Tag name not found in any database [ERROR 067].....	415
6.6.2.276 Tag name not upper case [ERROR 263].....	417
6.6.2.277 Tag not an exception report [ERROR 256].....	417
6.6.2.278 Tag not found by name [INTERNAL 293].....	418

Table of Contents

6 Build Reports and Analysis

6.6.2.279 Tag not found in any database [ERROR 284].....	418
6.6.2.280 Tag not found in correct database [ERROR 288].....	419
6.6.2.281 Tag not found where specified [ERROR 068].....	419
6.6.2.282 Tag referenced in PNT database has no LPMB info [COSMETIC 250].....	420
6.6.2.283 Tag renames block [ERROR 144].....	420
6.6.2.284 Tag type mismatch [ERROR 102].....	421
6.6.2.285 Tag used in graphic not exception-reported [ERROR 141].....	422
6.6.2.286 Tag/trend mismatch [ERROR 105].....	423
6.6.2.287 TagAtom mismatch [CHECK 229].....	423
6.6.2.288 Tagname is illegal in PCView [ERROR 104].....	424
6.6.2.289 Tested block on different page [CHECK 211].....	424
6.6.2.290 Tokenization unrecognized [INTERNAL 269].....	425
6.6.2.291 Too big for float [ERROR 10027].....	425
6.6.2.292 Too big for short: integer value [ERROR 10009].....	426
6.6.2.293 Too big for unsigned short [ERROR 10024].....	426
6.6.2.294 Too many fields [INTERNAL 223].....	427
6.6.2.295 Too many outputs [ERROR 137].....	428
6.6.2.296 Too many segment control blocks [ERROR 305].....	428
6.6.2.297 Too many tag databases for tagmap [BUILD 249].....	429
6.6.2.298 Too small for float [ERROR 10028].....	429
6.6.2.299 Too small for short [ERROR 10023].....	429
6.6.2.300 Too small for unsigned short [ERROR 10025].....	430
6.6.2.301 Trend block is wired from constant [COSMETIC 258].....	431
6.6.2.302 Trend block not used on any graphics or archived [CHECK 257].....	432
6.6.2.303 Trend definition file not found [CHECK 196].....	432
6.6.2.304 Trend does not have matching tag in any associated database [ERROR 107]..	433
6.6.2.305 Trend index (tag) does not reference a trend in any associated trend database [ERROR 142].....	433
6.6.2.306 Trend index does not reference a trend in any associated trend database [ERROR 143].....	434
6.6.2.307 Trend referenced, but no tags configured in any database [COSMETIC 254]....	434
6.6.2.308 Trend tag not found in any database [ERROR 163].....	434
6.6.2.309 Trend tag referenced by two tags [ERROR 161].....	435
6.6.2.310 Trend tag refers to missing or invalid entry [ERROR 162].....	435
6.6.2.311 Trip on fault (defined I/O point bad) [ERROR 148].....	436
6.6.2.312 Trip on fault (will trip on failure) [ERROR 147].....	437
6.6.2.313 TSTALM checks subsequent block [CHECK 212].....	438
6.6.2.314 TSTALM does not test base block [CHECK 310].....	439
6.6.2.315 TSTQ on import quality only [CHECK 070].....	439
6.6.2.316 TSTQ tests block not propagating quality [ERROR 071].....	440
6.6.2.317 TSTQ tests block with no quality [ERROR 063].....	440

Table of Contents

6 Build Reports and Analysis

6.6.2.318 TSTQ tests executive block [CHECK 156].....	441
6.6.2.319 Type code cannot be rendered [INTERNAL 278].....	442
6.6.2.320 Unable to find bitmap [BUILD 100].....	442
6.6.2.321 Unable to parse node [INTERNAL 027].....	443
6.6.2.322 Unconnected wire end [ERROR 339].....	443
6.6.2.323 Unexpected function code [INTERNAL 017].....	443
6.6.2.324 Unhandled calculation [ERROR 326].....	444
6.6.2.325 Unhandled complex source configuration [INTERNAL 341].....	444
6.6.2.326 Unhandled data after record [INTERNAL 234].....	444
6.6.2.327 Unhandled escape type [INTERNAL 236].....	445
6.6.2.328 Unhandled line [INTERNAL 237].....	445
6.6.2.329 Unhandled LMM spec value [INTERNAL 235].....	445
6.6.2.330 Unhandled pe header data [INTERNAL 233].....	446
6.6.2.331 Unknown AC 800M block type [INTERNAL 338].....	446
6.6.2.332 Unknown access code [INTERNAL 273].....	446
6.6.2.333 Unknown atom [CHECK 047].....	447
6.6.2.334 Unknown Bin file touchpoint type [ERROR 332].....	447
6.6.2.335 Unknown Component Number [INTERNAL 270].....	447
6.6.2.336 Unknown dynamic bar direction [INTERNAL 232].....	448
6.6.2.337 Unknown Exception [INTERNAL 053].....	448
6.6.2.338 Unknown logic state type [INTERNAL 245].....	448
6.6.2.339 Unknown mimic file touchpoint type [ERROR 330].....	449
6.6.2.340 Unknown picture type in ctx [INTERNAL 218].....	449
6.6.2.341 Unknown record type [INTERNAL 240].....	449
6.6.2.342 Unknown shape (missing library) [ERROR 042].....	450
6.6.2.343 Unknown shape (no user library) [ERROR 041].....	450
6.6.2.344 Unknown shape (not in library) [ERROR 043].....	451
6.6.2.345 Unknown subtype for block ed 50 [INTERNAL 242].....	451
6.6.2.346 Unknown subtype on line [INTERNAL 241].....	452
6.6.2.347 Unknown tag type [INTERNAL 259].....	452
6.6.2.348 Unknown type code [INTERNAL 276].....	452
6.6.2.349 Unkonwn mimic file type code [ERROR 331].....	453
6.6.2.350 Unparsable XML Line [ERROR 329].....	453
6.6.2.351 Unrecognized binary data [INTERNAL 274].....	453
6.6.2.352 Unreferenced graphic file [INFO 214].....	453
6.6.2.353 Unreferenced graphic name [INFO 215].....	454
6.6.2.354 Unresolved connection [ERROR 344].....	455
6.6.2.355 Unsupported dynamic symbol subtype [INTERNAL 239].....	455
6.6.2.356 Unsupported FC used in CFC [10045].....	455
6.6.2.357 Unsupported function [INTERNAL 238].....	456
6.6.2.358 Unsupported graphic primitive [INTERNAL 217].....	456

Table of Contents

6 Build Reports and Analysis

6.6.2.359 Unsupported non-literal tag name [INTERNAL 343].....	456
6.6.2.360 Unused extra inputs [INTERNAL 136].....	457
6.6.2.361 User macro registered multiple times [INFO 10039].....	457
6.6.2.362 Vertical text on drawing [INFO 050].....	458
6.6.2.363 Wired gain spec is adapted [INFO 120].....	458
6.6.2.364 X coordinate adapted [INFO 029].....	459
6.6.2.365 X coordinates out of order [ERROR 033].....	460
6.6.2.366 XOR output forced to 0 [CHECK 281].....	461
6.6.2.367 XOR output forced to 1 [CHECK 280].....	462
6.6.2.368 XOR output forced to complement input [CHECK 283].....	462
6.6.2.369 XOR output forced to track input [CHECK 282].....	463
6.6.2.370 XOR output forced to track input [CHECK 279].....	463
6.6.2.371 Y coordinate adapted [INFO 030].....	463
6.6.2.372 Zero gain on input block [CHECK 262].....	464
6.7 Other error checking techniques.....	465
6.7.1 Finding graphics links made by tag index instead of tag name.....	466
6.7.2 Using WinMerge to compare build reports.....	466
6.7.2.1 Compare Files.....	466
6.7.2.2 Compare Directories.....	467

7 Using CIULink.....468

7.1 Running CIULink.....	468
7.2 Running CIULink as a Viewer.....	469
7.2.1 Configuring CIULink.....	470
7.2.2 Adding Points for Monitoring.....	471
7.2.3 Set Update Interval.....	472
7.2.4 Selecting Viewing Options.....	472
7.2.4.1 Debug View.....	473
7.2.4.2 Export View.....	473
7.2.5 Monitoring data points.....	474
7.3 Logging with CIULink.....	474
7.3.1 Log New Data as It Arrives.....	474
7.3.2 Log Data Already Collected.....	475
7.3.3 Choosing where to save logged data.....	475
7.3.4 Logging Options.....	477
7.3.5 Reading CIULink logs.....	478
7.3.6 Understanding the format of CIULink logs.....	478
7.3.7 Dates and times in CIULink logs.....	479
7.3.8 Displaying dates and times in a spreadsheet.....	479
7.4 Loading, Saving and Deleting Tag Lists.....	480
7.4.1 Export Tag List.....	481

Table of Contents

7 Using CIULink

7.4.2 Import Tag List.....	482
7.4.3 Significant Digits.....	482
7.5 Running from Microsoft® Excel.....	483
7.5.1 Creating a new workbook.....	483
7.5.2 Export lists from CIULink and import them into Excel.....	483
7.5.3 Running from a workbook with existing DDE function.....	484
7.6 Running from Microsoft® Access.....	484

8 DBDOC Installation and Setup.....485

8.1 Overview of installing, updating, and administering DBDOC.....	485
8.2 Installing DBDOC.....	485
8.2.1 Installing the DBDOC build system.....	486
8.2.2 Installing the CIUMon online data server.....	487
8.3 Installation of CIUMon.....	487
8.4 Privileges and Setup.....	487
8.5 Installing more than one CIUMon.....	488
8.5.1 Sharing Hyperview and the M14 over a network.....	488
8.6 Choose a location for the shared folder.....	488
8.7 The Hyperview installer.....	488
8.7.1 Option 1: Install Hyperview directly to a network share.....	489
8.7.2 Option 2: Install Hyperview locally first, then create a network share.....	489
8.8 Configure BuildPlus to automatically copy project files to the shared location.....	489
8.8.1 Installing Hyperview on an individual computer.....	490
8.8.2 Installing the 800xA DLLs for Hyperview support of 800xA graphics.....	490
8.8.3 Installing the DBDOC Demo.....	491
8.9 Operating system requirements and installation notes.....	491
8.9.1 DBDOC build system hardware performance requirements.....	492
8.9.2 Working with the DBDOC security dongle.....	492
8.9.3 What happens when I install over a previous version of DBDOC?.....	493
8.10 Updating your DBDOC installation.....	493
8.10.1 Step 1: Rebuild your project before updating.....	494
8.10.1.1 Reasons for rebuilding first.....	494
8.10.1.2 How do you rebuild?.....	494
8.10.1.3 What to do after a successful build.....	494
8.10.2 Step 2: Update CIUMon.....	494
8.10.2.1 Stop CIUMon.....	494
8.10.2.2 Install the new version of CIUMon in the same location.....	494
8.10.2.2.1 A: If CIUMon is installed in C:\Program Files\GMCL\DBDOC\Programs.....	495
8.10.2.2.2 B: If CIUMon is installed in C:\DBDOC\CIUMon\Programs.....	495
8.10.2.3 Verify that the new version works.....	495
8.10.2.4 Restart CIUMon services.....	495

Table of Contents

8 DBDOC Installation and Setup

8.10.3 Step 3: Update the Build System.....	496
8.10.3.1 Install the new version.....	496
8.10.4 Step 4: Updating Hyperview.....	496
8.10.4.1 Option 1: Updating Hyperview when nobody is using it.....	496
8.10.4.1.1 Find or schedule a time for an update.....	496
8.10.4.1.2 Check the current install location.....	496
8.10.4.1.3 Install the new version in the same location.....	497
8.10.4.1.4 Make sure that the new version works.....	497
8.10.4.2 Option 2: Updating Hyperview when users may be using it.....	497
8.10.4.2.1 Check the current install location.....	497
8.10.4.2.2 Install the new version in the same location.....	498
8.10.4.2.3 Make sure that the new version works.....	498
8.10.5 Step 5: Rebuild your project.....	499
8.10.5.1 Rebuild in BuildPlus.....	499
8.10.5.2 Make sure that Hyperview users can access the new project file.....	499
8.10.5.3 Make sure that Hyperview users can view live data.....	499
8.11 Licensing DBDOC.....	499
8.11.1 How many documents can I build?.....	501
8.11.2 When your license expires.....	501

1 DBDOC Introduction

1.1 DBDOC 11.6

This manual describes the components in DBDOC 11.6, released November 2025.

1.2 What is DBDOC?

DBDOC builds a **safe, read-only snapshot** of your Harmony INFI 90[®] DCS system, presenting an **integrated view** of the configuration, graphics and databases. From the first version handling CADEWS and SODG graphics, DBDOC has evolved to include WinCAD[®], Composer[®], Conductor VMS[®], Conductor NT[®], Process Portal B[®], AutoCAD[®], MicroStation[®], supplementary text and even disparate control system components, such as P&IDs, WonderWare[®], Honeywell[®] and OSISoft PI[®]. More recently, support for 800xA PG2, PGP and SPlus graphics has been added to DBDOC. *Most* recently, AC 800M system support was added with DBDOC 11.5.

DBDOC is in many ways a cooperative project with our clients, users of ABB[®] Harmony INFI 90 systems. We rely heavily on your feedback to enhance and improve the tools that we provide. Many of DBDOC's present features have been developed as a direct result of client requests and feedback. G. Michaels Consulting Ltd is working in cooperation with ABB to further the development of DBDOC.

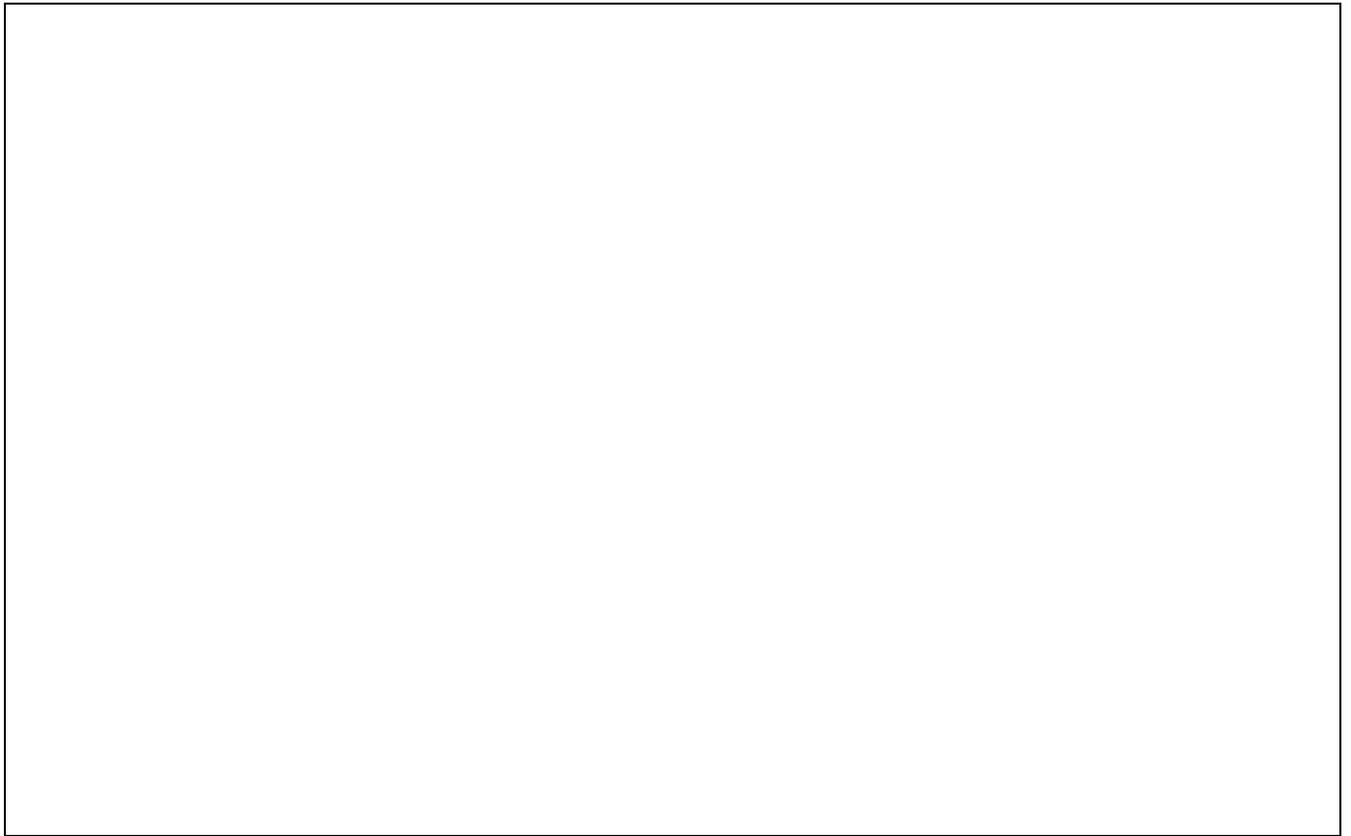
1.3 Who Can Benefit From DBDOC?

Getting DBDOC into the hands of more people that use the ABB Harmony INFI 90[®] system will pay off with easier and better execution of any tasks they have to do. This applies to many types of workers, including technicians, engineers, planners, managers, trainers, safety and loss prevention personnel, students, interns and operators.

1.4 How Does DBDOC Work?

The DBDOC snapshot, a single file, is like a custom website representing your system. It is an **amazing interface for troubleshooting and analysis**. Every document and resource is cross-linked, and with the DBDOC browser you can navigate using the intuitive point and click interface. Powerful search capabilities give instant access to any part of your system. Built-in tools and analysis reports can help you eliminate common misconfigurations and improve your system. The DBDOC compiler builds a representation of your system in the form of a .dbdoc (previously .M14) file. This project file can be rebuilt regularly, and is viewed using the DBDOC browser. Live data is fetched from your system and presented on the graphics and configuration diagrams displayed in the DBDOC browser. All this is completely safe and read-only.

1.5 System Overview



1.6 How is DBDOC Used?

DBDOC meets a variety of needs for users of Harmony INFI 90® systems:

1. **Troubleshooting:** DBDOC is an unrivalled tool for system analysis and troubleshooting.
2. **Documentation:** Integrate notes, documentation, and error reports from any source into the system snapshot. Tag names in the documentation are automatically cross-linked with the rest of the system.
3. **Development:** Check the system, use Management of Change to keep on top of updates, use DBDOC's error checks to catch problems, enjoy faster browsing of the configuration.
4. **Commissioning and Checkout:** Save time with DBDOC during the implementation of a new system or subsystem.
5. **System Simulation:** See the configuration and use emulated CIU data. Manually walk through the response to proposed inputs.
6. **Training:** Safely introduce new personnel to the configuration. Track through the logic from start to end, or to the source.
7. **Detection of Errors and Misconfigurations:** Use DBDOC's Error Browser to view your errors and eliminate potential problems.
8. **Portable System:** The DBDOC system snapshot is portable, and lets you view your system remotely.

9. **NERC Compliant Live Data on Business LAN:** DBDOC can provide a remote NERC compliant view of your system complete with live data.
10. **Collaboration Tool:** With DBDOC you can annotate drawings and graphics and integrate notes, making it easy to share
11. **Console Upgrades:** Use DBDOC to understand where various tags and fields are used.
12. **System Audits:** Check out the whole system in an easy, efficient and systematic manner.

1.6.1 DBDOC for Troubleshooting

DBDOC's unique point and click browser interface makes it easy to trace signals throughout your system. Every resource is cross-linked and at your fingertips, making for efficient and effective troubleshooting. Live data from your system is integrated into the DBDOC snapshot providing even more information for productive troubleshooting.

DBDOC increases the number of people who can safely and effectively troubleshoot routine system problems. System experts can be freed up to focus on other tasks.

1.6.2 DBDOC as Documentation

DBDOC can integrate existing text and RTF documentation right into the system snapshot. You simply create a documentation folder and tell the Build Wizard its location. You can also build in archived error files to document fixes in your system.

DBDOC automatically links tag names mentioned in documentation into the system snapshot. This feature allows the creation of important documentation useful in the operation and support of the system with no extra effort on your part.

1.6.2.1 Uses of DBDOC for Documentation

DBDOC has been used by clients to enhance system documentation in a variety of ways, including the following:

1. **Alarm Documentation** - Simply documenting an alarm tag with its name gets a link to it that people can look up easily. Mentioning associated tag names, graphics and other documents will give you links from the information about the alarm in question to other parts of the system.
2. **Calculation Support** - A simple document describing calculated values gives you a permanent record of how the values are calculated and the tags that are involved. This can serve you well when changes are contemplated by preventing the inadvertent elimination of tags that are in use in more or less obscure ways.
3. **Logs and Reports** - Creating a text / document form of data logs and reports that mention the tagname gives you instant documentation to assist in tracking down bad values.

1.6.2.2 Optional Documentation Extensions to DBDOC

DBDOC can also be used for block number linked documentation. If you have tables of information that mention block numbers for specific reasons, we can optionally (at additional cost) extend DBDOC to compile the documentation with links on the blocks. To do this, we need a Loop / PCU / Module context. This can be created easily by folder names or document names, or by the inclusion of definitive statements in the document.

1.6.3 DBDOC During Development

DBDOC can be used to support development work in a rapidly changing environment in a number of ways:

- With DBDOC, you can build a new snapshot of your system nightly, and keep up with changes in configuration.
- Use DBDOC to browse the logic instead of Composer or WinCAD.
- Use DBDOC's powerful Management of Change capability to monitor daily changes.
- Use DBDOC's error-checking capabilities to catch problems before you compile modules.

1.6.4 DBDOC for Commissioning and Checkout

If you put DBDOC to use at the right place at the right time, it will help support the implementation of a new system or subsystem. People that have used it this way have been enthusiastic in its praise. It should be possible for a project to be done better, cheaper and faster using DBDOC.

1.6.4.1 Experiences with DBDOC for Commissioning and Checkout

DBDOC has been widely used as an aid to the commissioning and checkout process. The following are the experiences of several sites:

- **Hibernia Oil Platform, Newfoundland, Canada** - Our first implementation. They claimed that DBDOC saved them one month in startup time in a multi-billion dollar project.
- **Aluminum Partners of Jamaica, Nain, St. Elizabeth, Jamaica** - In this case, DBDOC was actually purchased before the INFI 90 system. DBDOC was used to explain the methodology implemented at a partner aluminum plant using INFI 90.
- **Yallourn "W" Power Station, Yallourn, Victoria, Australia** - DBDOC was used over a number of years as this plant was implemented unit by unit.

1.6.5 Using DBDOC with a Simulator

DBDOC is a powerful aid in simulation projects as you are able to see the logic. Simulations that include emulated CIU support gain a great deal from its use. This topic has significant potential for application in many systems:

- Before you download a new or changed module, build it in DBDOC.
- See how typical live values look in your new logic.

- Use DBDOC's error-checking and nitpicking to find and fix problems before they're online.

Note: This capability is based on the existence of a simulation system such as TRAX that uses a CIU style interface to logic that gets valid block values.

1.6.6 DBDOC as a Training Tool

DBDOC's read-only unified presentation of the consoles and configuration make it a powerful, safe, and easy-to use training tool for new personnel ranging from operators to engineers. Other training-friendly features include:

- Built-in help for function codes.
- A flattened view of the databases to make it possible to become more familiar with tags in the system.
- Unparalleled support for signal tracing in order to become familiar with a system.

1.6.7 DBDOC for Error Detection

DBDOC checks your system for hundreds of types of errors and misconfigurations, with more added in each update. In addition to actual errors, ambiguous non-error situations in your system such as extra wires, constant blocks in ladder logic and unused tags are reported. Usually these non-errors cause no trouble, but sometimes they can mask real problems.

DBDOC makes it easy to clean up your system, identifying undefined tags, tags with no source, missing or out of order adapted blocks, graphics that cannot be triggered by the operators, and hundreds of other misconfigurations and potential issues of varying severity.

DBDOC can make it possible to find parts of the logic that are unused. If you walk through the AO/L and DO/L blocks, for example, you will find a subset that are not used. If you also find that they are not alarm points, you might choose to re-evaluate the logic that creates them. Similarly, tags often become unused without being removed.

The DBDOC Error Browser, part of Hyperview, allows you to walk through these errors, automatically display error locations in Hyperview, filter, group, and sort errors in various ways, and hide or mark off errors as "reviewed" so you don't have to be repeatedly checking the same set of errors once you have reviewed them once. You can even share the process of reviewing errors, with error flags set by others automatically displayed in the Error Browser.

1.6.8 DBDOC as a Portable System

DBDOC enables you to put a copy of your system on your laptop so that you are able to:

- Troubleshoot from home
- Support the system when you are on the road

- Laptops, notebooks, and tablet PCs, coupled with wireless LANs, give you the ability to put DBDOC out in the field, far from AC power and networks and use DBDOC anywhere in your plant for troubleshooting and other analysis.
- With a WAN or dial-in, you can even monitor live data remotely.
- DBDOC allows you to do a static analysis of you system anywhere, offline.

The Serial Port Module is a relatively inexpensive way to get realtime data directly to your laptop at a PCU. We went to the trouble of marking data that comes from another PCU in blue so you do can see values you cannot get directly in that PCU.

1.6.9 DBDOC for NERC Compliant Live Data on Business LAN

DBDOC allows you to safely troubleshoot, perform system analysis, and train support staff on your business LAN, as well as read-only access to your system, at home, on courses, and in other parts of the plant. For total NERC compliance, a relay server can be used to pass live data via an intermediate LAN.

1.6.10 DBDOC as a Collaboration Tool

DBDOC can be used as a collaboration tool in many ways. It can be used to:

- Send your consultants a copy of your system as built.
- Mark up your system with annotations and bookmarks.
- Share your annotations and bookmark files with your colleagues via e-mail.

1.6.11 DBDOC for Console Upgrades

With DBDOC you are able to build your old console and new console together in a single project. You can then use DBDOC's Audit Window feature to verify that the new graphic accesses the same tags as the old one.

1.6.12 DBDOC for System Audits

DBDOC gives you unprecedented capability to check things easily. When you have something to do, just think of the old way that you had to do it, before DBDOC. For example, one client told us that a significant safety audit was completed by an engineer in ten days instead of the estimated thirty days.

DBDOC allows you to walk through all function blocks of particular types to check them. You can check parameters, tag names, documentation that should be linked and a multitude of other things. This capability extends to checking ABB software revisions directly by walking through the blocks with a particular function code.

The Audit Window allows you to look at the blocks and tags being used on graphics or CAD/CLD sheets. Sometimes, this shows incorrectly entered data. Asserting the hotspots (**Ctrl+Tab**) is a useful feature in looking for "dead spots" on graphics.

2 Hyperview Browser

2.1 How to run Hyperview, load a project file, and get help

DBDOC organises all your INFI 90 documents into a single package to give you fast, easy, read-only access to the information you need to do your job. Hyperview is the browser that navigates through this view of your system.

2.1.1 How to Run Hyperview

2.1.1.1 Starting Hyperview from a desktop shortcut

Usually Hyperview users are provided with a desktop Hyperview shortcut, allowing Hyperview to be maintained in a central location accessible to many users. To run Hyperview, double click on the shortcut. The last project file viewed will be automatically loaded.

2.1.1.2 Starting Hyperview from the Start Menu

If Hyperview is installed directly on your, machine, access it via the **Start** menu - under **Programs** look for either a **GMCL DBDOC** or **GMCL DBDOC Hyperview** folder and click on the **Run Hyperview** option.

2.1.1.3 Launching an additional Hyperview window

You can also launch a new Hyperview window by selecting the **New Window** option from the **File** menu inside of Hyperview. This will launch another copy of Hyperview viewing the same project and topic that you are currently viewing. The new copy will inherit the same command line parameters as the copy that spawned it.

2.1.1.4 Advanced: Running Hyperview from the command line

Command Line Options:

You can launch Hyperview via the command line. You may specify the name of the project file to open, as well as, optionally, a particular document within the project to display when it first opens up. You may also append additional optional arguments.

The command line takes the following form:

```
"<Installed_program_location>\hyperview.exe"  
    <dbdoc file>  
    </document_switch=document_name>  
    </additional_optional_switches>
```

In practice this will take one of three forms:

Specifying a general document to open:

```
"<Installed_program_location>\hyperview.exe"  
  <project file>  
  </doc=document_name> </type=document_type>  
  </additional_optional_switches>
```

Specifying a block whose source in the configuration will be displayed:

```
"<Installed_program_location>\hyperview.exe"  
  <project file>  
  </block=block_specification>  
  </additional_optional_switches>
```

Specifying a tag whose source in the configuration will be displayed:

```
"<Installed_program_location>\hyperview.exe"  
  <project file>  
  </tag=tag_name>  
  </additional_optional_switches>
```

Command line arguments:

- **<project file>** is the location and name of the .dbdoc (previously .M14) file to open.
 - ◆ In lieu of an actual filename, you can use the asterisk (*) wildcard to open the newest project file in the location specified.
 - ◆ If no filename is present, Hyperview will open the last project file that you were viewing.
- **</doc=document_name>** is the name of the document (without suffix) to open.

-- document_switch is one of the following arguments followed by the **document_name** (do not include the file extension in the name):

Argument	Explanation	Example
/doc	<p>Specify a document to open. The optional <i>/type</i> switch specifies the document type to be opened. The first document (of the specified type) that has a name with a leading substring that matches the string specified in <i>/doc</i> will be opened. If <i>/type</i> is not specified, then the first matching document of any type will be opened.</p> <p>Strictly speaking, it is only necessary to specify <i>/type</i> in cases of document name ambiguity. A wide variety of types, such as graphic, cad, dxf etc. can be specified. See Launching Hyperview from the command line in the Hyperview Reference Guide for complete</p>	<pre>/doc="G1WATER" /type=graphic</pre>

	information.	
/tag	Tag Name	/tag="AMD-108" (Note: This will bring up the tag index in the Table of Contents panel and the CAD or CLD source, if it exists, in the content pane.)
/block	Database record	/block="Module 1,1,1 Block 1" (Note: The index pane should show everywhere this block is referenced. The content pane will display the CAD or CLD sheet source of the specified block.)

2.1.2 How to load a project file

2.1.2.1 Opening a specific project file

To open a different project file follow the following steps:

1. Go to the **File** menu and click **Open**.
2. A standard Windows browse dialog box will open.
3. Navigate to the location of the file.
4. Click **Open**.

2.1.2.2 Opening a recent project file

You are also able to navigate to the **File** menu and click the **Recent Project Files** selection to pick a project file from the last twelve files that were viewed.

The name of the last viewed project file is now stored in the registry. When Hyperview is run from the **Start** button, it will be loaded with the last project viewed. Hyperview also stores the names of the last twelve viewed project files and lists them in the **Recent Project Files** option under the **File** menu. This allows for quick navigation between recently viewed project files.

2.1.2.3 Advanced: Opening the most recently built project file

Shortcuts to Hyperview can specify a wildcard project file specification (i.e., containing *). Hyperview opens the newest file that matches the wildcard specification. The advantage of using wildcards like this is that you do not have to rework your shortcuts with each new build and you can use the automatic renaming technique in BuildPlus to archive your built project files.

2.1.2.4 Advanced: Associating project files with Hyperview in Windows

To make Hyperview open when you double-click a project file, you can associate the project file type (.dbdoc, previously .M14) with Hyperview in Windows by following the steps below:

1. Go to the **Start** menu, click on **Control Panel**, locate the **Folder Options** icon and double-click on it.
2. Once the **Folder Options** window is open click on the **File Types** tab.
3. Click the **New** button below the Registered File Types.
4. In the **File Extension** box enter **dbdoc** as the File Type.
5. Once it is added click the **Change** button below the **File Types** box.
6. Choose the select the program to open the file from a list selection.
7. Browse to `C:\Program Files\GMCL\DBDOC\Programs` and select `hyperview.exe` as the program to open the file.

If you are using Windows 11, you may need to do the following steps:

1. Go to the **Start** menu and click on **settings**.
2. Select **Apps** and then **default apps**.
3. In the search bar under "**Set a default for a file type or link type**" type **.dbdoc**.
4. Click on the DBDOC file option and choose Hyperview from the listed apps. If Hyperview does not appear, you will need to scroll to the bottom, select "**Choose an app on your PC**", and then select the `hyperview.exe` in your install location (usually program files(x86)\GMCL\DBDOC\Programs\hyperview.exe)

2.1.3 Help and Documentation

2.1.3.1 How to access the built-in help documentation

The **Contents**, **Index**, or **Search for Help** on the Hyperview **Help** menu will launch the DBDOC compiled help. Selecting any one of these will launch the DBDOC compiled help, opened on the appropriately named tab. The help file contains information on how to use BuildPlus, Hyperview, and CIUMon, appendices on other programs and components, and our FAQ (Frequently Asked Questions) with useful information about builds, set-ups or other problems you may encounter.

2.1.3.2 Help and tutorials on the GMCL website

Go to GMCL website on the Hyperview **Help** menu will launch Internet Explorer and take you to the Help page on the GMCL website. There you can access all the same information as in the built-in help, as well as interactive tutorials, user guides, and other documentation.

2.1.3.3 Finding the version of Hyperlink that built the current project file

About the build and project file on the Hyperview **Help** menu will bring up a dialog box that contains information about the current project being viewed - the version of hyperlink that was used to build it and when it was last built.

2.1.3.4 Figuring out what version of Hyperview is running

About on the Hyperview **Help** menu contains the following information about the version of Hyperview currently running:

- Hyperview version
- DBDOC Kit Version
- Winsock / No Winsock
- Mode (user or system)
- Command line used to launch Hyperview

2.1.3.5 Other ways of getting help

You may also click on various items to get more information. If it is a dialog box that you need more information on, use the associated contextual help. Click the **Help** button in the upper right hand corner of the dialog box and then click on the desired item to see a description.

2.2 Guide to basic navigation in Hyperview

2.2.1 Panes in Hyperview

The Hyperview browser shows two panes, or display areas. The pane on the left (or top) is called the index, or **Table of Contents** pane. It is used for indexes and summaries of information. The pane on the right (or bottom) is called the content, or topic pane. It is generally used to display documents and graphics, although certain types of indexes and reports can also be displayed there.

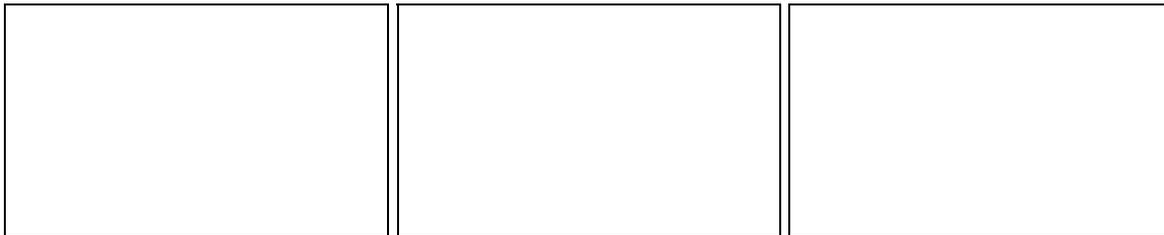
2.2.1.1 Specifying the pane orientation

You can set Hyperview up so that the index pane can be located at the left or top while the content pane will be located on the right or bottom respectively.

To change the orientation of Hyperview, click one of the following buttons in the toolbar:



They will produce the following results:



You can use the following menu options or commands to change the view from a menu or shortcut key:

- Select **Change Pane Orientation** from the **View** menu.
- Select the **View** from the drop down list on the **View** tab in **Options**.
- Position the cursor on the border until it becomes the split cursor and drag the cursor to the view that you would like.
- Press **J** on the keyboard or select **Split Adjust** from the **View** menu to get the split cursor.
- Adjust the view to full screen by pressing **F** on the keyboard, selecting **Full Screen** from the **View** menu or double-clicking the border between the two panels.
- Drag the borders between the index and content panes until the index pane disappears.

Your viewing preferences are stored so that each time Hyperview is launched it will open to your last used settings.

The settings it saves are:

- Size of the Hyperview window

- Position of the window
- Size of index and content panes
- Classic or Treeview Table of Contents

2.2.2 Index pane

When you first start Hyperview, the index pane provides a view of the **Table of Contents**, a summary of all the content contained in your project file.

2.2.2.1 Displaying the Table of Contents

To display the general Table of Contents:

- Click the Table of Contents toolbar button 
- **OR** press **C**
- **OR** on the **Go** menu, click **Table of Contents**.

2.2.2.2 Switching between Classic and Treeview Table of Contents

There are two ways of displaying the **Table of Contents**. The first view is **Classic View**. It provides a detailed view of your build content, but you generally have to scroll down a long way to get to what you want. **Treeview** is a compressed, collapsible/expandable, easy-to-use overview, similar to Windows Explorer.

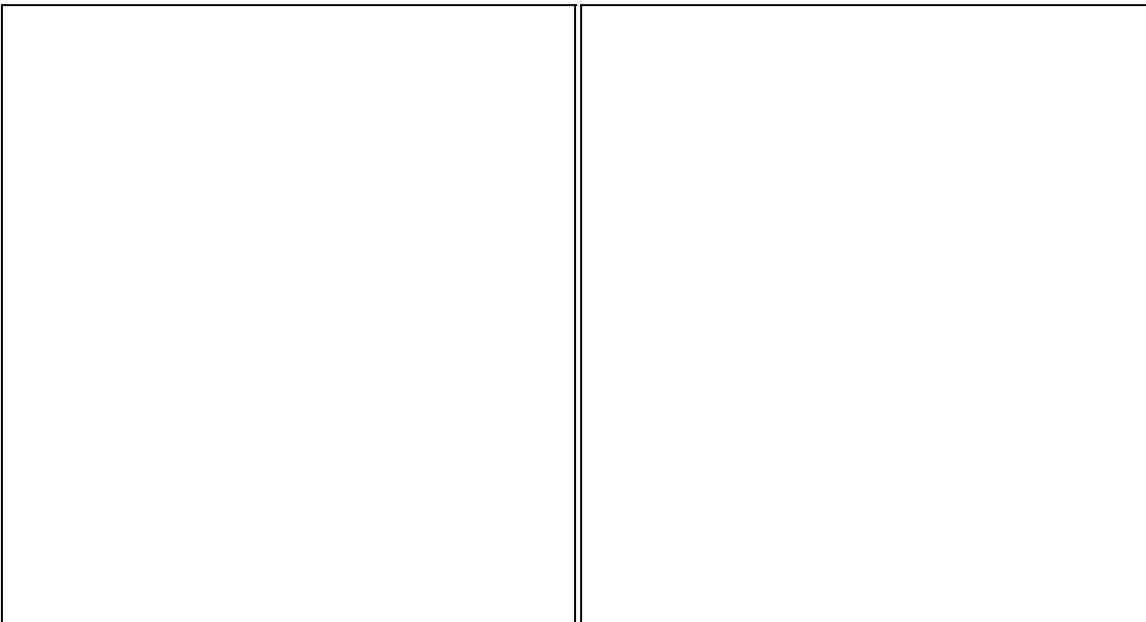
To switch between views for the Table of Contents:

- Press **Ctrl+T**.
- **OR** on the **View** menu, click **Table of Contents as Tree**.

To collapse the open nodes in the tree view table of contents:

- Press **C** .
- **OR** click the **Table of contents** toolbar button .

The following are examples of the the **Classic** (left) and **Treeview** (right) table of contents.



2.2.2.3 Clicking on hotspots in the index pane

Some of the items in the index pane are hotspots, or hyperlinks (indicated in blue). In general, clicking on a name-of-document hotspot in the **Table of Contents** causes that document to be displayed in the content pane below or on the right. The exception is when the item in the index pane is the name of another index. Clicking on a name-of-index hotspot in the **Table of Contents** will cause the new index to be loaded into the index pane. Hyperview indexes will be discussed in more detail later.

Rarely, the hotspot in the index pane is a link to an external document, i.e. one that is not built into the project file. When this is the case, a "proxy" document (with a title and link to the "real" document) is displayed in the content pane. Clicking on the link in the proxy document causes the external document (e.g. a PDF file with function code documentation) to be opened in your default PDF viewer application.

2.2.2.4 Index navigation

- **Go to Next/Previous Topic**
 - ◆ Go to the next/previous topic in the browse sequence.
 - ◆ This is dependent on which side the focus is on.
 - ◆ Use **B** and **Shift-B** on the keyboard or select from the **Go** menu.
- **Go to Next/Previous Index link**
 - ◆ In an index, go to the next/previous hyperlink in sequence.
 - ◆ Use **L** and **Shift-L** on the keyboard or select from the **Go** menu.

2.2.3 Content pane

The content pane is where documents such as Operator Graphics, CAD/CLD diagrams, reports, text documents, Tag and Trend Database entries, and other resources are displayed. These documents generally contain hotspots, or hyperlinks. Clicking on different sorts of hotspots will have different effects in the Hyperview browser, depending on the type of hotspot. Some link directly to other documents, while

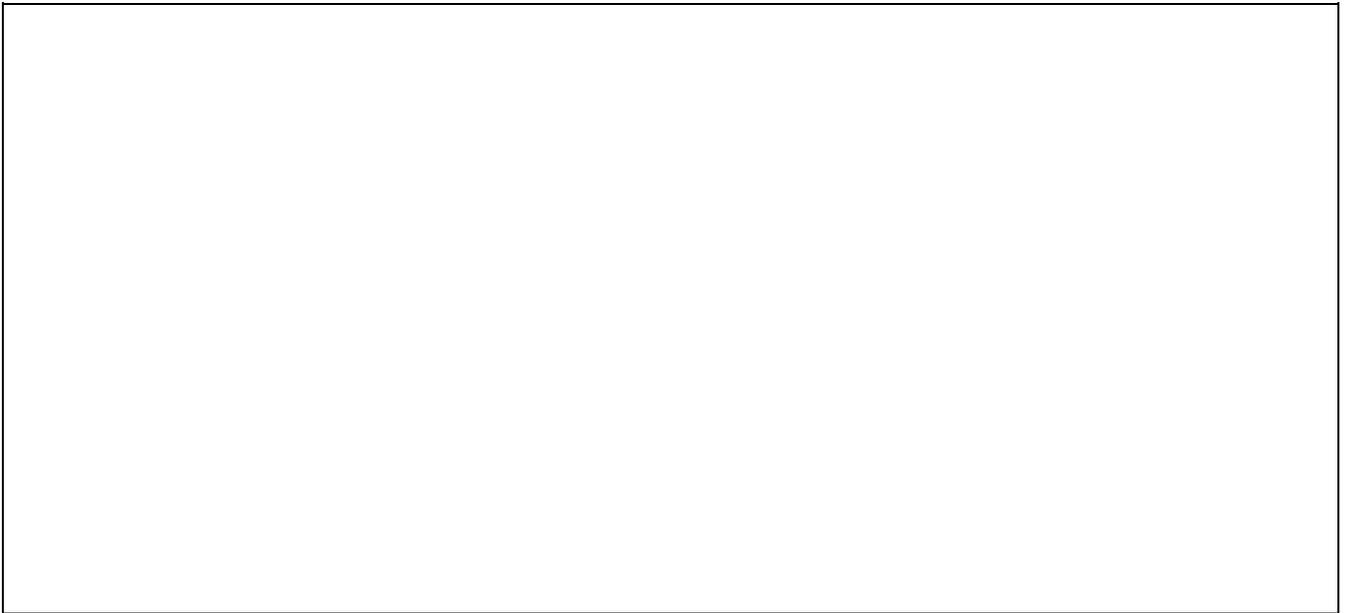
others represent blocks and allow signal tracing.

2.2.3.1 Displaying hotspots

There are typically many hotspots on a document in the content pane. When you move the mouse over a hotspot, the cursor will change to a "pointing finger" shape. You can also highlight all the hotspots on a page by pressing **Ctrl+Tab** or selecting **Hotspots** on the **Highlight** menu. You can also hover over a hotspot to see a brief description in a tooltip.

2.2.3.2 Clicking on a hotspot to see all the uses of a value

Many hotspots represent a block or tag value. Clicking on a block or tag hotspot will cause all the places the block or tag is used to be displayed in the index pane.



Clicking on the value on the right brought up all the places in the system that that block is used, both as a tag name and as a block. In this example, the selected block appears in the tag definition, a CAD/CLD configuration sheet, and on this graphic. All this information is right in front of you.

2.2.3.3 Double clicking on a hotspot to go to the source of a value

Most hotspots are associated with blocks in the configuration. Double click on any block-hotspot to display the block source in the configuration.

2.2.4 Focus for the active window

Focus is shown by the color border on the index or content pane and shows which side of the browser you are working in. It affects printing and the **Next Topic** and **Previous Topic** buttons. To set focus in a particular pane, click the mouse on some white space (where the cursor remains an arrow) in that pane.

2.2.5 Setting a "Home Page" for your index and content panes

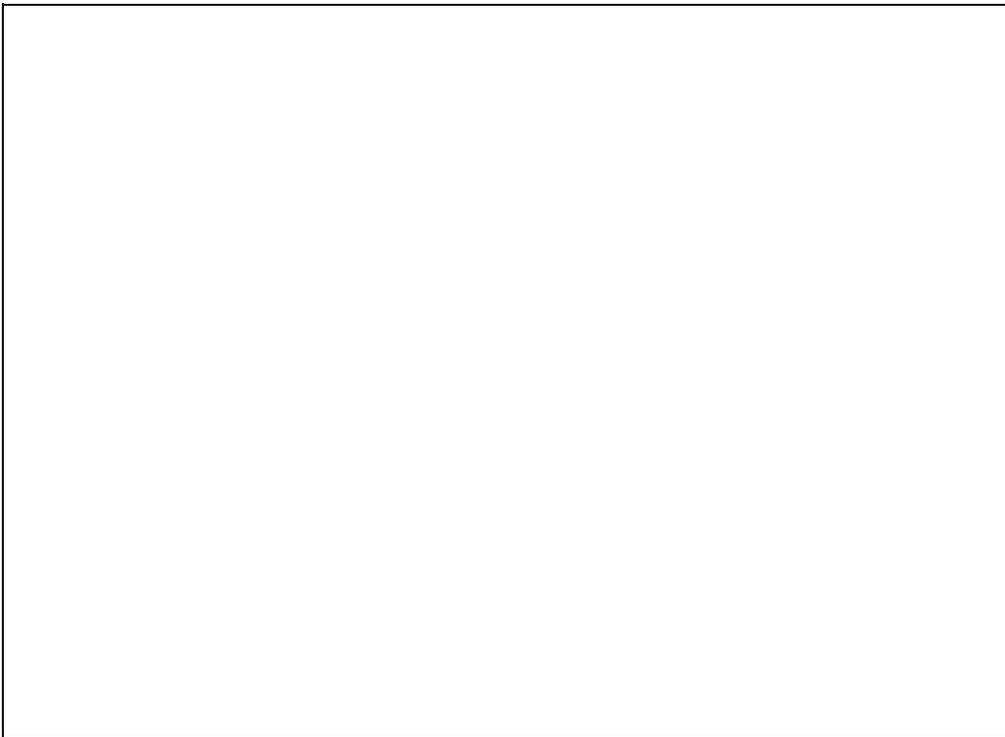
You can specify home pages for your index and content panes. If none are specified, then the **Home** button will return you to the classic opening view.

2.2.5.1 Setting your "Home Page" to the current document

You can easily set your homepage to a topic that is currently displayed in the Hyperview browser.

- Right-click on a topic. In the popup menu that appears, you can easily set the home button to call up the desired page using **Make This Topic My Index Pane Home Page** (if the index pane has focus) or **Make This Topic My Content Pane Home Page** (if the content pane has focus).

Your home pages are saved in your user logon (in the same location as your annotation, bookmark and history files). They will stay intact when the DBDOC files are updated and the project rebuilt. Typical content pane home pages are overview graphics that get you anywhere in the plant in one click.



2.2.6 Scrolling

Note: As of DBDOC 11.3, Hyperview supports two different scroll and zoom modes. The original mode (can be selected as **Legacy scroll & zoom mode** in the View tab of the Options Dialog) supports scrolling via the scrollbar and scroll wheel. The new mode, now the default, allows scrolling by dragging, and the scroll wheel zooms. Holding down the **Ctrl** key will temporarily enable the non-selected mode.

2.2.6.1 Default mode scrolling

By default, graphical documents can be scrolled with the scrollbar, or by dragging with the mouse. The scroll wheel now zooms graphical documents. Text documents can be scrolled with the scrollbar or with the scroll wheel.

2.2.6.2 Legacy scroll & zoom mode scrolling

When Hyperview is in **Legacy scroll & zoom mode**, all documents can be scrolled with the scrollbar or scroll wheel. Dragging does not scroll, but rather zooms on graphical documents.

Note that the unselected mode can be enabled transiently by holding down the **Ctrl** key. So, you could access drag-panning even in **Legacy scroll & zoom mode** by pressing **Ctrl**, and then dragging with the mouse.

2.2.7 Zooming

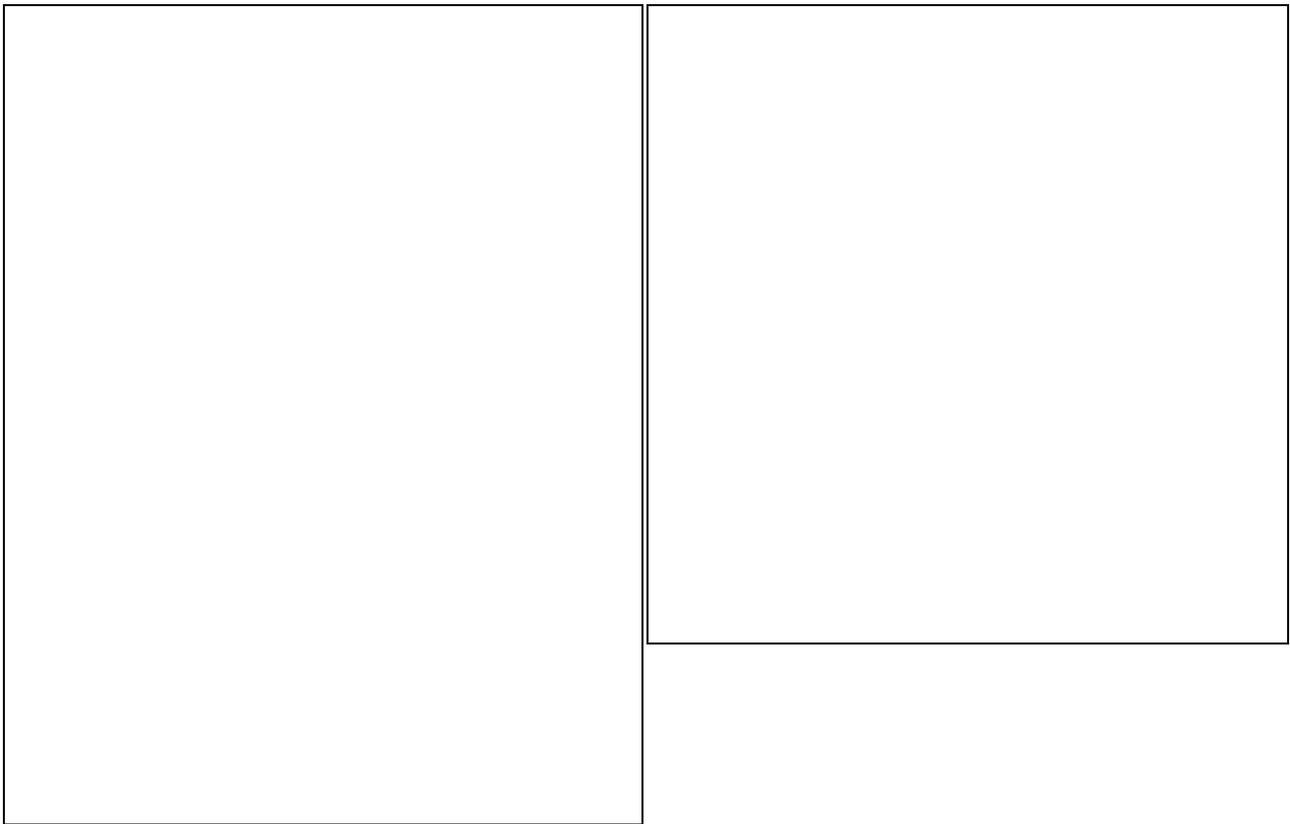
Note: As of DBDOC 11.3, Hyperview supports two different scroll and zoom modes. The original mode (can be selected as **Legacy scroll & zoom mode** in the View tab of the Options Dialog) supports "rubber-band" zooming on graphical documents. The new mode, now the default, enables scroll wheel zooming. Holding down the **Ctrl** key will temporarily enable the non-selected mode.

2.2.7.1 Default mode zooming with the mouse

Place the mouse above the spot you wish to zoom in or out on, and scroll up with the scroll wheel to zoom in, or scroll down to zoom out.

2.2.7.2 Legacy scroll & zoom mode zooming with the mouse

Click and drag a rectangle in the content pane around the area that you want enlarged. When you have finished dragging, Hyperview will centre your selection in the pane, magnified to fit.



2.2.7.3 Zooming with toolbar buttons or the keyboard

- To zoom in to double the zoom level, click the **Zoom In** button  or press +
- To zoom out to half the zoom level, click the **Zoom Out** button  or press -
- To toggle between full view and the previous zoom level, click the **Zoom Toggle** button  or press **Z**
- Zoom functions are also available through the **View** menu.

2.2.7.4 Manually setting the magnification factor

You can manually set the magnification factor for any graphical topic (graphic, CAD/CLD sheet and AutoCAD drawing).

- Right click on the drawing and select **Set Zoom Factor** from the popup menu.
- Enter the zoom factor you wish to apply to the topic.



2.2.8 Using browser history to return to previously viewed documents

Hyperview history is similar to that of other browsers like Internet Explorer.

Going backward and forward in the browser history:

- Click the **Go Back** button or **Go Forward** button to go to

the previous or next item.

- Activate the drop down menu for the forward and backward lists by clicking on the arrow to the right of the button and selecting the desired item.
- The last document viewed is retained across builds. This is stored in `<project_name>.hst` in the same location as your annotation, bookmark and home page files.

2.2.9 Using bookmarks for quick access to pages of interest

Flag special pages for quick access! These can be added to any document.

To bookmark a topic:

- Click on the **Add Bookmark** button on the toolbar.
- **OR** Choose **Define** under the **Bookmark** menu



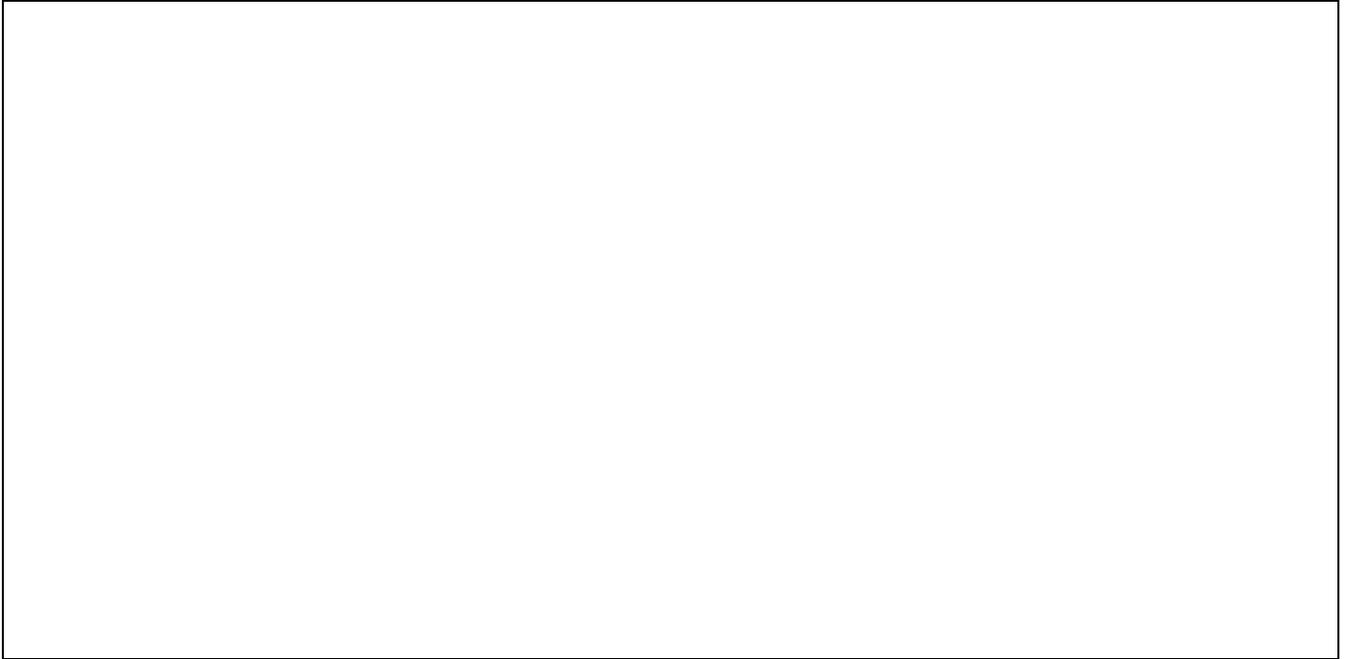
- **OR** Press **Ctrl+D** to add the document directly. This option will add the document to the bookmark list without bringing up the **Add Bookmark** dialog where you can edit how the text will read in the list.



Multiple bookmarks with the same name are not allowed. If you try to define a bookmark with a name that already exists, Hyperview will bring up a message box to warn you of the duplication. It will give you the option of editing the current bookmark name or canceling the operation.

Using and managing bookmarks:

Bookmarks will be appended to the **Bookmark** menu as a new menu item. Select the desired bookmark from the menu and jump to the corresponding topic. Edit or delete bookmarks by selecting the **Manage** option from the **Bookmark** menu which will open the following window:



Bookmarks not only remember the document you were on, but the zoom factor, scroll position and hotspot highlighting as well.

"System" bookmarks (green flag) can be built into the project file, and be seen by all users.

2.2.9.1 System bookmarks

As of DBDOC 11.6, a new type of bookmark has been introduced. These "System" bookmarks are built directly into the DBDOC project file, and are thus distributed to all users sharing that file. The System bookmarks can be created by the DBDOC administrator as part of the build process.

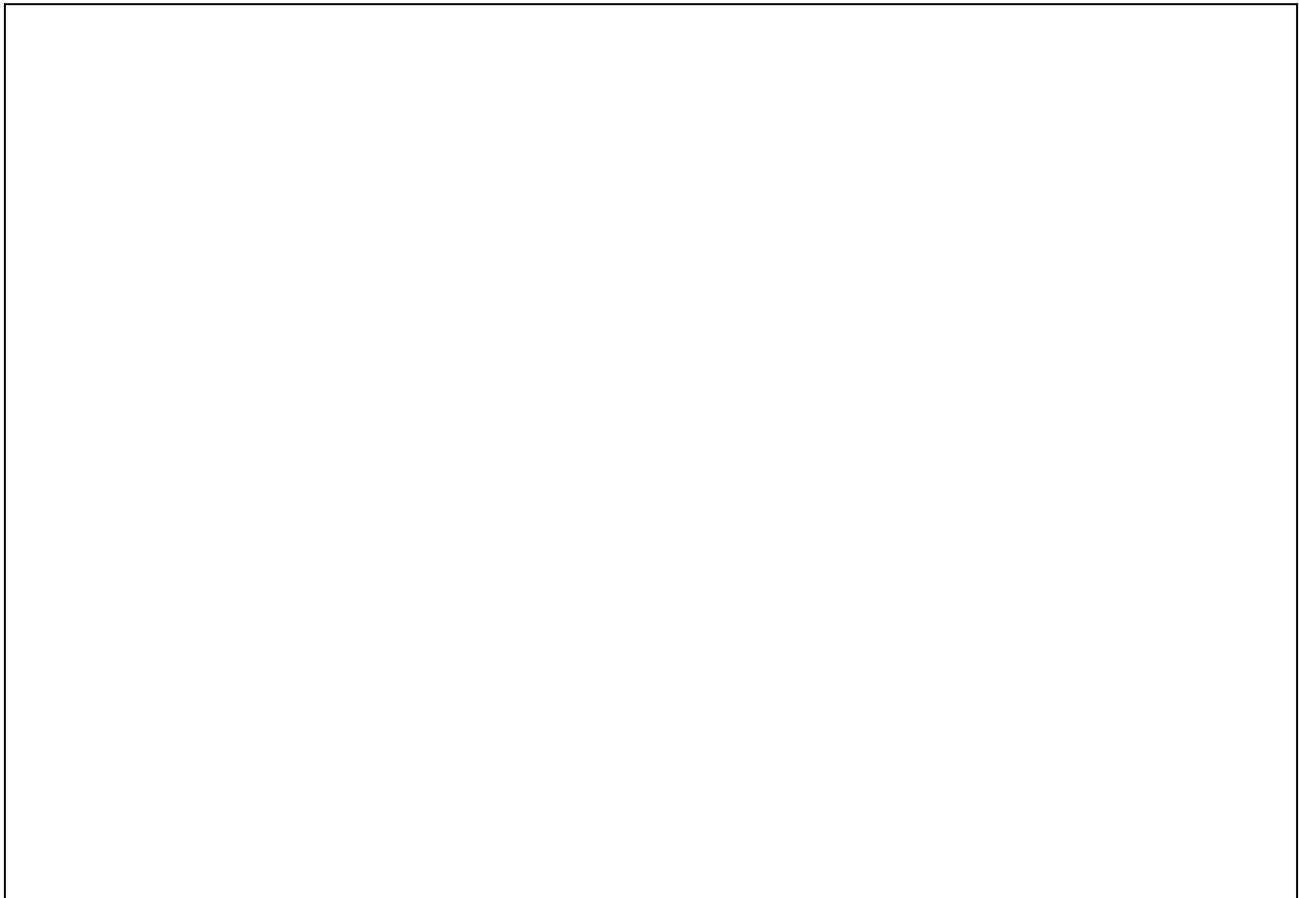
2.3 Techniques for tracing inputs and outputs throughout your system

2.3.1 Overview of signal tracing in Hyperview

There are two basic ways to trace signals in Hyperview:

1. You can click on hotspots in in the content pane in order to display uses of the hotspot value in the index pane. Clicking on a use in the index pane causes the document with the use in it to be displayed in the content pane. This method of signal tracing requires that the index pane be visible, so it cannot be used when the browser is in "full screen" mode. On the other hand, the list of uses of a value is displayed in the index pane until you change it, and this list can be printed and perused at leisure.
2. You can double-click or use the context menu on hotspots to jump directly to the source of a value (bypassing the index pane), or to display a list of uses in the context menu that you can select from. This method of signal tracing can be used even when the browser is in "full screen" mode with the index pane not visible. The downside is that the index of uses of a value is only temporarily visible.

2.3.2 Tracing signals using hotspots and the index pane



2.3.2.1 Tracing back to the source of a value

To trace a signal, use the follow steps and refer to the image below:

1. Click on an item of interest in the right side content pane
2. Click on item labeled source, in red in the index pane, if it exists, then click on an input to the block newly highlighted in the Content pane
3. Repeat until you have traced back far enough.

Viewing live data on the fly:

- At any point, you can turn on live monitoring to check what the values are.
- If you use **Slow Mode** , monitoring will stay on from document to document, and values will be fetched whenever you change documents.
- In **Continuous Mode** monitoring, monitoring is automatically turned off when you go to another document.

2.3.2.2 Finding and displaying the uses of a value

Simply click on any tag or block in either the content pane or the index pane. Its source and uses will be displayed in the index pane. The source is always displayed in red. By default all uses are displayed in blue.

2.3.3 Tracing signals using Go To Use, Go To Source, and double-clicking on hotspots

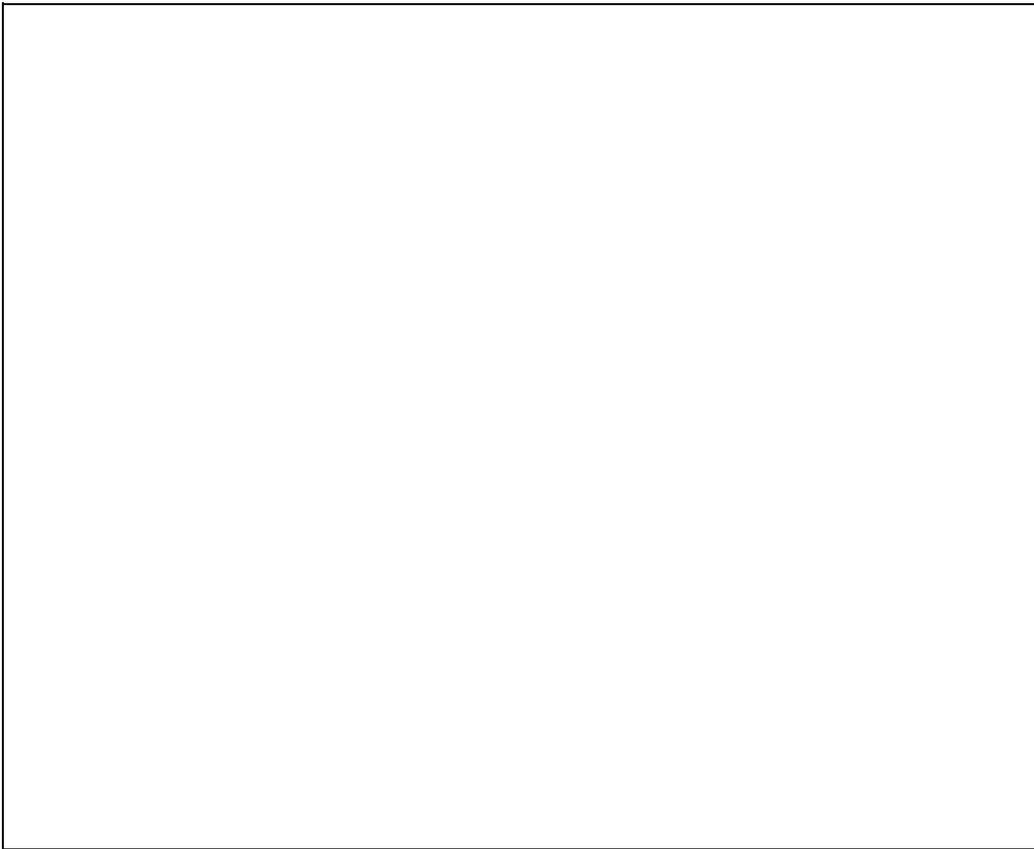
2.3.3.1 Jumping directly to the source of a value with Go To Source or double-click

The **Go To Source** feature is a powerful and natural feature that allows you to find the source of an value without using the index pane at all.

There are two ways to use Go To Source:

- Double-click on an input on a CAD/CLD or block on a graphic
- **OR** Right-click on a point and choose **Go To Source** on the popup menu

Hyperview will automatically take you to the source of the input.



2.3.3.2 Jumping directly to a use of a value with "Go To Use" or double-click

To access any use of a block, right-click on the block's output or input and point to **Go to Use**, then click on one of the block's uses. If you have recently navigated to the block using **Go To Source**, the block from which you used **Go To Source** will be listed first.



You can also double-click on an output to go to a use. This will take you to a use of the block as follows:

1. If there is only one use, or only one high-importance use, you will be taken to it automatically.
2. If you have recently navigated to the block using **Go To Source**, you will be taken to the block you used **Go To Source** from.

3. Otherwise, a popup menu will appear to let you select a use.

2.4 Full text, database, or title searches and search result navigation, and finding coordinates

2.4.1 Searching

When you build an project file with DBDOC, Hyperview can search it to find almost anything. There are three search options that you can use to find the information that you need:

Full Text Search

Database Search

Topic Title Search

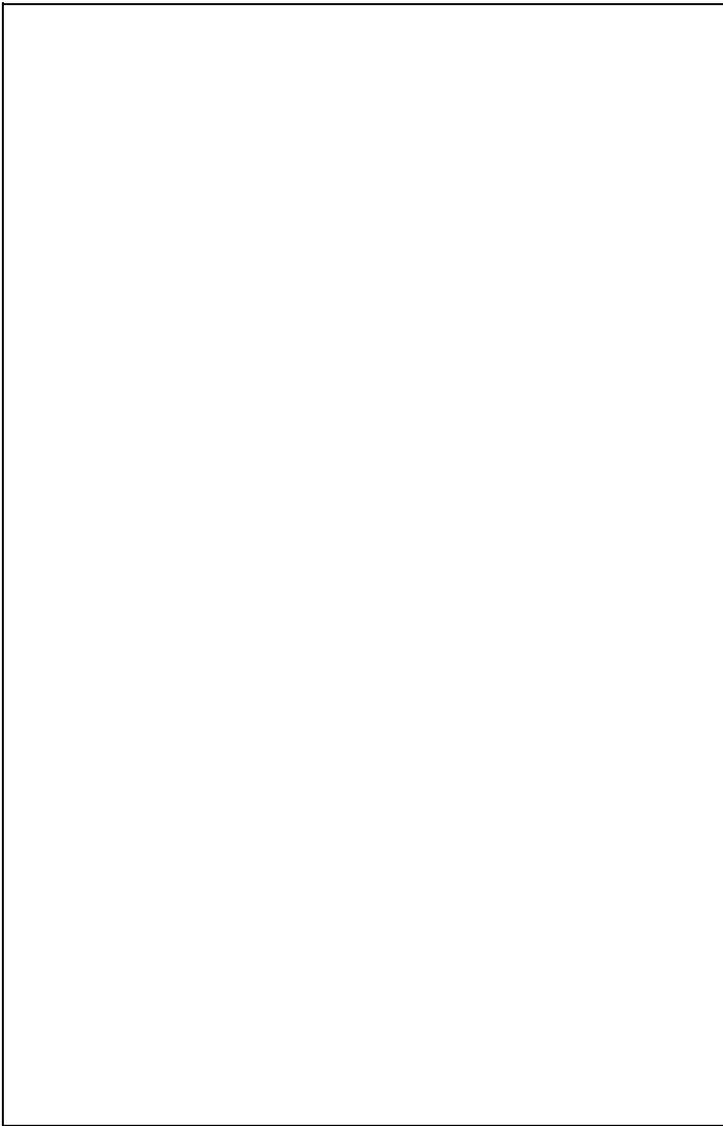
Just look for the **Big Binoculars** and move your mouse over them. The button will change to reflect the last type of search that you did. Click on the button and the **Search** dialog will open to that particular type of search.

2.4.2 Text and database searches

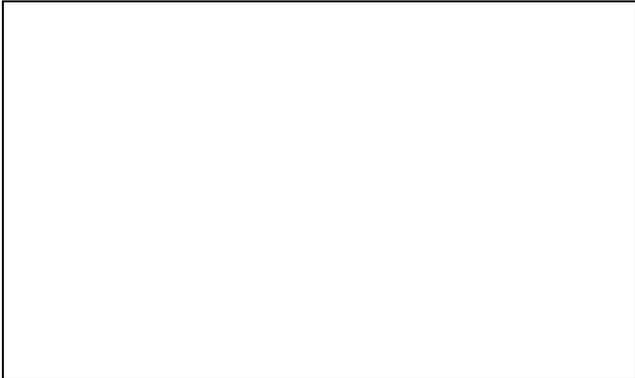
2.4.2.1 Using Full Text Search to search your system for a word or phrase

Search for any word or phrase throughout the entire project, or restrict it to certain types of documents. To launch the Full Text Search utility, you can:

- Click the **Search** toolbar button , then click the **Full Text Search** tab
- **OR** Select **Full Text Search** from the Search menu
- **OR** Press **Ctrl+F** on the keyboard
- **OR** Select text in a document and right-click. Go to the **Search For Selection** option from the popup menu that appears and choose **Full Text Search**.



Multiple word search terms are automatically treated as if they are within quotes. For example, if you look for the term *condenser pump*, you will only find documents with the word **condenser** followed by the word **pump**. Documents that contain only one of the two words **condenser** or **pump** and documents where the word **condenser** is not immediately followed by the word **pump** will not show up.



Match type drop-down list:

- **Match exact phrase:** searches for "condenser pump"
- **Match each term + exactly:** searches for **condenser AND pump**
- **Match each term + to the beginning of a word:** searches for **condenser* AND pump***
- **Match each term + to the end of a word:** searches for ***condenser AND *pump**
- **Match each term + to any part of a word:** searches for ***condenser* AND *pump***

Note: The more wildcarding there is, the slower the search will be.

In **Match each term** mode, you can specify whether you wish search for **ALL** the terms, or **ANY** of the terms by selecting either

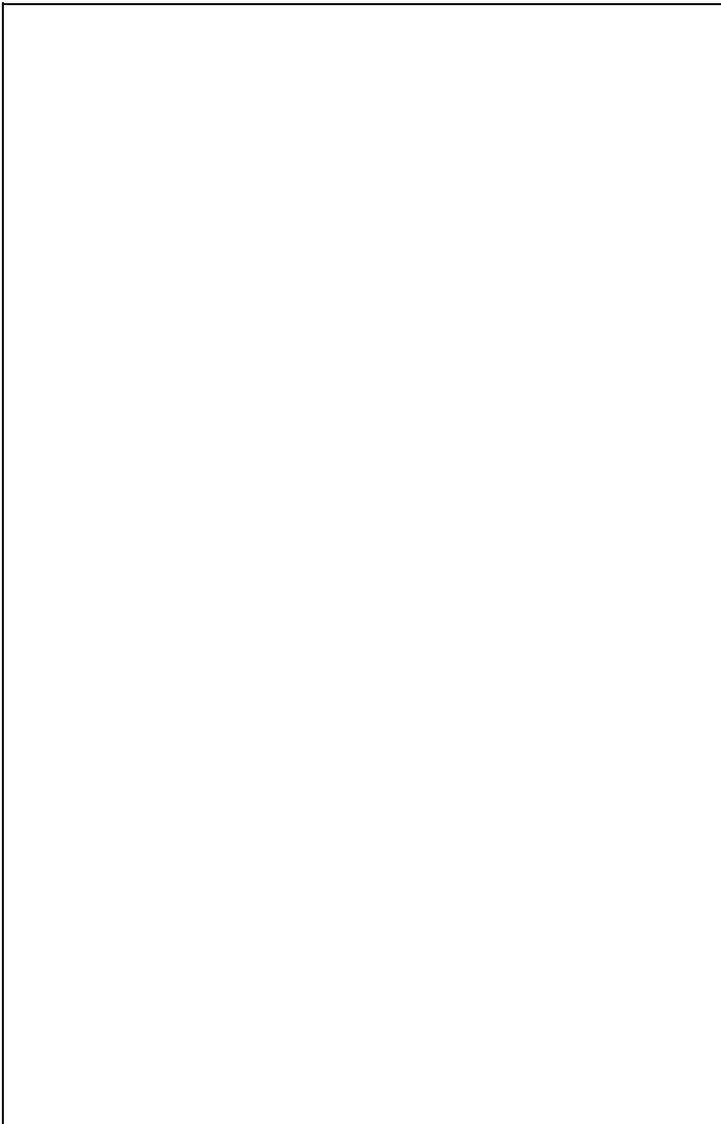
- **Find any search term**
- **OR Find all search terms.**

Note: The logical operators **AND**, **OR**, and **NOT** and parentheses can be used in **Match each search term** mode. In **Match exact phrase** mode they will just be treated as ordinary text to match.

2.4.2.2 Using Database Search to find particular blocks or tags

A database search allows you to search the databases built into your project. To launch the **Database Search** utility you can:

- Click or on the toolbar and select the **Database Search** tab
- **OR** Select **Database Search** from the **Search** menu
- **OR** Press **Alt+D** on the keyboard
- **OR** Select text in a document and right-click. Go to the **Search for Selection** option from the popup menu that appears and choose **Database Search**.



You can search by entering the tag index value, trend index value, tag name (or partial tag name) or any combination of Loop, PCU, Module and Block number. You can select any previously entered tag names or miscellaneous text from the drop down list. Miscellaneous text is very useful if you know part of the description field of what you are looking for but none of the above information.

2.4.2.3 Setting search options for text and database searches

For both **Full Text Search** and **Database Search**:

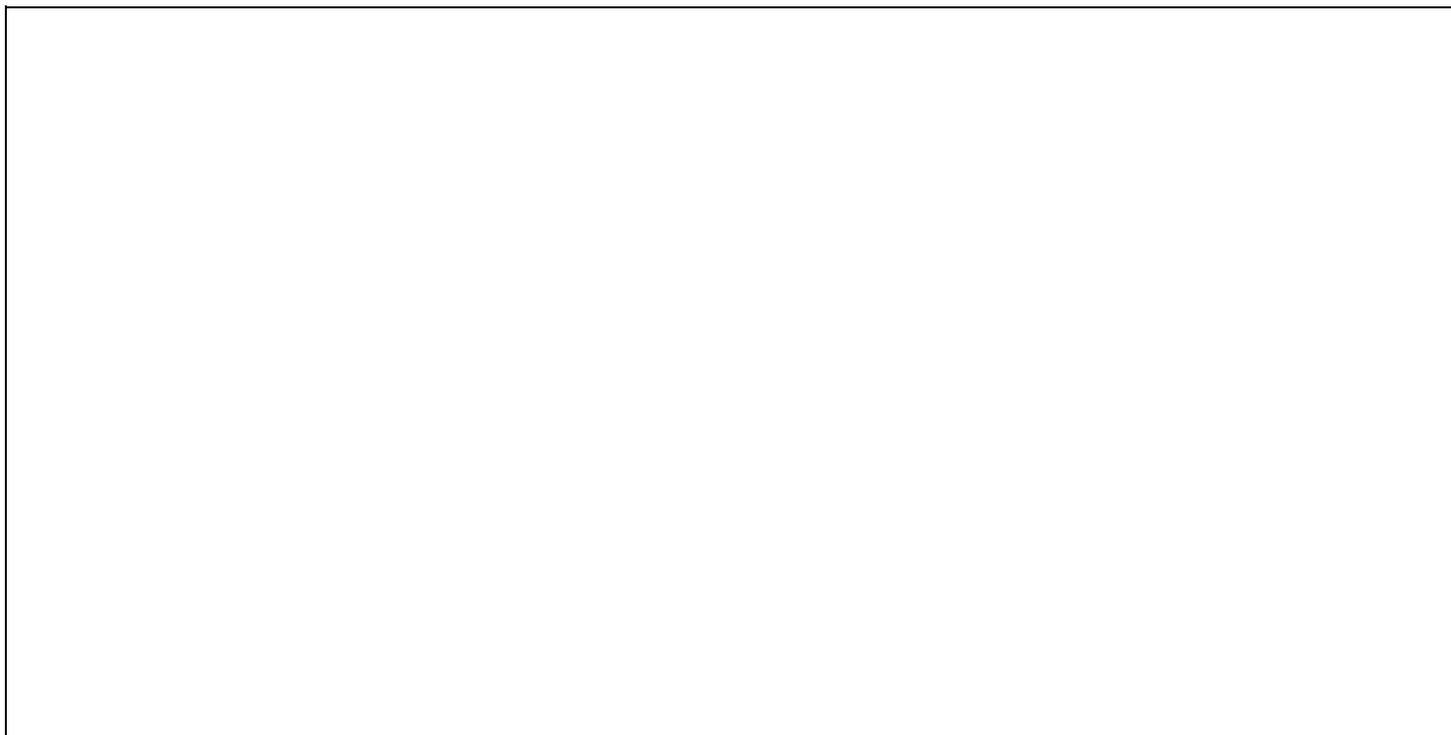
- **Show Loop, PCU, Module, Block columns** shows these attributes of a result topic, where applicable.
- **Show Number of Hits per Topic** shows you the number of times that your search item has shown up in one document.
- **Show Group** shows you the category to which your hit document belongs.
- **Clear Search Hits** on the **Search** menu will remove all the highlight marks from the search results.

For **Full Text Search** only:

- **Show Hit Strings** shows the context in which a hit occurs (i.e. surrounding text)

2.4.2.4 Viewing search results for text and database searches

After you do a text or database search, you will see the following search results dialog:



- **File:** Gives you the option to **Save** the search results to a file (a default filename is automatically supplied.)
- **Topics Containing Search Key:** Lists all the topics that met the search criteria. The list can be sorted based on Topic Title, group or hit frequency. You can jump to a topic by double-clicking on it.
- **Keep Open:** Keep the dialog box open after going to a document rather than automatically closing.
- **Search Again:** Click to bring up the full text search dialog with all your previously specified search criteria filled in to narrow down your search hits.
- **Go To:** Go to the currently selected topic in the list.

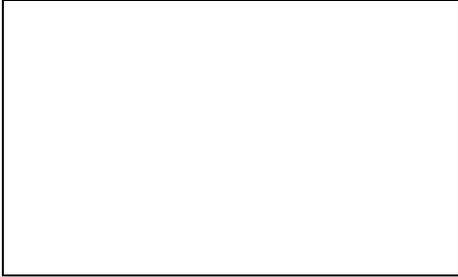
2.4.2.5 Displaying and stepping through search results using the Hyperview toolbar

There are a number of ways you can browse search results later, even after the **Search Results** dialog has been closed.

- Use the **Next Highlight** and **Previous Highlight** toolbar buttons to step through the search results in the document you are looking at.
- Click the **Search Results** toolbar button to bring up the list of search results

- Use the **Next Search Hit** and **Previous Search Hit** toolbar buttons to step through the documents that were found.

If no search hits were found, Hyperview will bring up an informational message box to let you know.

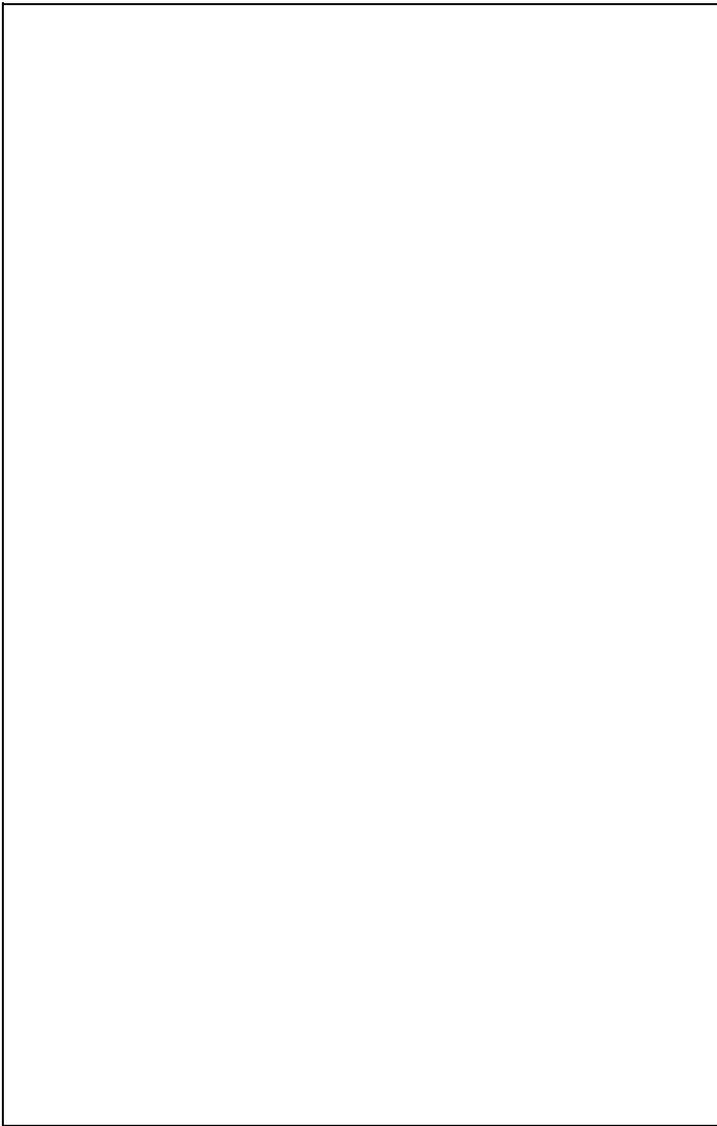


You can again choose to suppress this dialog from appearing the next time no matches are found by checking the **Don't show me this again** option before dismissing the option.

Note: If you click the box, when you search for something that is not found, you will simply see the search box remain as you launched it when the hourglass disappears. You can then easily try the search in a different way, but you have to realize that the search was done, but was unsuccessful.

2.4.3 Using the Topic Title Search to jump directly to a topic

This highly convenient search mode makes it very quick and easy to jump to a topic of interest. Simply start typing the topic name. You will see a list of possible topics as you type. The more you type, the shorter the list will get. Double-click on one of the results any time to display it in the browser.



To launch the **Topic Title Search** utility, you can:

- Click or on the toolbar and select **Topic Title Search**.
- **OR** Select **Topic Title Search** from the **Search** menu.
- **OR** Press **Alt+T** on the keyboard.
- **OR** Select text in any document and right-click. Go to the **Search For Selection** option from the popup menu that appears and choose **Topic Title Search**.

Other useful tips by searching for title:

As you enter the document name, Hyperview narrows the scope of the match. **Module LL,PP,MM Block NNNN** is a topic. **NNNN** will give you Conductor NT graphics by number. Select the document you want and click the **Go To** button, press return, or double-click on the selected document.

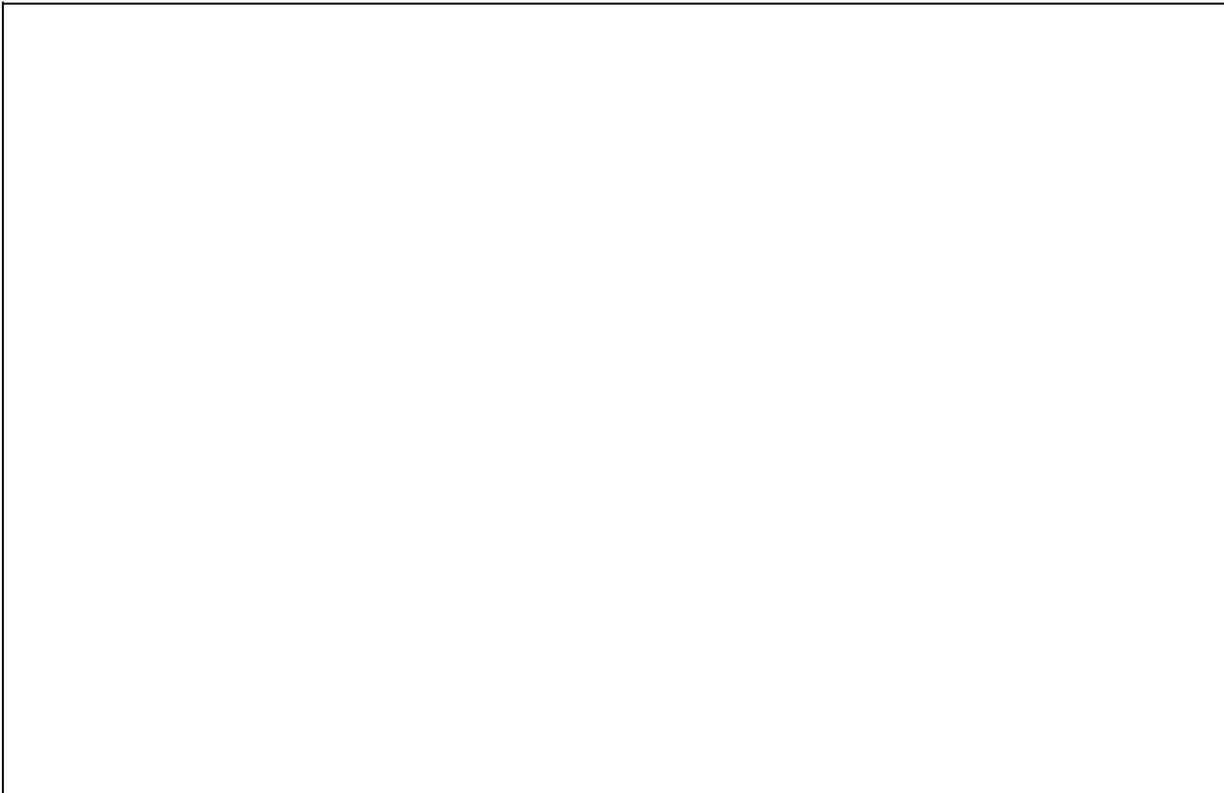
- **Documents Only:** Check this option if you would only like to search document titles (this excludes tag names, ref names, AC 800M function block names, etc. from the search)

- **Keep Open**: Check this option to keep the dialog box open after going to a document rather than having it automatically close.

2.4.4 Advanced: Finding a Coordinate

You can also enter a coordinate set to go to on a specific document - a helpful feature for finding messages reported in the DBDOC.ERR files.

Spot the blinking red ball! Find the coordinates you are interested in on a given document easily without having to scroll through it. Hyperview will go directly to the coordinates entered and zoom in.



Three ways to bring up the coordinate locator:

- Select **Find Coordinate** under the **Go** menu option - enter the coordinate set that you want to go to on the current document.
- **OR** Right-click on the drawing to bring up the popup menu and select **Find Coordinate**.
- **OR** use the keyboard shortcut Ctrl+G when the content pane is in focus.

When launched via the **Go** menu or the popup menu, the coordinate locator dialog displays the minimum and maximum x and y values on the current drawing. The minimum values or the last entered coordinate set are automatically filled into the edit boxes.

A message will appear when you enter a value that is out of range. Negative values are accepted where applicable.

Cancel the coordinate locator by double-clicking on it.



Check the **Keep Open** option to keep the dialog open after the **Go To** button has been clicked. This is useful if you are iterating to pinpoint an exact coordinate set.

2.5 Guide to Live Data on Documents

2.5.1 Live Data Monitoring

Hyperview provides read-only access to limited amounts of live data. Hyperview will display the data for individual blocks as well as displaying analogue values in bar graphs where called for on graphics.

Note: As of DBDOC 11.6, Hyperview can provide live text data for FC 151 text selector blocks.

There are three ways to fetch live data from your system:

- **Slow Mode (Camera)** - preferred in general
- **Continuous Mode (Clock)**
- **Live Loop Annotations** - live data for individual points

2.5.1.1 Using Slow Mode live data

To take a snapshot of the live data for all the outputs in the viewing area, click on the Slow Mode button on the toolbar. Hyperview will fetch the values of the outputs displayed. Click the camera button again to fetch the live data values on the display again.

- Slow Mode will remain on until it is turned off by double-clicking the Slow Mode Button.
- When you view a different document, the values for it will be fetched automatically.
- If you scroll or zoom on the current document, Hyperview will automatically refresh the values.
- If you do not scroll, zoom, or change documents, Hyperview will refresh the values every ten seconds.

The shortcut keys used for Slow Mode are listed below:

- **M** - Turns on Slow Mode and will fetch all values on the CAD or CLD sheet you are viewing. Pressing **M** successively will refetch the data on the sheet you are viewing.
- **Shift+M** - Turns off Slow Mode.
- **F5** - Refetches the values on the CAD or CLD sheet you are viewing.

2.5.1.2 Getting live data in Continuous mode

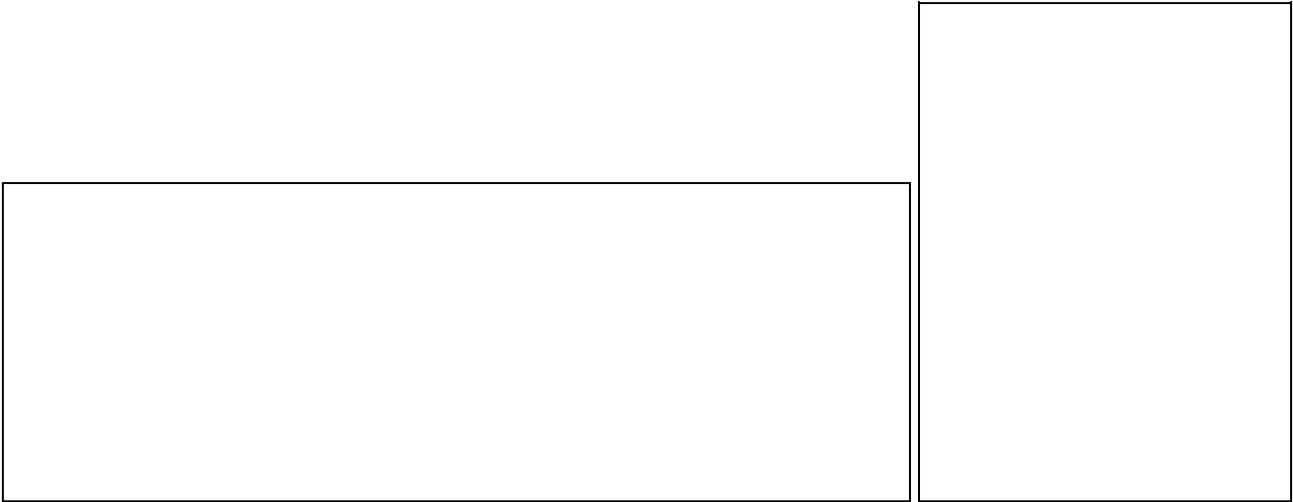
You can activate **Continuous mode** live monitoring of your system by clicking on the **Live Loop Display** button or selecting the option from under the **View** menu.

- Hyperview will send out requests at the desired frequency (by default, every second) for data for all visible points in the drawing pane.
- The update interval can be changed in the **Options** dialog.

Note: On dynamic bar graphs: The color of the bar displayed will indicate the status of the data. The colors of the status bar are as follows:

- **Valid:** bar will be green.
- **Stale or unavailable:** the bar will be drawn in red crosshatch.
- **Bad quality:** bar shown in the user specified bad quality color.

Note: If you attempt to monitor without having any live loop settings entered, Hyperview will automatically alert you and give you the opportunity to enter settings at that point. This helps to take the guesswork out of why you would not be seeing live data!



2.5.1.3 Managing the Slow Mode Data Load

In **Slow Mode**, data is fetched automatically every ten seconds, and when you move the screen, zoom, or go to another document, or explicitly refresh the values, providing an excellent balance between having the most relevant up-to-date data, and not overloading the data bandwidth available. For most purposes, **Slow Mode** will be the best option for minimizing data load due to on-document live data.

Slow Mode tips:

- Double-click the camera button to turn off **Slow Mode**. **Slow Mode** is never turned off automatically.
- On graphics, zoom in to the area you are interested in. Data is only fetched for visible points. If you do not zoom in, it may take Hyperview several seconds to fetch data for the entire graphic.
- Switch to CAD / CLD sheets sooner rather than later. This will further reduce the amount of data fetched.

If you use Slow Mode, you will leave a lot more bandwidth for other Hyperview users.

2.5.2 Live Loop Annotations

Live Loop Annotations are "sticky notes" displaying live data. You can add them to any document. You can create a **Live Loop Annotation** to display live data for any block in your system, or module status for any module. **Live Loop Annotations** are a convenient way to monitor block values, or module status for one part of your system while looking at a different part of the system in Hyperview, or to display live data for just a

few items on a document.

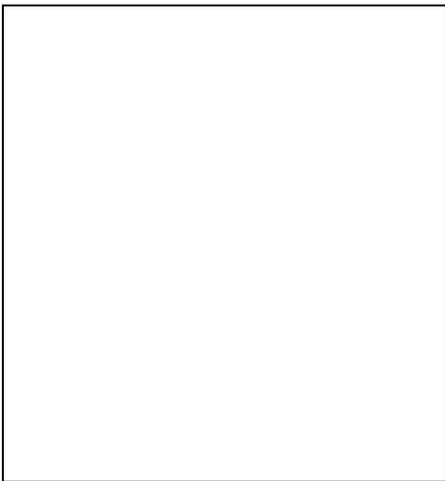
You can place **Live Loop Annotations** on any document that you can view in Hyperview's content pane, including graphics, text files, and database entries. If you want to monitor live data independently of what document you happen to be viewing, consider using the **Watch Window** instead.

Live Loop Annotations are automatically saved, and will reappear on the documents where you created them the next time you open Hyperview, even if you switch to a new build of your system. So if you always need to refer to a certain block value while viewing a particular document in Hyperview, **Live Loop Annotations** will let you do just that.

Standard live loop annotation:



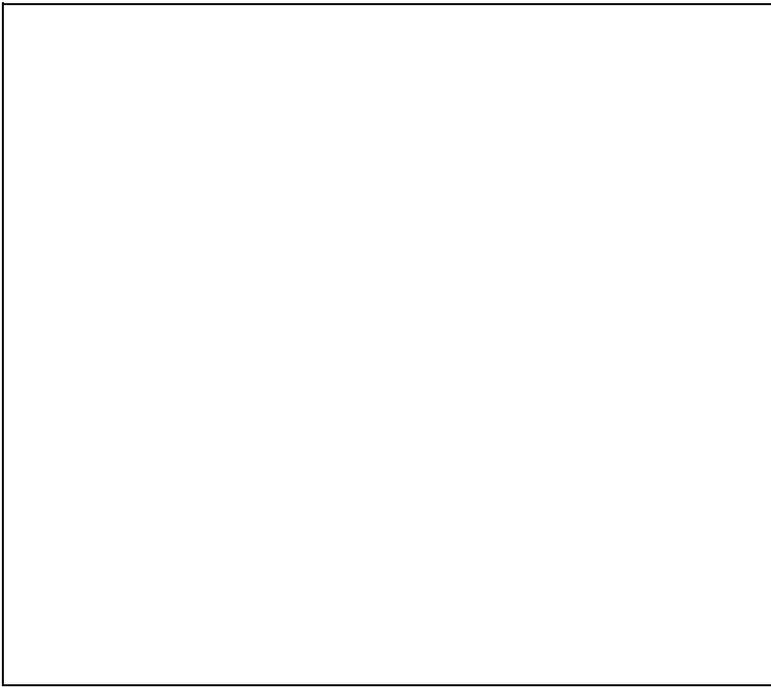
Live loop annotation with dynamic bar:



Live loop annotation with data specs:

Note: As of DBDOC 11.1 this type of annotation has been replaced by **Live Specs**.

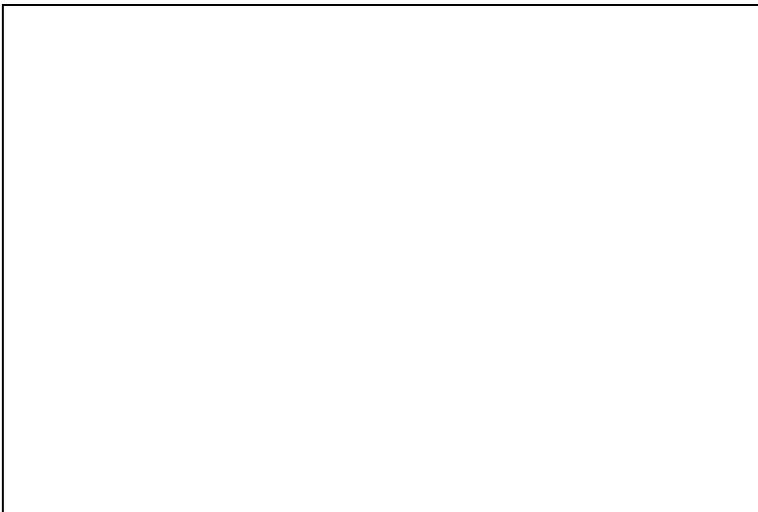
Live loop annotation with module status:



Use live loop annotations to:

- View live loop data for just one block in your system.
- Monitor live loop data from any Hyperview document.
- View the live module status for any module in your system.

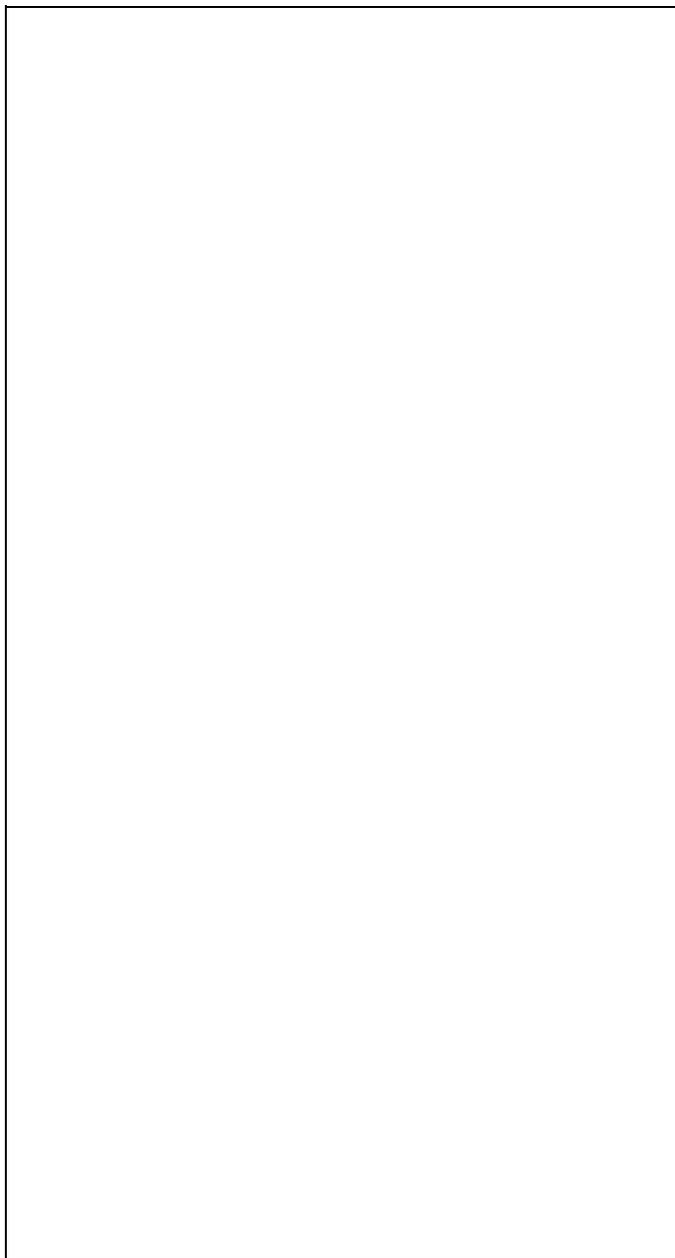
For example, you can easily use **Live Loop Annotations** to keep track of the value of a tag, while viewing that tag's database entry.



To view help on **Live Loop Annotations**, right-click any **Live Loop Annotation**, then click **Help**.

2.5.3 Creating a Live Loop Annotation

The **Live Loop Annotation** dialog is used to specify a value to monitor and to set or change the options for a **Live Loop Annotation**.



The **Live Loop Annotation** dialog is used to specify a value to monitor and to set or change the options for a live loop annotation.

There are three ways to specify the point that you wish to monitor:

- By creating the live loop annotation on a hotspot
- By entering the block number
- By entering a tag name

You can also monitor the status of a module.

To create a **Live Loop Annotation** on a hotspot:

- Click the **Live Loop Annotation** toolbar button  then click the

hotspot. A **Live Loop Annotation** displaying the output value of the block will be created.

OR

Right-click on the hotspot, point to Create Live Loop Annotation for this Block, then click one of:

- **Show Value**
- **Show Dynamic Bar**
- **Live Loop Annotation** dialog

Choosing the **Live Loop Annotation** dialog will allow you to change settings such as the update interval and the dynamic bar range. The other three options will create a **Live Loop Annotation** with the default settings.

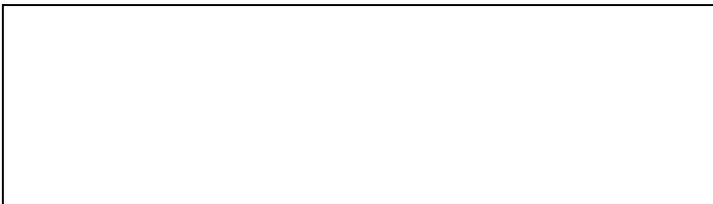
You can create a live loop annotation on any hotspot with LPMB information, including on text documents and batch files.

To create a live loop annotation that is not on a hotspot:

- Click the **Live Loop Annotation** toolbar  button , then click the point on the content pane where you would like to place the live loop annotation.

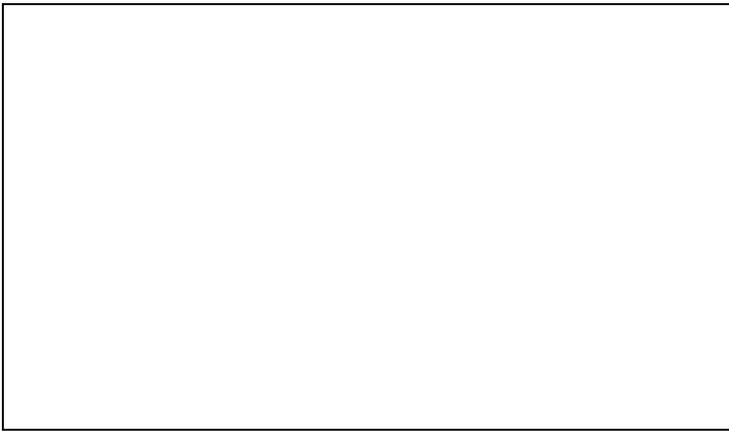
The live loop annotation dialog will appear so that you can specify which point to monitor.

To specify which point to monitor by loop, PCU, module and block number:



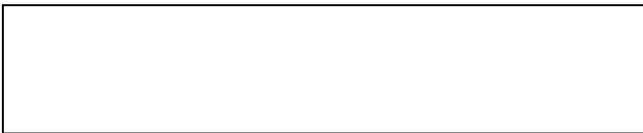
1. Click **Value for Block Number**.
2. Enter the appropriate information in the LPMB boxes.

To specify which point to monitor by tag name:



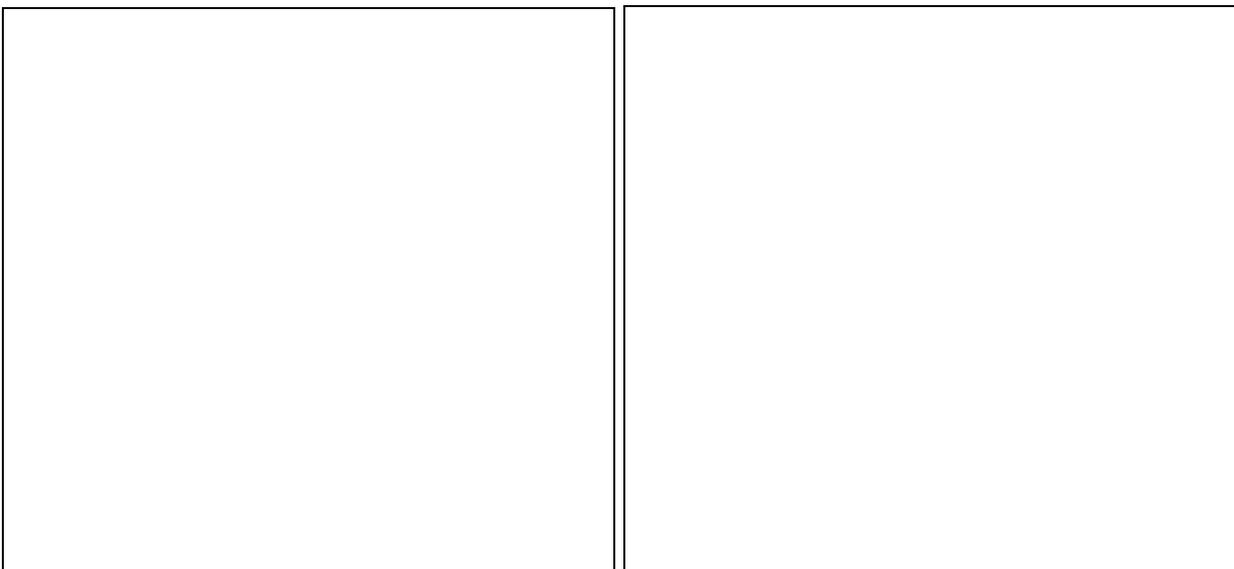
1. Click **Value For Tag Name**.
2. Enter part or all of a tag name in the **Tag Name** box and press the **Search** button. Hyperview will look through the databases and present any matching tags in the list.
 1. You can enter pseudo-tags to look for a particular point (e.g tagname@PV)
 2. If you leave the Tag Name box blank, the search results will contain all tags in all databases in the project.
3. In the search results list, click the tag name which you wish to monitor.

To monitor the status of a module:



1. Click **Module Status**.
2. Enter the appropriate information in the **LPM** boxes.

2.5.3.1 Block Options (for either Live Loop Annotations or Watch Window Blocks)



2.5.3.1.1 Basic Options

- **Description:** This text will be displayed when the mouse is hovering over a live loop annotation.
- **Update Interval:** The amount of time, in seconds, that the live loop annotation will display each value before fetching a new value. The default interval for new annotations is one second. This default can be changed in the **View** tab of the **Options** dialog (see p.58).

2.5.3.1.2 Annotation Options

- **Show Live Data:** If this is selected, the live loop annotation will show the current output value of the block as a number.
- **Dynamic Bar:** If this is selected, the live loop annotation will show the current output value of the block as a bar graph and as a number.

2.5.3.1.3 Watch Window Options

- **Reverse Y-Axis:** Reverse the vertical axis for the plot.
- **Color:** Change the plot line color for the selected block.
- **Schedule:** Set an automatic data collection timeout for the specified block. Blocks can be manually paused and restarted, as before, but starting in version 10.4.1 automatic timeouts can also be set. A block can be configured to automatically stop collecting data after a certain interval, for example 24 hours, or at a specified time.

2.5.3.1.4 Plot or Dynamic Bar Range

- **Built-In Range:** This option is only available if **Dynamic Bar** is selected and the block has minimum and maximum values specified in one of the project databases. If this option is selected, the minimum and maximum values from the database will be used for the limits of the bar graph.
- **Auto Range:** This option is only available if **Dynamic Bar** is selected. If this option is selected, the bar graph limits will adapt to the data fetched.
- **Specify Range:** This option is only available if **Dynamic Bar** is selected.. If this option is selected, the bar graph will have the limits you specify here.

2.5.3.2 Reading Live Loop Annotations

You may see the following symbols and abbreviations on live loop annotations:

<input type="checkbox"/>	Successful data fetch
<input type="checkbox"/>	Error
<input type="checkbox"/>	Bad quality
<input type="checkbox"/>	Alarm
<input type="checkbox"/>	Info - the version of CIUMon that you are currently using to monitor with has been configured for a certain loop and PCU and the point that you are requesting data for does not match that

	configuration.
<input type="checkbox"/>	The IP address of the computer you are currently monitoring from has not been approved by CIUMon to fetch live data. You will also see IP_BLOCKED displayed in the manual live loop data window.
H	High
L	Low
D	Deviation
C	Calibration
S	Service

If the symbols on an LLA are greyish, this indicates that the data is stale.

2.5.3.3 Managing Live Loop Annotations

You can view a complete listing of all live loop annotations by:

- Clicking the **Manage Annotations** button on the toolbar
- **OR** Selecting **Manage Annotation** from under the **View** menu and clicking the **Live Loop Annotations** tab.



For a selected annotation, you can:

- Change what value it is monitoring.
- Change any of the options, including **Mouse-over text**, **Update Interval**, and **Dynamic Bar** settings.

2.5.3.4 Moving Live Loop Annotations

To temporarily move a live loop annotation, click and drag it to a new position. It will be linked to its permanent position by a thin line, or "rubberband". If the annotation was created on a hotspot, the rubberband is useful as a pointer to that hotspot.

To move an annotation back to its permanent position, double-click it.

To minimize a live loop annotation, click the **Minimize** button on its upper right hand corner.

To permanently move a live loop annotation, minimize it, then click and drag the minimized annotation to a new position.

2.5.3.5 Deleting Live Loop Annotations

To delete a live loop annotation from the content pane:

- Click the **Delete** button  on the live loop annotation's upper right hand corner.
- **OR** Right-click the live loop annotation, then click **Delete Live Loop Annotation**.

To delete live loop annotations using the annotation manager:

- To delete a single live loop annotation, select it in the list of annotations, then click **Delete**.
- To delete all live loop annotations on a topic, select the topic in the list of annotations, then click **Delete**.
- To delete all live loop annotations in the entire project, select the top entry in the list of annotations, then click **Delete**.

2.5.4 Live Specs

Live specs can be turned on in the **View** menu by selecting **Enable Live Specs**. On layered specs, live spec values will then be displayed directly. Prior to DBDOC 11.1, these live specs would have to be displayed as their own type of live loop annotation.

Just as with refreshing live data values in Slow Mode, clicking the camera in the bottom right of the specs will refresh the values. Unlike in Slow Mode, these specs do not update automatically over time.

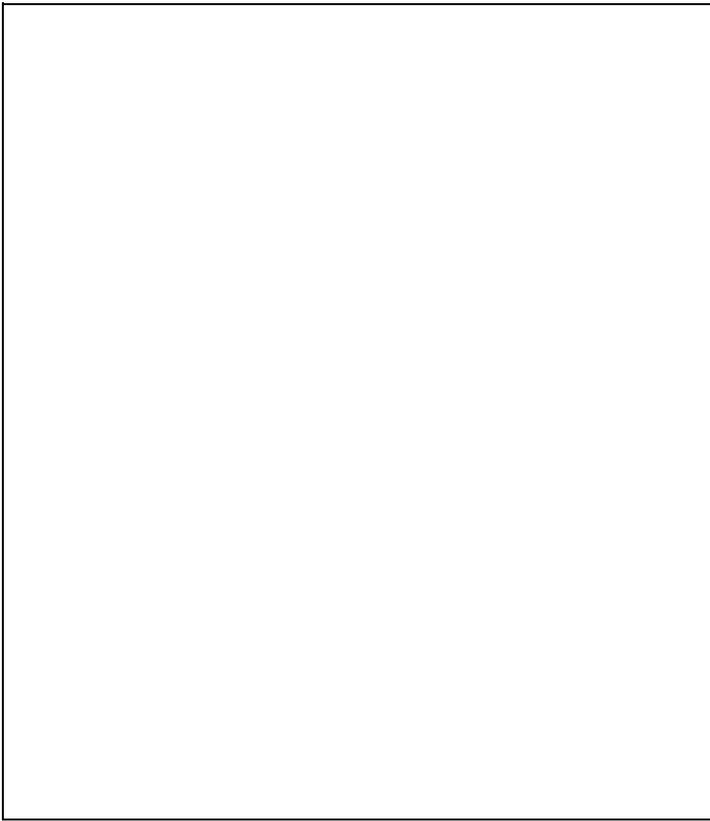
Live specs appear in two places: in the layered spec rectangle, and on the inputs.

1. General specs

- ◆ Appear in a large movable rectangle.
- ◆ Black text means the spec's live value is same as its static value.
- ◆ Green text means the spec's live value is different from its static value (probably not an error, though).

2. Input specs

- ◆ Appear as small rectangles around each of a block's inputs.
- ◆ A green rectangle means the spec's live value (i.e. the block number of the input) is the same as its static value.
- ◆ A red rectangle means the spec's live value (i.e. the block number of the input) is different from that implied by the wiring. This could well be unexpected, and so it is highlighted.



2.6 Trending Live Data in the Watch Window

2.6.1 Watch Window Overview

The Watch Window allows you to plot live data from one or more points in the same window. It helps troubleshoot problems and analyze the way the system works. Add blocks from anywhere in the system to a window that displays their values in an easy to use plot. Control the range, resolution and data collection rate of each block.



2.6.1.1 Using the Watch Window: quick summary

- Adding a block to the Watch Window can be done effortlessly by right-clicking on a block and selecting **Add this Block to the Watch Window**.
- Once a block has been added to the Watch Window, it will begin collecting data. As you move the mouse over the plot, crosshairs appear showing you exact data values.
- Click on the plot to place a **Timestamp** showing the value and time of a point of interest.
- Blocks are added in **Split Plot** mode with **AutoScaling** enabled. This gives you a natural starting point. One click and you display them in one plot. One click and you rescale the plot of the value.
- You can easily walk through Timestamps by using the **Back** and **Forward Navigation** buttons at the bottom of the Watch Window.
- Using the Block List, you can hide the plot for a block by unchecking the checkbox on the left-hand side. Autoscaling and pausing data collection for a block can also be enabled using the Block List.
- Using our simple **Export Data** tool, you can export the data for the group for a selected group into a .csv file for import into Microsoft Excel® or other external applications.

2.6.2 Opening the Watch Window

You can open the watch window from the Hyperview window in three ways:

- Click the **Live Data Watch Window** button  in the toolbar.
- Press **W** on your keyboard.
- Click **View | Watch Window** in the main menu.

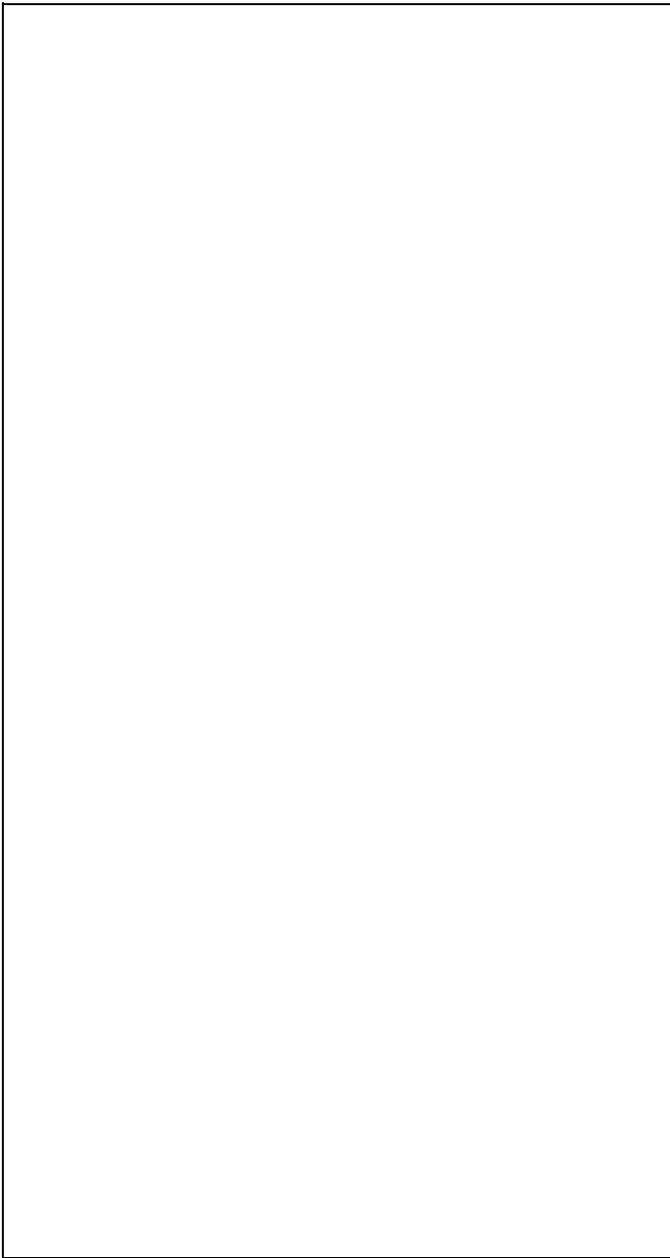
2.6.3 Adding blocks to the Watch Window so they can be plotted

There are several ways to add a block to a group in the watch window:

1. On any CAD, CLD, graphic, AutoCAD or Microstation drawing, right click on any source or output hotspot and select **Add this Block to the Watch Window**. The point will be added with the default options to the watch window group you choose.
2. Right click on any live loop annotation and select **Add to Watch Window**. The point will be added to the watch window group you choose using any live loop annotations options you already entered.
3. In the **Live Data Watch Window**, click the **Add Block** button and define the block you wish to add.
4. In the **Live Data Watch Window** menu, choose **Block > Add** and define the block you wish to add.
5. Right-click on any block in the block list and choose the **Add Block** menu option.
6. Right-click on a blank area of the block list and choose **Add Block** from the menu.

2.6.4 Defining and editing block plots in the Watch Window

If you are specifying a block to add to the Watch Window by block number or tagname, or editing options for an existing block in the Watch Window, you will use the **Block Live Data Options** dialog.



To edit the options of an existing block with a plot in the Watch Window:

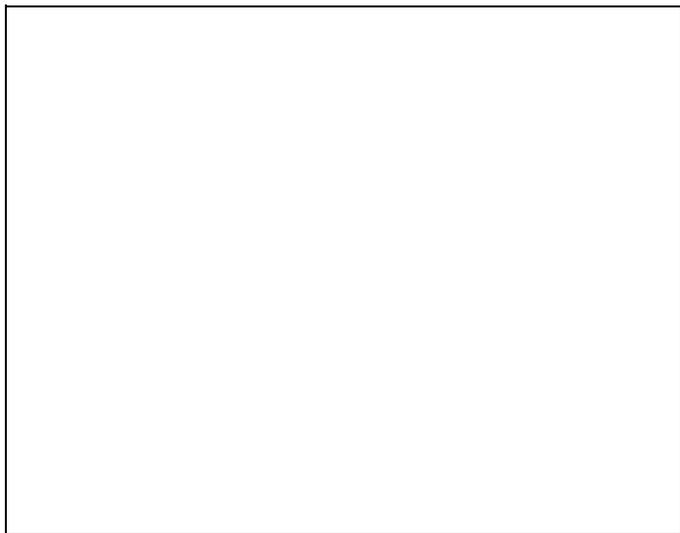
- Double-click the block in the Block List on the right hand side of the Watch Window..
- Right-click the block in the Block List and choose **Edit**.
- Select the block in the Block List and click the **Edit Block** button.
- Select the block in the Block List and choose **Block | Edit** from the menu.

The **Edit** dialog that appears is the same as that which is discussed in the section of the **User Guide on Live Loop Annotations**.

2.6.4.1 Specific Watch Window Options

The following block options are specific to Watch Window blocks.

- **Reverse Y-Axis:** Reverse the vertical axis for the plot.
- **Color:** Change the plot line color for the selected block.
- **Schedule:** Set an automatic data collection timeout for the specified block. Blocks can be manually paused and restarted, as before, but starting in version 10.4.1 automatic timeouts can also be set. A block can be configured to automatically stop collecting data after a certain interval, for example 24 hours, or at a specified time. If you like, by checking **Confirm before pausing** you can require that Watch Window obtain confirmation before automatically pausing data collection for any block.



2.6.5 Creating groups of blocks in the Watch Window

Create different groups of points to keep your watch window uncluttered. Make sure to specify a unique name for each group that you create. Create a new group by clicking on the **Add Group** button or choosing **Add** from the **Group** menu in the Watch Window. This will give you an empty group to which you can start adding the blocks that you want to monitor.

2.6.5.1 Importing a list of blocks from CIULink

If you are monitoring blocks using **CIULink**, you may want to transfer them to the Watch Window.

Import a taglist created in CIULink as a group. Choose the **Import** option from the **Group** menu or click the **Import Group** button.

You have the choice of browsing for an exported .LST file created by CIULink or choosing from the list of CIULink taglists that are currently stored in the registry.

The group will be populated with all the points in the chosen taglist - saving you from having to enter each point individually.

2.6.5.2 Managing groups of blocks

Manage your groups:

- Through the **Group** menu in the watch window.
- With the various Group buttons available on the Watch Window toolbar.
- By right-clicking the on the group tab name and selecting the appropriate action from the menu.

2.6.6 Watch Window Highlights

- **Crosshairs:** As you mouse over on the plots, you will see red crosshairs appear for each block that you are monitoring. The LPMB information and the data value will appear next to the crosshair making for easy interpretation of the plotted value. To leave a timestamp so are able to return to a point of interest just click the mouse. These stamps will persist as the Watch Window collects new data. They can be deleted by clicking the **X** in the top right-hand corner. In addition, you are able to move through the crosshairs by using the navigation button at the bottom of the Watch Window.
- **Navigation Plot:** A second timestrip-type plot below the main plot displays all the data that has been collected for the group. The beginnings and ends of data runs are indicated with green and red flags, and you can jump from flag to flag by using the **Data Run Marker Navigation** buttons below the navigation plot.
- **Freeze Scroll:** By clicking on the **Freeze Scroll** button located at the bottom left of the watch window or dragging the horizontal scroll bar, the plot will freeze, allowing you to look at the any portion of the data without having it continuing to scroll to the left. The background of the plot will change from white to a light gray to indicate that the scrolling is frozen. To unfreeze the scroll, click the **Freeze Scroll** button and the plot will jump to the current time.
- **Plot Time Scale:** You can adjust the time scale against which the plot is drawn; by default, the value is 1.00 sec/pixel. You can expand the plot by entering in a smaller value or compress it by entering a larger number in the Plot Time-Scale field. Enter the desired value in the field and click the **Apply** button or just press **Enter**.
- **Split Plot:** Split plot allows each block to be placed onto its own separate plot to be graphed separately. If you look on the left-hand side the range of values being graphed is shown at the top and bottom of the plots. This gives you a better view for analyzing your data.
- **Always keep the Watch Window visible:** Under the **Options** menu in the watch window, check the **Always display Watch Window on Top** option so that the window will float on top of all other windows and will always be visible to you.
- **Exporting Data:** You are able to select which data you wish to export from the Watch Window. By pressing the **Select Mode** button on the toolbar, it will allow you to drag the mouse pointer to select

the data you wish to extract for further analysis. Once you have made your selection, press the **Export Data** button, which will confirm your selection and export the data to a CSV file. The data in the CSV file displays very nicely in **Microsoft Excel**.

- **Go To Source or Go To Block Map:** These options are available by right clicking on a block in the block list. **Go To Source** will take you to the CAD or CLD sheet where the block is used. **Go To Block Map** will take you to that block's entry on the Block Map.

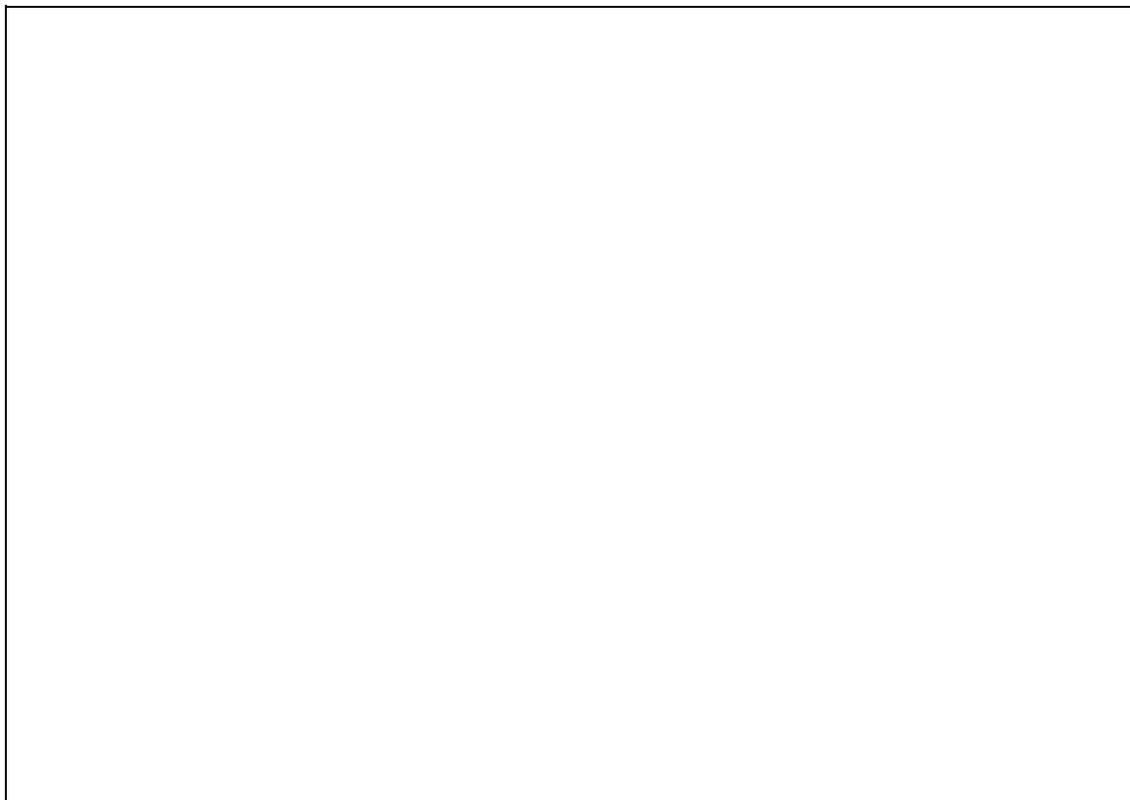
2.6.7 Timestamp Navigation

Timestamp Navigation buttons make the task of navigating through timestamps much easier.

-  Jump to the left-most timestamp on the plot.
-  Jump to the previous (next left) timestamp on your plot.
-  Jump to the currently selected timestamp.
-  Jump to the next (next right) timestamp on your plot.
-  Jump to the right-most timestamp on your plot.

2.6.8 Data Marker Navigation

A navigation plot below the main plot makes it easy to move through different datasets collected at different times.



With the **Data Run Marker Navigation** buttons you can step through data collected at different times.

- Jump to the left-most data run marker.
- Jump to the previous (next left) data run marker.
- Jump to the next (next right) data run marker.
- Jump to the right-most data run marker.

2.6.9 Managing Watch Window related bandwidth, and disk usage

These topics are covered briefly here, and in more detail in the User Guide chapter Guide to Managing Live Data Load and Increasing Live Data Bandwidth.



This **Live Data Status Bar** provides useful information regarding how many data update requests are being made per second, how many updates are received, how much memory is being used for the Watch Window plots, and how much disk space is being used for Watch Window data files.

2.6.9.1 Bandwidth

In general, a CIU is able to provide responses to no more than 30 requests per second. If multiple users are sharing one CIUMon to CIU connection, they will be limited collectively by that data bandwidth.

With a bit of care, you will be able to use the full power of the Watch Window without overwhelming the limited bandwidth available. Here are some tips to keep the Watch Window data fetch rate from being any higher than necessary:

Tips for managing Watch Window data bandwidth:

- Look at **Send/s** and **Recv/s** in the **Live Data Status Bar** at the bottom of the main Hyperview browser to see what data requests Hyperview is sending and receiving.
- If **Send/s** is larger than **Recv/s**, data updates are not "keeping up" with requests. Consider reducing the data load.
- Be aware of the data update requests being generated by the main browser window. Consider turning off live data in the main browser or switching to **Slow** mode .
- Be aware that the Watch Window collects data even when it is not open.

- When you add a block to the Watch Window, it will fetch new data once every 10 seconds. If you don't need to collect data that frequently, increase the update interval.
- If you don't need the data from a block, but wish to keep it defined in the Watch Window, pause data collection for that block.

2.6.9.2 Memory

Prior to DBDOC 10.4, Watch Window plot data was stored in memory. If a very large amount of data had been collected, this could affect performance.

As of DBDOC 10.4, trend data is loaded for display on an as-needed basis. Large datasets will not cause the general memory requirements of Hyperview to increase.

2.6.9.3 Disk Usage

All data collected in the Watch Window is automatically saved to CSV files. Over time, these files can become large.

Tips for managing Watch Window memory use:

- Look at **Data Files** in the **Live Data Status Bar** at the bottom of the main Hyperview browser to see how much disk space is being used for Watch Window data files (including those not currently in the Watch Window).
- Deleting data from inside a plot (i.e. by selecting it and deleting it) in the Watch Window will remove it from memory **and also from the data files**.
- If necessary, you can delete the data files directly. See Guide to Managing Live Data Load and Increasing Live Data Bandwidth for details on their location.

2.6.10 Importing data into other applications

The best way to use Watch Window data in other applications is to export it. It should rarely if ever be useful to use the raw data file directly.

How to export data:

- Select the data in the Watch Window you wish to export. Data for all the visible plots in the visible group will be exported. Hiding a plot will prevent it from being exported.
- Press the **Export Data** toolbar button. You will be able to choose the interval size for the exported data and the file to which the data should be saved.
- You will have the option of viewing the data immediately in the application associated with CSV files on your system, typically Microsoft Excel.

2.7 Viewing and Managing System Errors with the Error Browser

2.7.1 Error Browser Overview

The Error Browser is a powerful tool that allows you to easily examine the errors in your system configuration that DBDOC detected while compiling your project. Because DBDOC integrates information across many sources throughout the system, it is in a position to detect inconsistencies and possible problems, ranging from serious errors to cosmetic issues. For many years, these errors have been available for viewing in compiler generated text files, but as of DBDOC 10.5 they are presented in Hyperview in a dedicated error browser. This enables you to click on errors to jump directly to the associated location in the project file, review errors methodically, and hide errors that you don't need to see. Errors can be filtered, sorted, and grouped in a variety of ways to help you get a handle on them.



2.7.2 Opening the Error Browser

You can open the Error Browser from the Hyperview window in three ways:

- Click the **Error Browser** button  in the toolbar.
- Press **E** on your keyboard.
- Click **View | Error Browser** in the main menu.

The Error Browser will be displayed showing the errors that were detected the last time the project file was

built. Click on the + next to any of the branches to expand it and show the individual errors inside. Clicking on an individual error will cause the full error message along with brief documentation to appear in the panel below the error tree. Most errors contain hyperlinks (in blue). Click on these to display the associated locations in the main browser.

2.7.3 Marking errors as Starred, Reviewed or Hidden

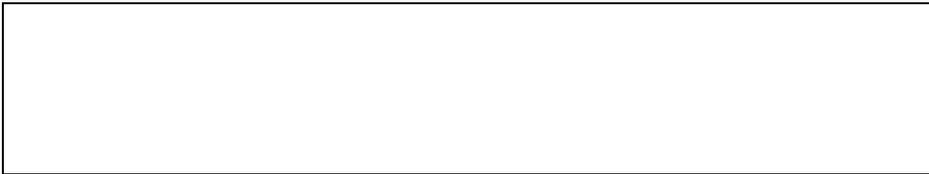
In order to help manage errors in the Error Browser, you can mark them in one of three ways:

-  **Star** – use to indicate that an error requires attention.
-  **Review** – use to indicate that an error has been looked at and evaluated.
-  **Hide** – use to indicate that an error does not need to be further examined and can be hidden from view.

Using the **Show these errors** option on the left, you can filter what errors are shown in the tree based on how you marked them. For example, you can show only errors that are "not reviewed." The error marks are persistent, and will reappear when you next run Hyperview, and also when you view a new version of the project file in Hyperview.

2.7.3.1 Marking individual errors

You can toggle whether an individual error is marked as "starred," "reviewed," or "hidden" by clicking on the small icons to the left of every error item in the tree.

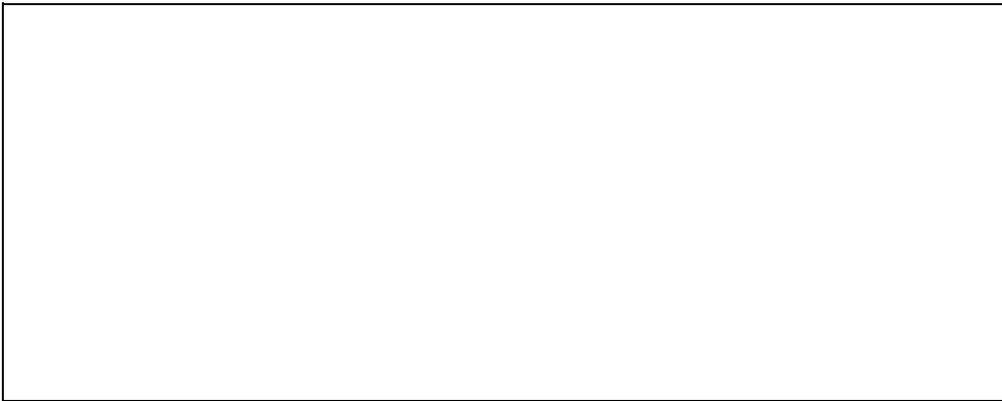


2.7.3.2 Marking selected errors

You can also select an error, and use the toolbar icons to mark it.



If you click on a "branch" item in the error tree, all of the errors in that branch will be selected. You can then use the toolbar to mark all of those errors at once. **Shift+Click** can be used to select a range of errors, and **Ctrl+Click** can be used to select multiple non-adjacent errors.



2.7.4 Filtering errors

All the errors that were detected during the last DBDOC build are built into the project file and can be displayed in the error browser. Because large number of errors are common in systems that have not yet been "cleaned up," it is important to be able to filter them in various ways.

The **Show these errors** drop-down menu to the left of the error tree allows you to display only errors that have been marked in certain ways. For example, choosing "Hidden" (and then pressing **Rebuild**) will cause only the errors that have been marked as "hidden" to be shown in the error tree. "Active" will show the non-hidden errors. Similarly, starred, reviewed, and not reviewed errors can be specifically displayed. You can always see every error by selecting "All" as this option.

As you become familiar with the errors in your build, you will invariably discover categories of potential errors that are not "real" errors in your system because some context explains their existence and they are known to not be harmful. Errors with "CHECK" severity will tend to fall into this category. Since those errors are not "real," they may never be corrected at the source (in your actual configuration) but you can nevertheless still mark them as "hidden" so that they do not clutter up the Error Browser error display.

2.7.5 Grouping errors

The error tree can be organized in a wide variety of ways based on the attributes of the individual errors. Most errors are associated with particular locations in the project, and can be grouped by file. Errors can be associated with particular subsystems, such as graphics, configuration, or database, and can be grouped accordingly. Many errors refer to particular blocks, and can be organized by Loop, PCU, and Module. Errors can be classified according to their severity. Some errors are serious and should be corrected. Others are cosmetic, and could be tidied up, but can also just be hidden if you don't want to deal with them. If you have worked with DBDOC's previous text file error support (.ERR files) you may find it convenient to group errors according to what error file they appear in.

Because entire branches of the tree can be selected with a single click, and then marked with another, the ability to group errors provides a powerful way to understand and manage your errors.

2.7.6 New errors

Every time your project file is rebuilt, the errors generated during the build are built into the project file for viewing in the error browser. When you open the error browser, information about the errors is stored in a local database by Hyperview. This allows Hyperview to detect "new" errors (i.e. errors that have newly appeared in the new version of the project file you are looking at). These "new" errors are highlighted in the error browser to draw them to your attention.



2.7.7 Sharing error information with other users

If you register for data sharing, other Hyperview users will be able to see which errors you have checked or starred, and you will be able to see which errors they have marked.

This is a powerful capability that makes it easy for a group to collaboratively assess and review errors. You could, for example, filter to show only the errors that had been reviewed by another user, select them with a single click, and then hide them all in your Error Browser. Or you could selectively display only the errors starred by another user, select them, and then mark them as "reviewed."

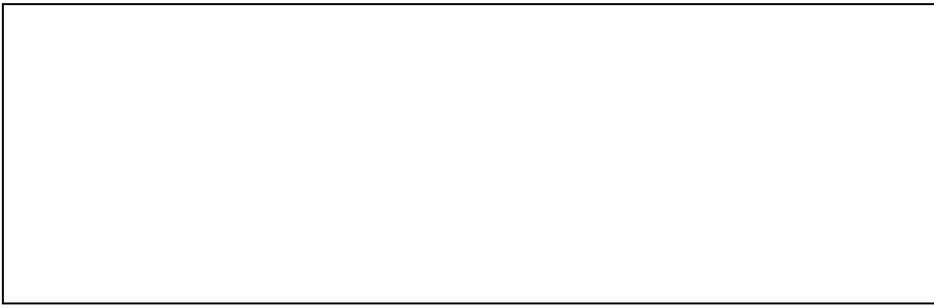
You can register to share data using the **Sharing** tab of the **Properties** dialog, accessible from the **Options** menu in the main Hyperview browser. Once you are registered, your error information will be automatically shared, and you will see the information of other sharing users.

Once you have registered for sharing, other registered users using the same data sharing folder will show up in this checkbox list. Check the checkbox for a user to make their data show up in the tree. In the example below, stars and checks from Jane and John will be displayed to the right of the errors in the tree.



By selecting various combinations of **Show these errors** and **Include stars/checks from** options, you can easily display very specific categories of errors.

For example, this combination shows only the errors starred by John:



Hyperview automatically detects changes made to errors by other users. When such a change is detected, the **Rebuild** button  in the Error Browser activates. The detected changes will be included when the tree is rebuilt.

2.8 Customizing the Browser Display

2.8.1 Showing and hiding specs and attributes

2.8.1.1 Viewing specs on CAD/CLD diagrams

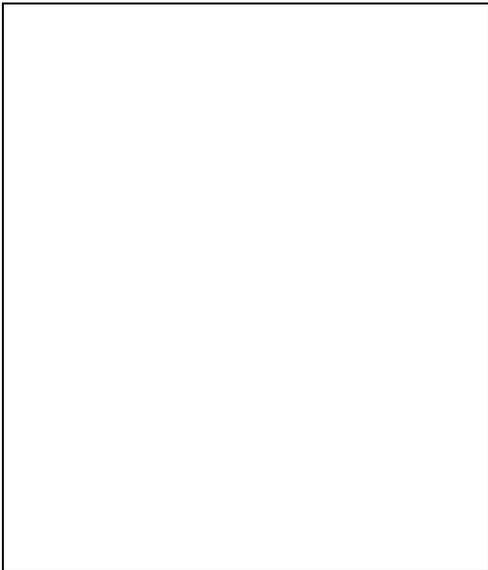
On a CAD/CLD sheet, block specifications can be hidden, or viewed in two different ways - either drawn directly on the sheet or drawn on a gray background.

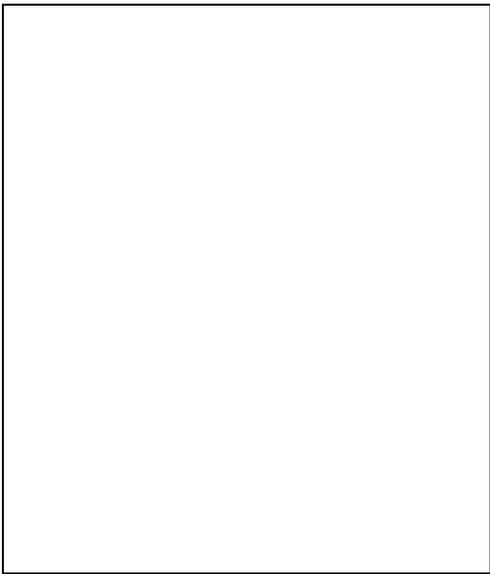
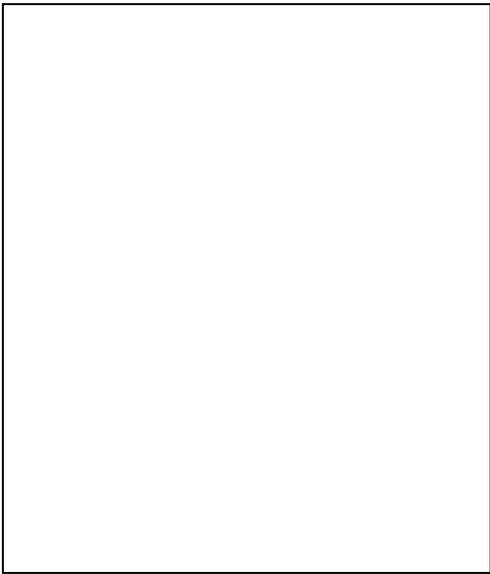
To view specs:

- Press **S** on the keyboard
- **OR** Select **Specs** on the **View** menu and choose the view you want.
- **OR** Right-click on a CAD/CLD Sheet, select **Specs** in the **View** option of the popup menu and choose the view you want.

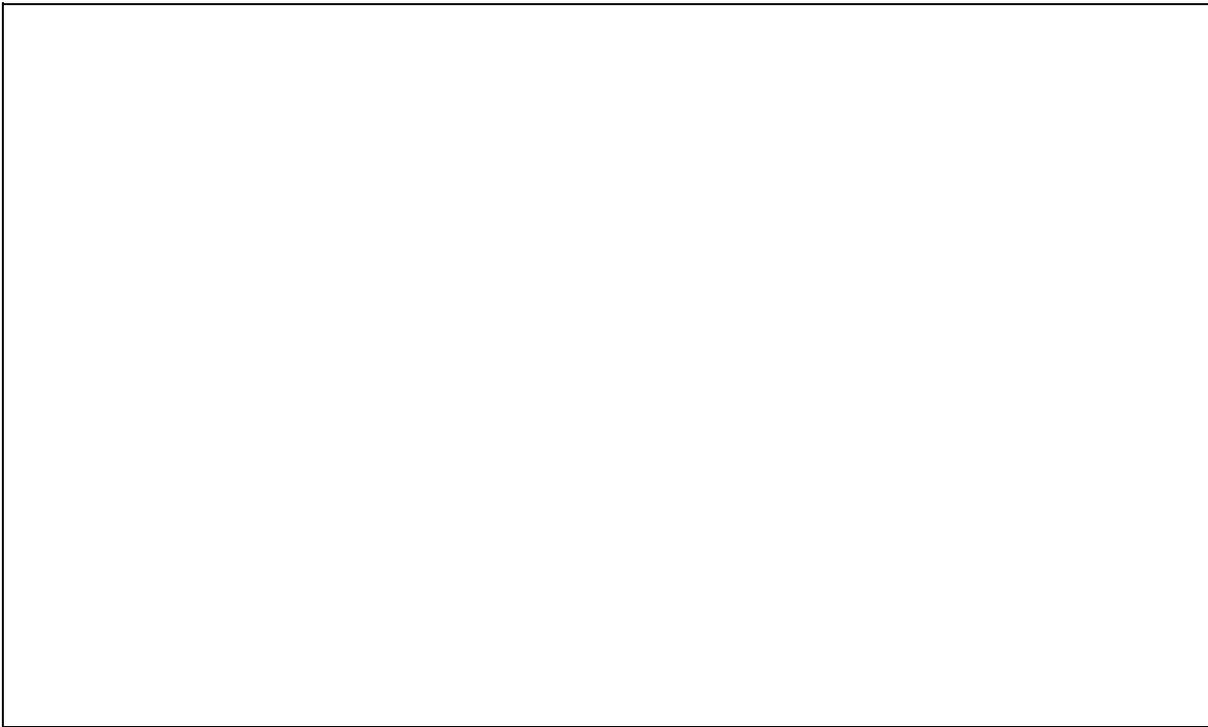
Note: The key **C** formerly was used for this purpose, until DBDOC 10.2.

Cycle through the different displays and turn off the specs by pressing **S** on the keyboard.





What if the specs are obscuring other information on your CAD/CLD sheet?



- In the **opaque** mode (layered specs), click and drag the specs windows to different locations.

Some interesting features of the Specifications window:

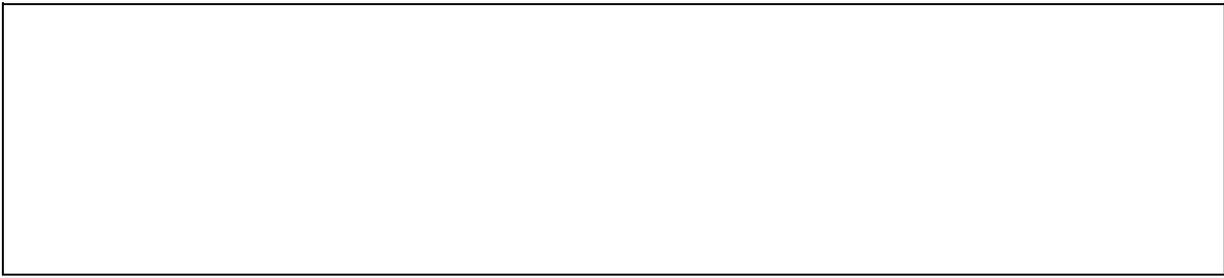
- A "rubberband" links the spec window to its original position so that it is always clear which set of specs belongs to which block.
- Double-click on the spec window and it moves back to the original position at which it appeared.
- The locations of your specs are retained between Hyperview runs so once you have moved your spec windows to the locations that you like, they will stay there until you move them again.
- Live values for specs modified by Adapt Blocks is shown right next to the block they belong to.

2.8.1.2 Viewing attributes on CAD/CLD diagrams

On a CAD/CLD sheet, DBDOC displays the tag name and tag description above a tagged, exception-reported block. Like the specs, the attributes can be hidden, drawn directly on the sheet or drawn on a gray background. To view attributes:

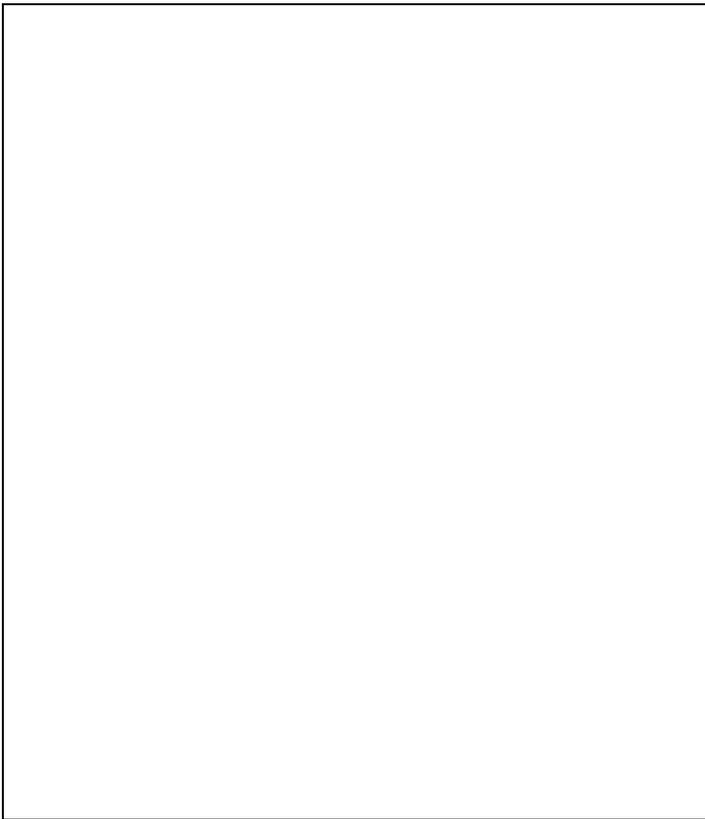
- Press **A** on the keyboard
- **OR** Select **Attributes** on the **View** menu
- **OR** Right-click to bring up the popup menu and select **Attributes** from the **View** menu option.

Cycle through the different displays and turn off the attributes by pressing **A** on the keyboard.



2.8.1.3 Showing live specs on CAD/CLD diagrams

New in DBDOC 11.1, live specs can now be displayed on top of static specs. These can be enabled by selecting **Enable Live Specs** in the **View** menu.



Live specs can then be turned on and off individually for each block by double clicking the camera in the bottom right corner.

2.8.2 Highlighting options in Hyperview

2.8.2.1 Highlighting search hits

Search Hits: With this option checked, search hits are highlighted on the document and you can scroll to the next or previous search hit. This may also be turned on via the popup menu that is brought up by a right-click. Select **Search Hits** from the **Highlight** sub-menu.

2.8.2.2 Highlighting hotspots

Hotspots: Highlights all the hyperlinks on a page. This is useful for determining the location of hyperlinks in a textual or graphical display. You can also use the **Ctrl+Tab** key combination to show all the hotspots on a page.

2.8.2.3 Showing "Vegas Lights"

Vegas Lights: With this option checked, a colorful marquee will appear to highlight your target. You can press **V** on your keyboard to toggle **Vegas Lights** on and off.

2.8.3 Options Menu: Properties for Hyperview

2.8.3.1 View options



An overview of the features available on the **View** tab of the **Options** window:

- **General Settings:**
 - ◆ **Wrap toolbar:** Controls whether the toolbar will wrap around instead of getting cut off if the Hyperview window is too small.
 - ◆ **Full document path in title bar:** If set, the title bar will display the entire path to the open project file.
 - ◆ **Legacy scroll & zoom mode:** Controls scrolling and zooming are inputted on graphics/CADs/etc. See the Guide to Basic Navigation in Hyperview for more information.
 - ◆ **Default Date Format:** Specify the date format you prefer to be used throughout Hyperview.
 - ◆ **On Startup:** If set to "Open to last viewed page," Hyperview saves the last document viewed before it closes so that when it is reopened, the project will be opened to that document. If set to "Open to Home page," the project will open to pages specified by the **Home** button.
 - ◆ **Magnify text on text documents:** Magnifies the all text documents in Hyperview by the set amount.
 - ◆ **Enable GDI+ drawing:** Under certain circumstances, CAD/CLD drawings can have "missing lines". This tends to happen on certain platforms when displayed on a remote desktop. Usually switching to the GDI+ graphics library solves this problem if you encounter it.

- **Hotspot Enabling:** Specify whether you wish to enable Function Code and Function Block hotspots. These are not critical to regular signal tracing, and can be disabled for convenience.

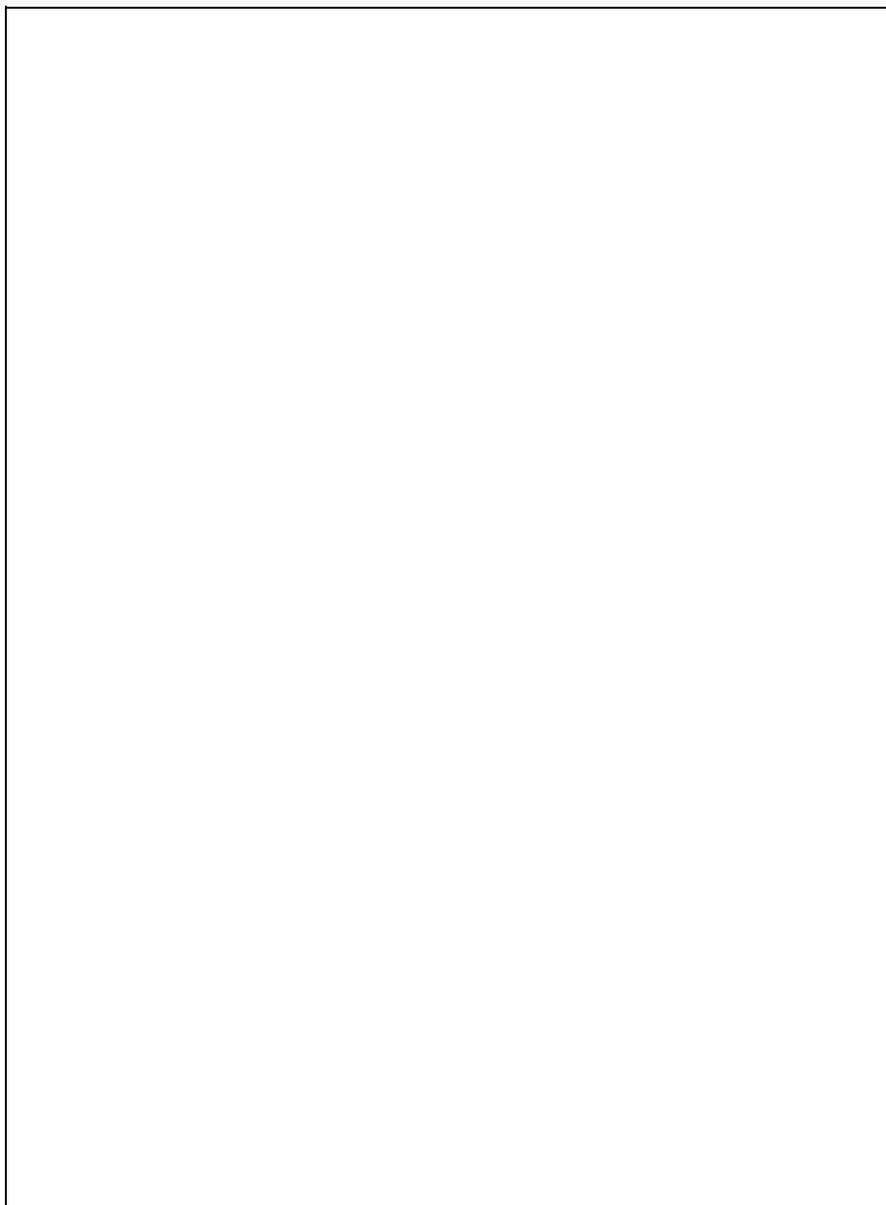
- **Live Data Appearance:** Select the font and font size used to display live data.
- **Text Annotations:** Select the size of the font for the text that appears in your text annotations.

- **Live Data Communications:**
 - ◆ **Update interval:** Set how often on-document live data is fetched in continuous mode.
 - ◆ **Stale data interval:** Set how long data is displayed as fresh before being displayed as stale.
 - ◆ **Enable legacy networking code:** This causes Hyperview to use an older method for connecting to CIUMon, required on some older platforms. It should not be harmful on newer platforms.
 - ◆ **Enable networking log:** Debugging feature.
 - ◆ **Enable live data for Live Loop Annotations and Watch Window blocks with no source:** Attempt to fetch on-document data even for blocks that neither have a source in the configuration built into the project file, nor have been marked available in BuildPlus. Turning this on may cause problems, since data for such blocks may not actually be available.

- **Folders to Search for External Files** Specify additional folders to be searched for external files (files not built into the project file). See **Viewing external files** for more information.

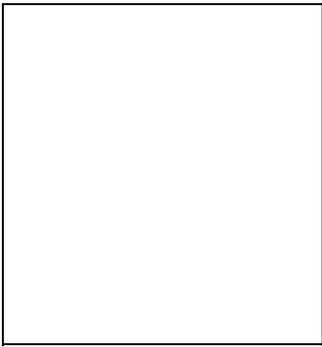
- **Reset to show all suppressed dialogs:** Check this option to have the **No Search Results** and **Bookmark Message** dialogs appear the next time they are triggered.

2.8.3.2 Color options



These options allow you to change the colors of various features within Hyperview. For example, you can change the colors for annotations.

Text Annotations:



Live Loop Annotations:



Restore default annotation colors with the **Reset colors to default** button.

2.9 Making Notes on Documents

2.9.1 Overview of Annotations

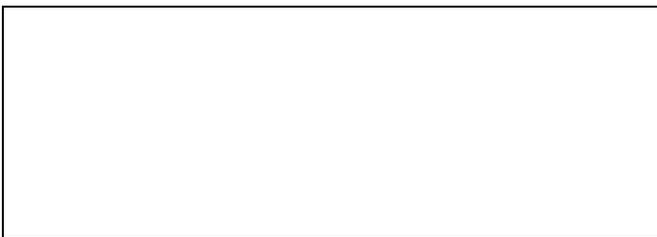
Annotations are like electronic sticky notes that you can attach to the majority of the documents contained within your project. There are two types of annotations that you can create:

1. **Text-based Annotations:** Make comments on a document to keep track of important things to note.
2. **Live Loop Annotations:** Create a note on a document that can fetch the live data value of any particular point in your system from anywhere in your project .

2.9.2 Text Annotations

Annotations are notes that can be created by two methods:

- Right-click on the document and select **Create Annotations** from the **Annotations** option on the popup menu.
- Click the **Create an Annotation** button on the toolbar, and place the annotation on the desired document.



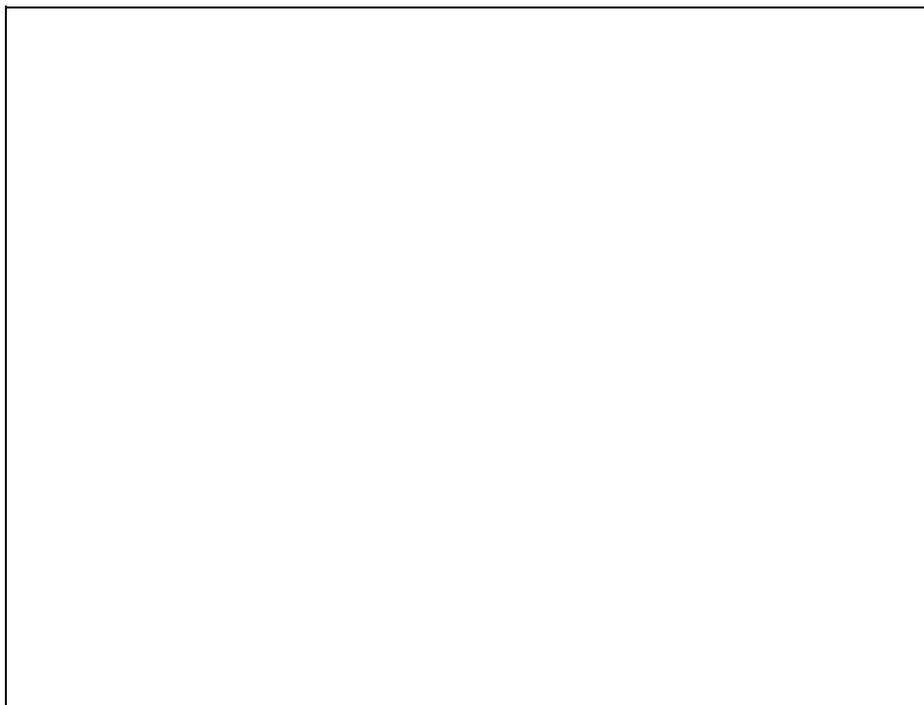
Text annotations will persist across builds of your system. If you remove a topic, it will be listed as an *Invalid Topic* in the **Manage Annotations** dialog box. If you add a file back into your build it will activate the annotations that were originally on that document.

2.9.3 Managing Annotations

To move an annotation, drag and drop the annotation window or anchor point to the location you want.

If you have to make the same note in different places, you can make a copy of the original annotation and paste it anywhere you want. On the original annotation, right click and select **Copy Annotation** from the

popup menu. Navigate to where you want to paste the annotation, right-click, and choose **Paste Annotation** from the **Annotations** menu.



You can minimize or open all text annotations on a document by selecting the **Minimize All Annotations** or **Restore All** options respectively in the **Text Annotation** option on the popup menu.

- Minimized annotations will appear on the document represented by an anchor point.
- To open an individual annotation, click on the anchor point.
- Click the minimize button in the upper right-hand corner to minimize an annotation.

You can view a complete listing of annotations by:

- Clicking the **Manage Annotations** button on the toolbar
- **OR** selecting **Manage Annotations** under the **View** menu

This will bring up the **Manage Annotations** dialog box: Click on the **Annotations** tab.



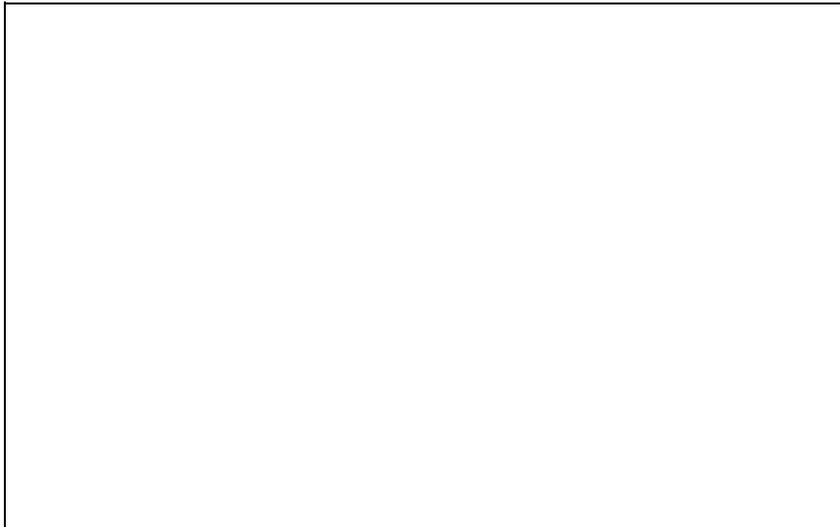
Clicking on either the topic or an annotation in the list will automatically take you to that document. You can edit text in the annotation by either:

- Changing the text in the original annotation window (double click on the window to open the editor or right click on it and select **Edit Annotation** from the popup menu)
- Edit the text in the right hand pane of the Annotations page in the **Manage Annotations** dialog. The changes will be reflected in the actual annotation window when the user makes a different selection in the document tree or the **Manage Annotations** dialog is closed by clicking the **OK** button.
- You can also delete individual annotations by right clicking on an open annotation and choosing **Delete Annotation** in the popup menu that appears or by clicking the in the upper right-hand corner.

How to restore a deleted annotation:

1. Close the project file.
2. Edit the [project name] .ann file.
3. Remove **Delete...** line
4. Modify **Visible** from 0 (zero) to 1. You do not have to change the associated hexadecimal value 0x00000000.

You can look for specific text in your annotations by clicking on the **Search Text** button in the **Manage Annotations** dialog.



Type the text you are looking for in the **Find What** edit box, choose what you want to search in, and then click the **Search** button. Any annotations found to match your criteria will be shown in the list. Click on any item in that list to go directly to the document.

You have the ability to print your documents either with the annotations visible or suppressed. When they are suppressed, you won't even see the anchor point on the printout.

To suppress annotations while printing, uncheck the **Print Annotations** checkbox in the **Print** dialog or on the **Print Settings** tab under the **Options** menu.

To print annotations, select **Print Annotated Topics** from under the **File** menu, or click the **Print Annotated Topics** button in the **Manage Annotations** dialog.

The annotation source file is called [project name].ann. The annotation file is located in the following directories:

- **For Windows Vista/7/8 and later:** C:\Users\\AppData\Roaming\GMCL\DBDOC
- **For Windows 2000/XP/2003:** C:\Documents and Settings\\Application Data\GMCL\DBDOC
- **For Windows NT 4:** C:\WINNT\Profiles\\Application Data\GMCL\DBDOC

2.10 Printing Documents and Groups of Documents

2.10.1 Printing a single document

You can print the active pane by clicking the **Print** button  on the toolbar or by selecting from the various print options under the **File** menu. The pane that currently has focus will be printed.

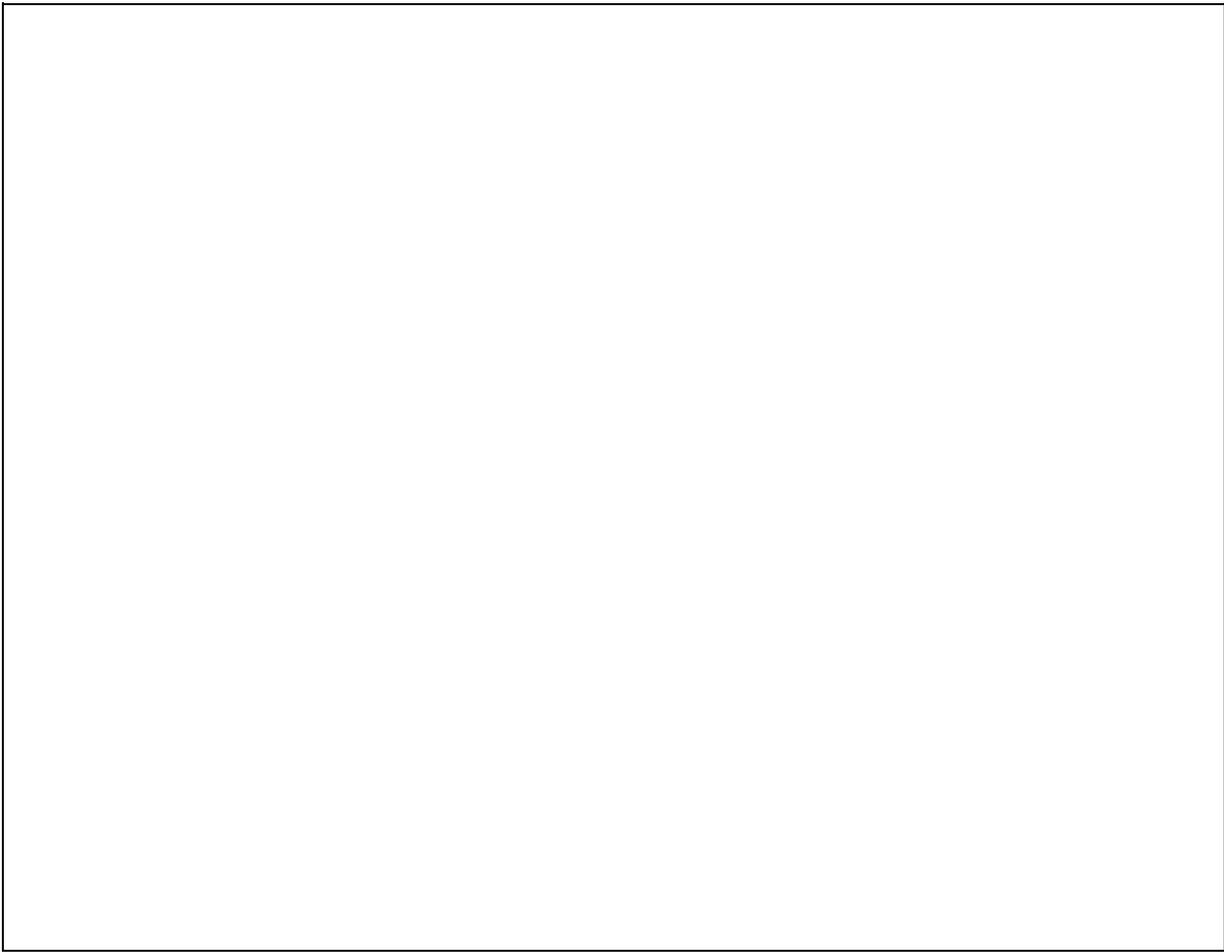
If a dialog box appears asking you if you really want to print an index page, the answer is probably No. This is just a safeguard in case you have accidentally left the focus on the wrong pane. If you want the content (graphic or CAD/CLD sheet), make sure the focus is on the content (right/lower) side and print again.

Pages are printed with the drawing name, date and time and custom captions can be inserted. Our caption feature (in **Advanced Options**) lets you record a useful comment for this topic that will be printed at the bottom of the drawing.

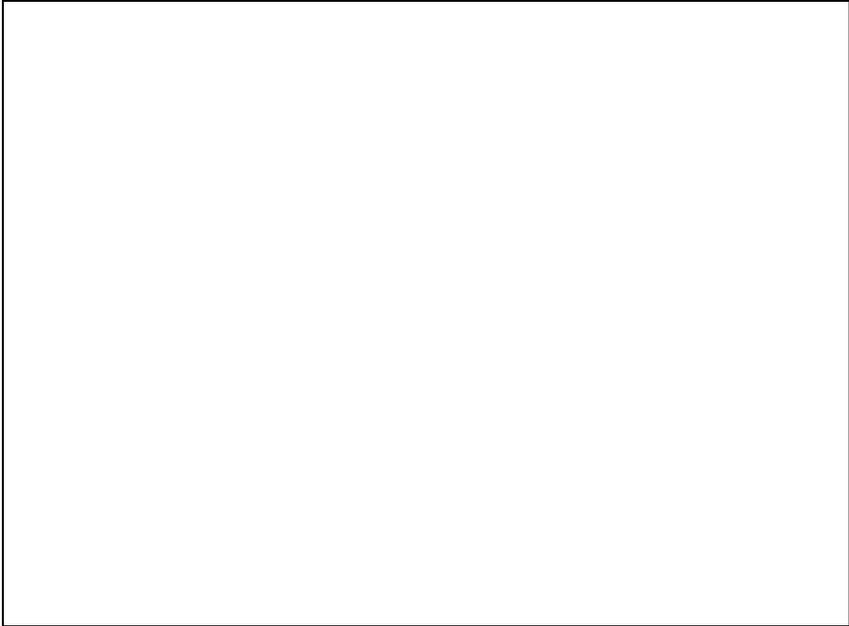
Zooming in on a particular area gives you better resolution. A diagram will appear in the lower right hand corner of the print out, showing the portion of the drawing being printed.

2.10.1.1 Printing options

Various options can be set for printing:



Any setting changes made here in the print dialog under the **Advanced Options** (e.g. **Suppress automatic color reversal**, **Annotations**, etc.) will be reflected in the **Print Settings** tab under **Options**.



2.10.1.2 Fast Print to use your previously defined print settings

After doing the first print, you can avoid seeing the **Print** dialog box before each subsequent print by:

1. Selecting **Fast Print** from under the File menu.
2. Pressing **P** on the keyboard.

Note: The print dialog box choices from the first print will be applied to subsequent prints.

2.10.1.3 Printing selected text

You can select a portion of any text document, including the contents of the index pane, to print. The **Selection** radio button in the print dialog box will be enabled.

2.10.2 Marking the current topic for later printing

This option is useful when a large number of topics needs to be printed at one time. You can use the easy popup menu interface to add a document. Right-click on the document and select the **Mark This Topic For Printing** option from the popup menu.

2.10.2.1 Marking the current topic for printing

When you choose **Mark This Topic For Printing** in a pop-up menu, the current topic will be marked for printing, and added to the **Marked Topics** list in the **Mark Topic for Printing** dialog.

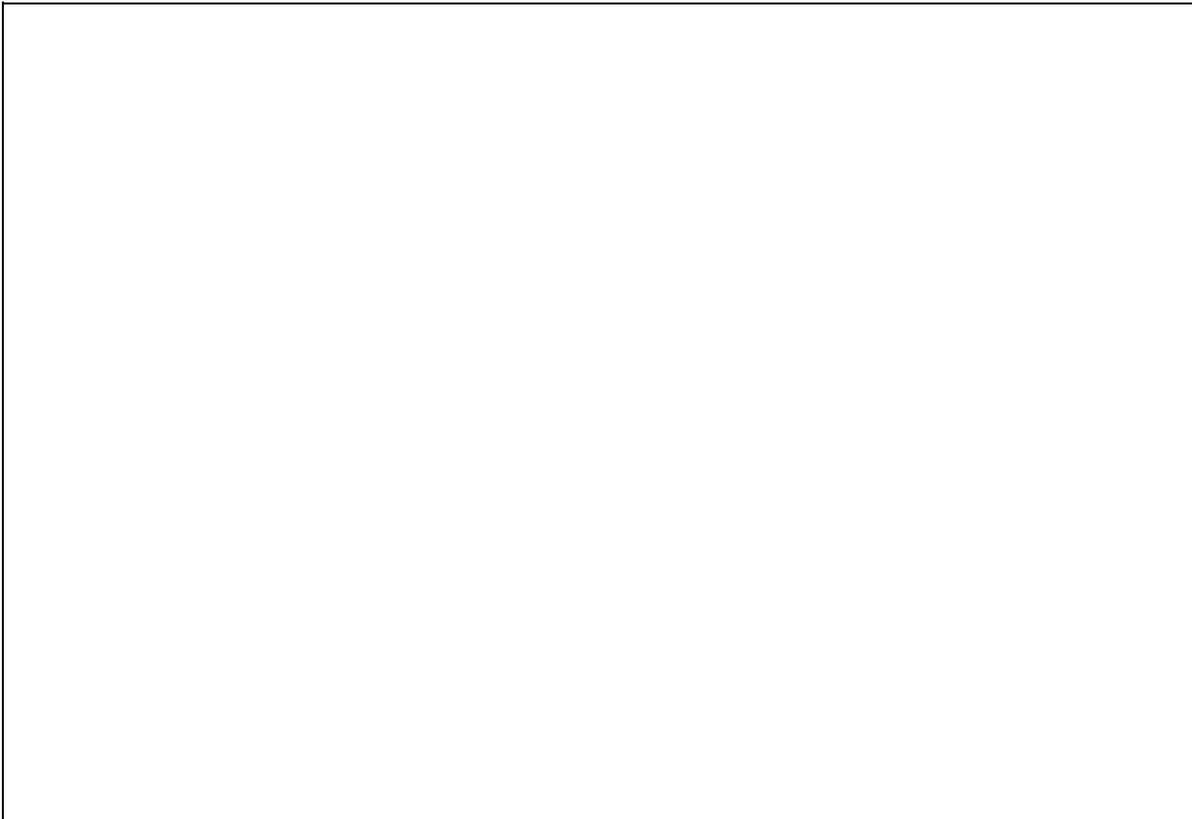
2.10.2.2 Marking a linked topic for printing

If you choose **Mark "[document]" For Printing** in a hotspot pop-up menu for a hotspot in the index pane than links to a document (for example, a graphic), the linked topic will be added to the **Marked Topics** list.

On the **File** Menu, click **Mark Topic For Printing....** This displays the **Mark Topic for Printing** dialog.

2.10.3 Using the Mark for Print dialog to print groups of topics

The **Mark for Print** feature allows you to "mark" or select a topic or group of topics for printing. The marked topics are stored in the **Marked Topics** list shown in the **Mark Topic for Printing** dialog. Once topics are in the list, they can be printed en masse. This feature allows you to easily collect topics for printing as you are browsing, and also to print all (or some) topics of a certain type, for example CAD sheets, BATCH files, or topics with annotations, without having to view and select each topic individually.



2.10.3.1 Marking a group of topics for printing

In the **Mark Topic for Printing** dialog,

1. Select **All Topics in Group** from the **Mark** Menu.
2. Select a group of topics from the **Groups** Menu.
3. Click **Add**

4. Make sure the the desired group has been added to the **Marked Topics** list box, and items to be printed are checked off. The topic groups in the **Marked Topics** list box are expandable, and individual items can be unchecked if you do not wish to print them.
5. If you wish to print the topics in the **Marked Topics** list now, click **Print Topics**.

2.10.3.2 Removing topics or groups of topics from the marked topics list

In the **Mark Topic for Printing** dialog,

1. Select the desired topic or group to remove by highlighting it in the **Marked Topics** list box.
2. Click **Delete** to remove the highlighted group or topic.

OR

1. Click **Delete All** to remove all items in the **Marked Topics** list box.

2.10.3.3 Other options

In the **Mark Topic for Printing** dialog,

- **Remove all topics from Marked Topics list after printing:** General topics shown in the **Marked Topics** list are cleared when the dialog box is closed if they have been printed.
- **Enter/modify caption for selected topic:** Allows you to create or change a caption for the selected topic. Use this option to include comments about the topic; the drawing title is printed automatically above the caption.

2.11 Sharing Data with Other Hyperview Users

2.11.1 Overview of User Data Sharing

Choose **Options** in the main menu to open the **Properties for Hyperview** dialog. Click the **Sharing** tab to display the Hyperview data sharing settings.

Certain user data can be shared "live" among users. For example, the stars and checks with which one user marks errors in the Error Browser can be displayed in the another user's Error Browser.. Sharing is not yet supported for all user data, but over time support will be increased. Eventually annotations, live loop annotations, and watch window groups will all be able to be shared easily among users.

When you run Hyperview, your personal user data files are stored in a hidden folder. When you enable data sharing, some of these hidden data files are copied to a central repository, the **Shared Data Folder** where they are available to other Hyperview users. The data in them can then be displayed in Hyperview browsers belonging to other people.

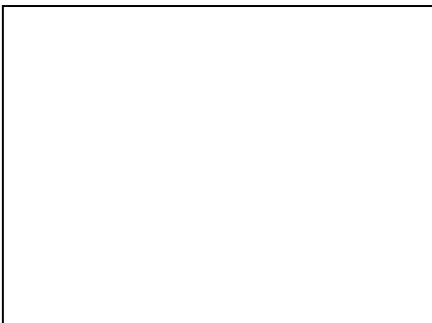
2.11.2 Specifying the Shared Data Folder

In order to share user data, a **Shared Data Folder** must be specified. This can be any folder on your network file system to which all the sharing users have read and write access. Usually the name of this folder will be built into the project file in BuildPlus, and you won't need to change it. But if it is not built in, you can specify it here.



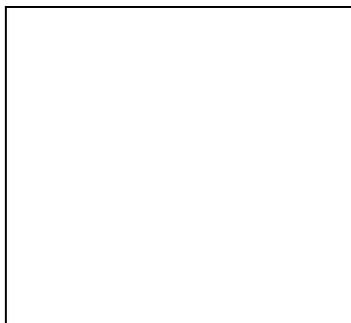
2.11.3 Updating your user information

You can specify or update your first and last name, and enter a note. Other users will see this information. If you leave these fields blank, other users will still be able to see your username and computer name.



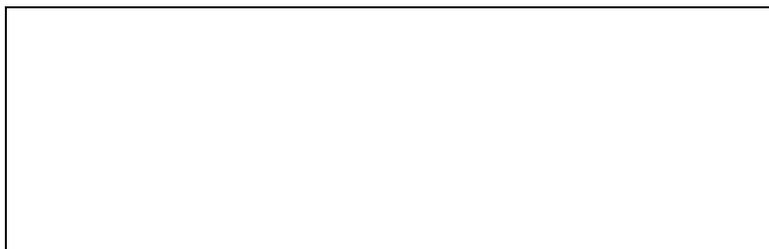
2.11.4 Specifying what sorts of user data to share with others

In this area, specify which data you want to share with other users. For example, if you don't want to let others see your error information, but still want to see theirs, just uncheck **Error Stars & Checks**.



2.11.4.1 Specifying whose data to include by default

You can specify the users whose data you wish to include by default (and override this default elsewhere if you need to).



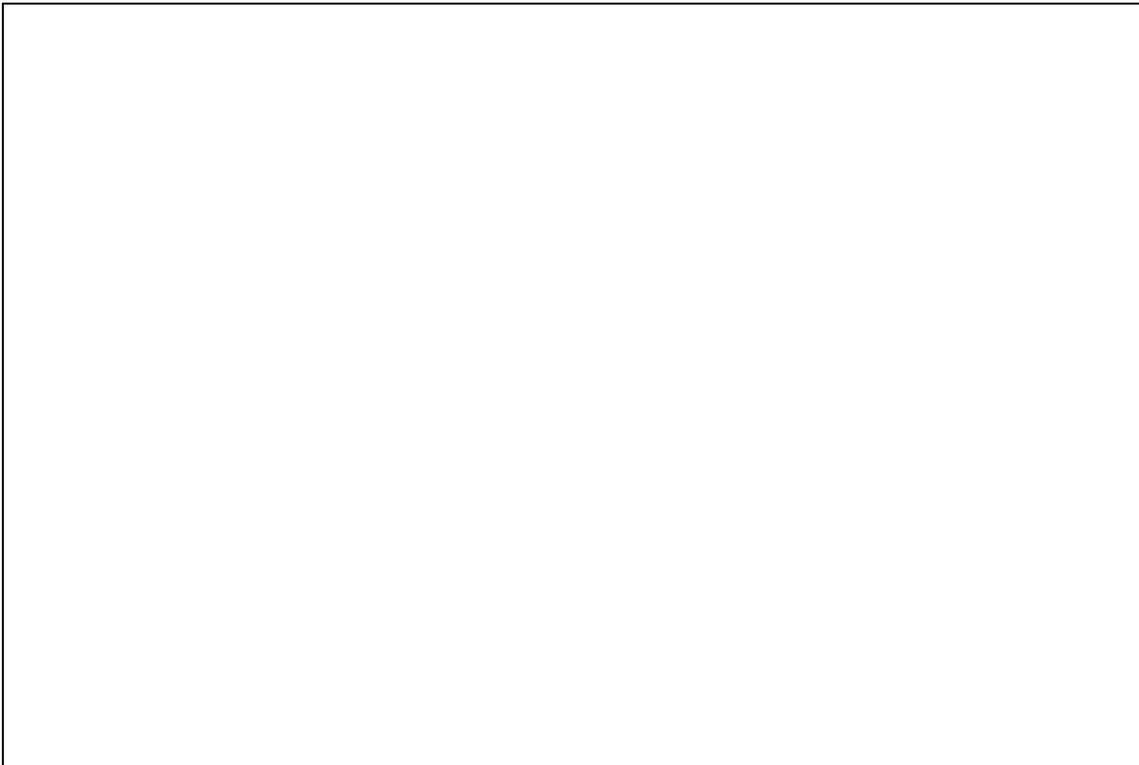
2.12 Tools, Reports and Indexes

2.12.1 Viewing the resources used in a drawing with the Audit Window

You can use the Audit Window to view the resources used in a CAD/CLD or graphic. This includes:

- Block and tag usage
- Input and Output references
- Function blocks
- Graphics / document links

You can open the Audit Window by either selecting it from under the **View** menu, from the popup menu on a right mouse-click, or the  button on the toolbar.



Hyperview will highlight the selected target from the Audit Window and zoom in to the appropriate level. You can save the Audit Window contents for printing or editing later.

- Select **File** then **Export**
- Or print them now with **File** then **Print**
- You can print or export the whole list or just the individual parts of the list, i.e. the Blocks, Tags, References, or Function Codes. There are also various formats in which the list can be printed or saved, allowing you to get the information in a format that best suits your needs. A default name will also be supplied.

Below is a drop-down menu for Audit Window/File/Export:

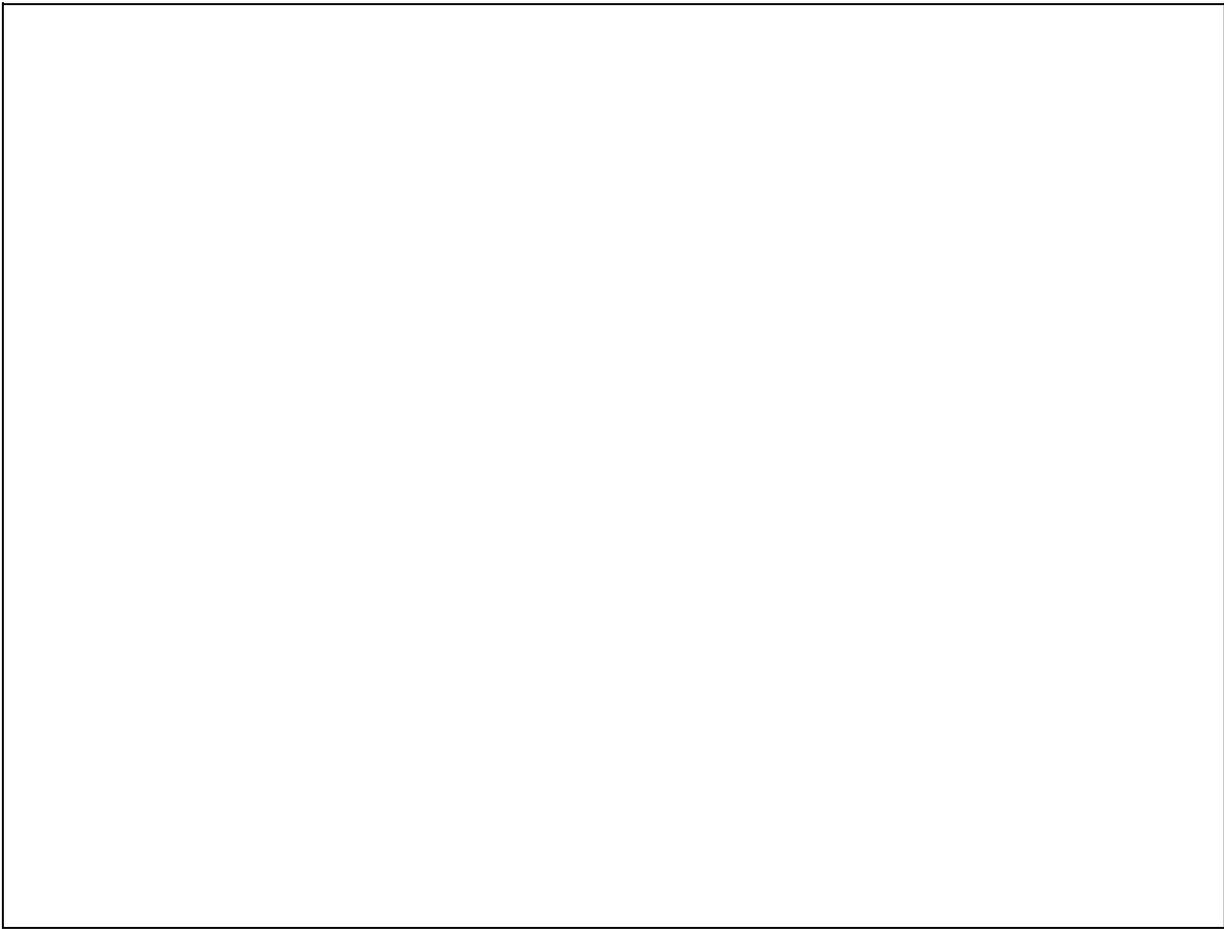


Why do you want to use Audit Window?

- Checklist for tags
- All places a block is used on the CAD/CLD or graphic
- Verify that a block is used consistently
- See the tags on an AutoCAD or MicroStation sheet
- On graphics, determine if the uses of a tag are spread across the page

2.12.2 Using PCU Map to understand interactions among PCUs

The PCU Map shows you the interaction between PCUs, listing what blocks are exported and imported between two units. Access the PCU Map from the **System Information** chapter by clicking on "**PCU Interactions (Block Imports and Exports)**". Click on the **PCU Report Page** link to bring up a table similar to the following:



Click on a link in the table to bring up the specific block interaction between two chosen PCUs.



2.12.3 Graphical Block Map: a graphical overview of block usage

Block map is a graphical overview of block usage, showing configuration aspects and issues at a glance.



The Block Map can be accessed by either:

- Clicking on the **Block Map** link available at the beginning of each chapter of CAD/CLD sheets.
- Right-clicking on a CAD/CLD Sheet and selecting **Go to Block Map** from the popup menu that appears.
 - ◆ If you right-click on a specific block, the corresponding module block map page will be brought up in the content pane and the Table of Contents will display the **Block Map Index** for the module. The selected block and appropriate block range will be highlighted by Vegas Lights (if that option is turned on under the **Highlights** menu).
 - ◆ If you right-click while on white space on the CAD/CLD Sheet, only the Table of Contents will be updated to bring up the **Block Map Index** for that particular module.

Explanation of the Block Map color legend:

 = (Green block) This block is used.

 = (Red Block) This block is used in configuration but the source does not exist (or is not built.)

 = (Orange Block) The block is used on a graphic, but not sourced in configuration.

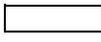
 = (Pink Block) This block is referenced in a database, but not sourced in configuration.

 = (Purple Block) This block does not have a source, but is sourced somewhere in the system.

 = (Blue Border) The block is exported, but not tagged.

 = (Black border) The block is tagged and exported.

 = (Yellow block) The block is tagged, but unused.

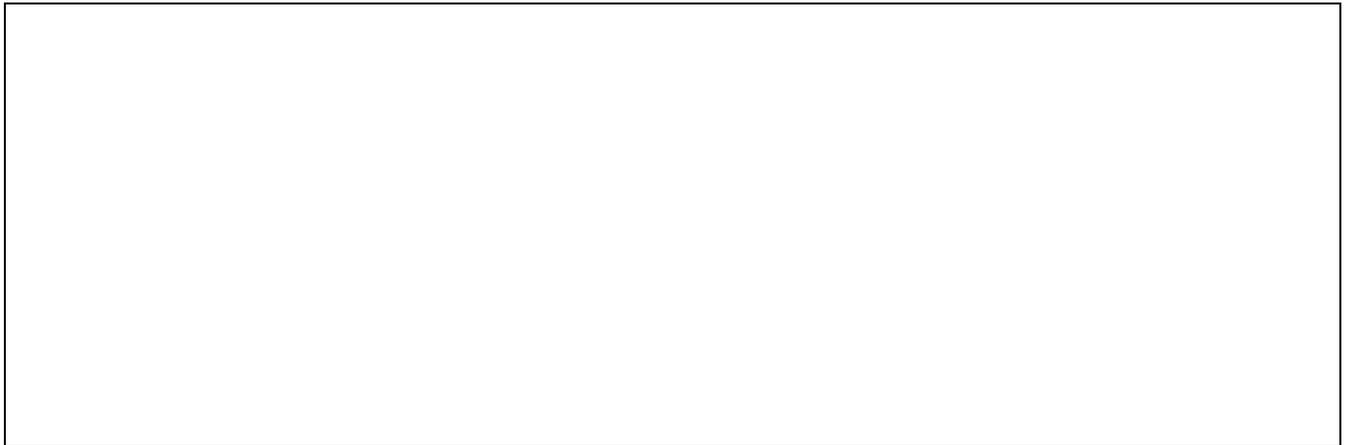
 = (White block and border) The block is completely unused.

Ideally, what you want to see for your blocks is: 

A green block with a black border is a "perfect tag" - tagged, exported and used. The Block Map allows you to see where blocks are available for insertion of a large function code.

2.12.4 Tag Index: an alphabetical index of all the tags in your system

The **Tag Index** is an overall view of all the tags in all your databases.



The **Tag Index** is found in the **Database** chapter in the Table of Contents. It is comprised of an alphabetical and numerical listing of all the tags in your system. Each tag database is a column and each tag is a row. If a tag is present in a database, it will appear in the appropriate column, with its associated Loop, PCU, Module and Block information. The LPMB information is a link that you can click on that will take you to the corresponding Block Index page. If the tag is not present in a database, the text **ABSENT** will appear in the column.

Tips for using the Tag Index:

- For quick loading, switch to **Treeview Table of Contents** mode (press **Ctrl+ T**).
- When you click on a Block Index link, it will highlight the first reference to that block, not the one that you clicked on.
- The database column headers are not repeated but the databases are in the same order as in the Table of Contents (alphabetic).

2.12.4.1 Using the Tag Index

To find inconsistent LPMB in your tags:

- Do a Full Text Search in current topic for Exact Match for ++.
- Inconsistent tags appear in red and are preceded by the text ++.
- In `DBDOC_TAG_MISMATCH.ERR`, look for the **Tag mismatch for X: see TAGMAP for details**.

To get a list of all tags in your system:

- Select and print the individual **Tag Index** pages.
- If you have more than four databases, set your printer to use landscape orientation.

To walk through the Tag Index pages:

- Click on the first letter combination and page down to see all the tags in that section.
- Type **B** for browse to see each section.

Note: You can also locate these graphics and symbols by looking at the **Undefined Tags, Missing Graphics**, and **Missing Symbols or Submodels** lists located in the **System Information** chapter of the project.

- For quick loading, switch to **Tree View Table of Contents** mode (**Ctrl+T** keys).
- When you click on a Block Index link, it will highlight the first reference to that block, not the one that you clicked on.
- The database column headers are not repeated but the databases are in the same order as in the Table of Contents (alphabetic).

Find inconsistent LPMB in your tags:

- Do a **Full Text Search** in current topic for Exact Match for ++.
- Inconsistent tags appear in red and are preceded by the text ++.
- In `DBDOC_TAG_MISMATCH.ERR`, look for the **Tag mismatch for X: see TAGMAP for details**.

Get a list of all tags in your system:

- Select and print the individual **Tag Index** pages.
- If you have more than four databases, set your printer to use landscape orientation.
- Walking through the **Tag Index**:
- Click on the first letter combination and page down to see all the tags in that section.
- Type **B** for browse to see each section.
- For quick loading, switch to **Tree View Table of Contents** mode (**Ctrl+T** keys).
- When you click on a Block Index link, it will highlight the first reference to that block, not the one that you clicked on.
- The database column headers are not repeated but the databases are in the same order as in the Table of Contents (alphabetic).

Sort columns in the tag index:

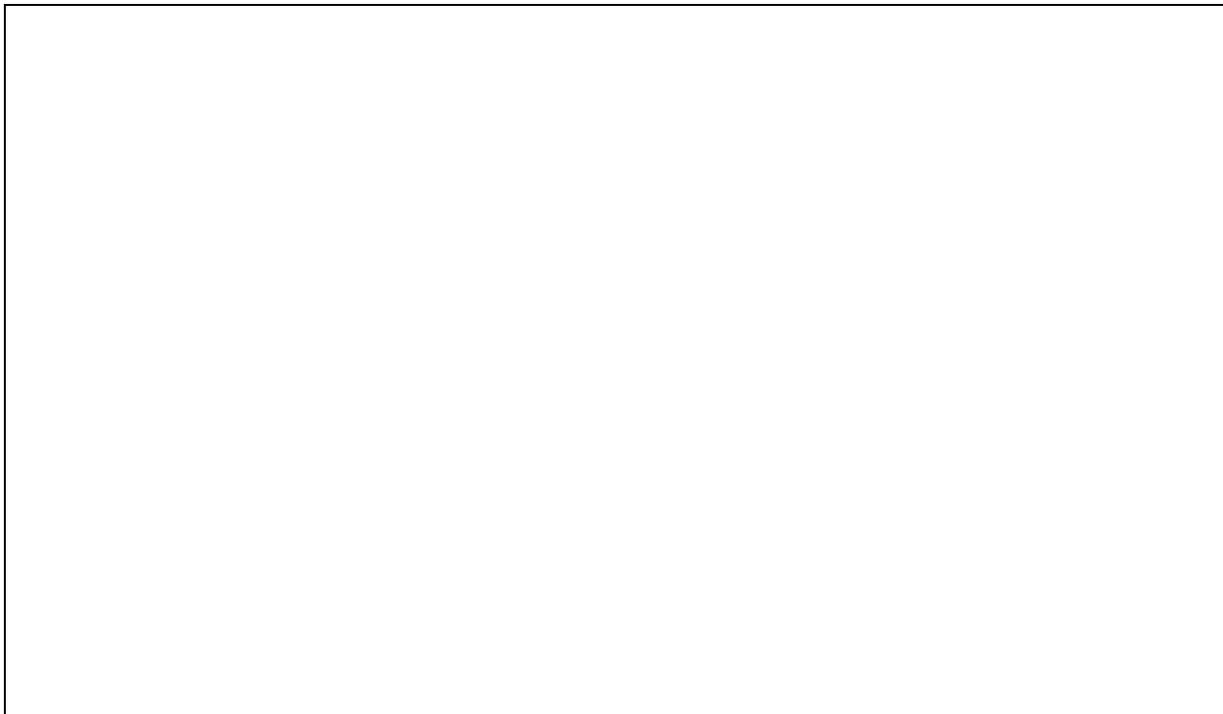
The columns in the tag index can be sorted either ascending or descending alphanumerically. When you mouse over a column it will be highlighted, clicking anywhere in that region will change the sort of the column. The current sort mode is indicated by the arrows to the right of the column name. When the down arrow is highlighted in grey, the column is currently sorted in descending order. When the up arrow is highlighted in grey, the column is currently sorted in ascending order.

2.12.5 Graphics Cross-Reference Window: index of all the graphics that link to and from the graphic, plus included symbols

The **Display Cross-References** option displays an index of all the graphics that link to and from the graphic as well as symbols included in the graphic. This tool also shows symbols individually and any graphics that include the symbol. The **Cross Reference Index** page for a particular graphic provides information similar to the Audit Window except across the whole DBDOC system. When building your project, in the **BuildPlus Project Options**, choose the **Advanced Options** on the **Processing Options** page and make sure the **Exclude Symbols and Submodels** chapters from build is unchecked to get full symbol cross references.

Accessing the Cross Reference Window while on a graphic:

- Click the **Display Cross References** button
- Under the **Go** menu select **Display Cross References**
- Type **X** on your keyboard



A separate floating window will be opened showing you the different Graphics Cross References categories, handily displayed in a tree view:

The position and size of the **Cross Reference Window** will be retained between runs so that you do not need to reset them to suit your preferences each time the feature is used. The window will remain open while you navigate to different documents in your project file until you choose to close it or open up a different project file.

You can always return to the original graphic referenced in the **Cross Reference Window** if you have moved onto different documents by clicking on the "(click here to return)" link at the top of the tree.

This feature can be used to cross-check a number of things in your system:

- Find out which user symbols are no longer used by your graphics
- Step through your symbol chapters
- Bring up the cross-references for each symbol
- Symbols that are never used will have a blank cross-reference page
- Locate graphics not being built but are still being linked to
- Step through your graphics chapters
- Bring up the cross-references for each graphic
- Black entries (no hotspot) in the **Cross Reference Index** indicate that DBDOC was unable to build the graphics called for; either the build was incomplete or some of the graphics are trying to link to graphics that no longer exist

2.12.6 Function Code Descriptions and Uses: an index of all the function codes in your system

The Function Code Descriptions and Uses index is located in the Table of Contents. Click on **Function Codes** under **System Information**. There are three columns of links in this index.

1. Function code name, e.g. **ADAPT**: items in this column link to the documentation for the function code, including basic spec and output descriptions.
2. Function code number, e.g. **[FC24]**: items in this column connect to the corresponding entry in the numerical function code index. Usually you would come at them from the other direction, via the **Function Codes by Number** index in the main table of contents.
3. Function code uses, e.g. **USES(45)**: this item brings you to the function code uses index, an index of all the CADs or CLDs where that particular function code is used. Items in this index are linked to the function code hotspots on CAD sheets.



The link FROM function code hotspots to CAD sheets back to the function code index is disabled by default. In the disabled state, clicking on the function code number will not take you to the function code uses index.

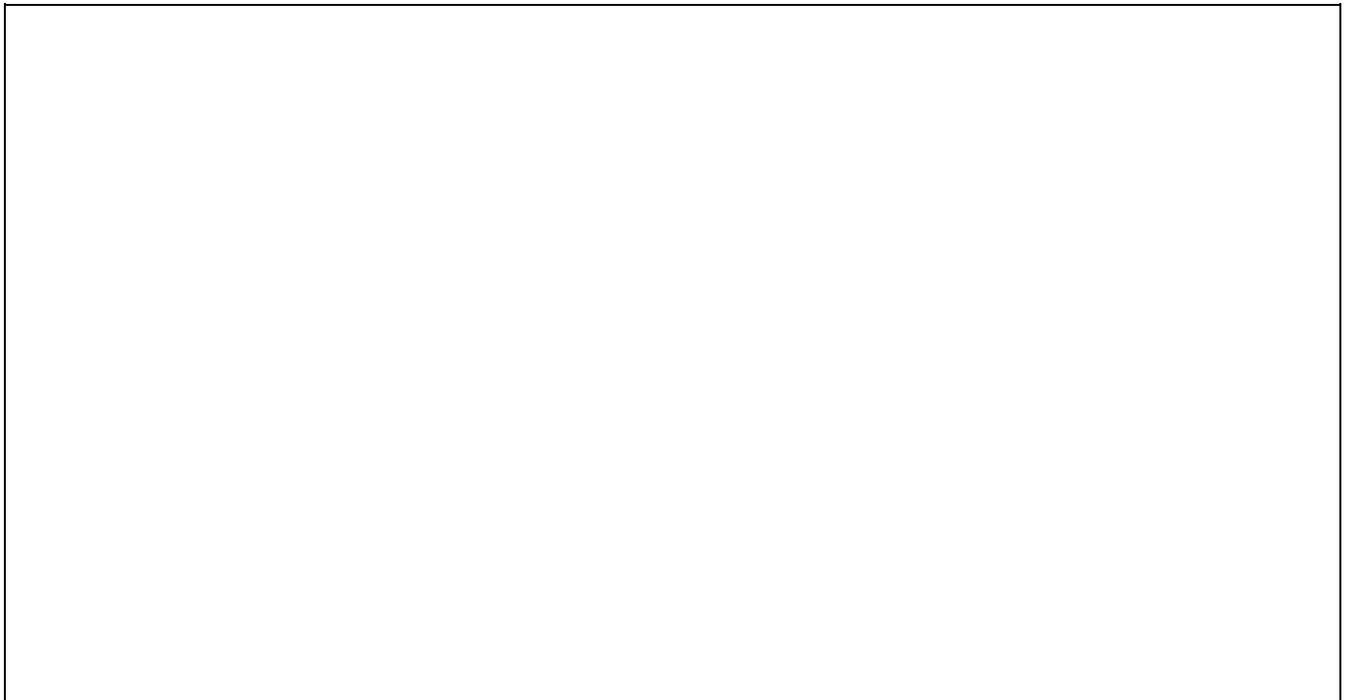
Function Code Hotspots can be turned on or off via the **Enable function code hotspots** checkbox on the **View** tab of the **Options** dialog (in the main browser menu). You can also turn on the option by left clicking on a function code hotspot position that will bring up the following dialog:

You can choose to:

- Enable the hotspots
- Disable the message box from appearing again the next time you click on a function code hotspot position
- Do nothing at all by clicking **OK** without checking either option or clicking the **Cancel** button

2.12.7 Function Block Index: a list of all the function blocks used on a CAD/CLD

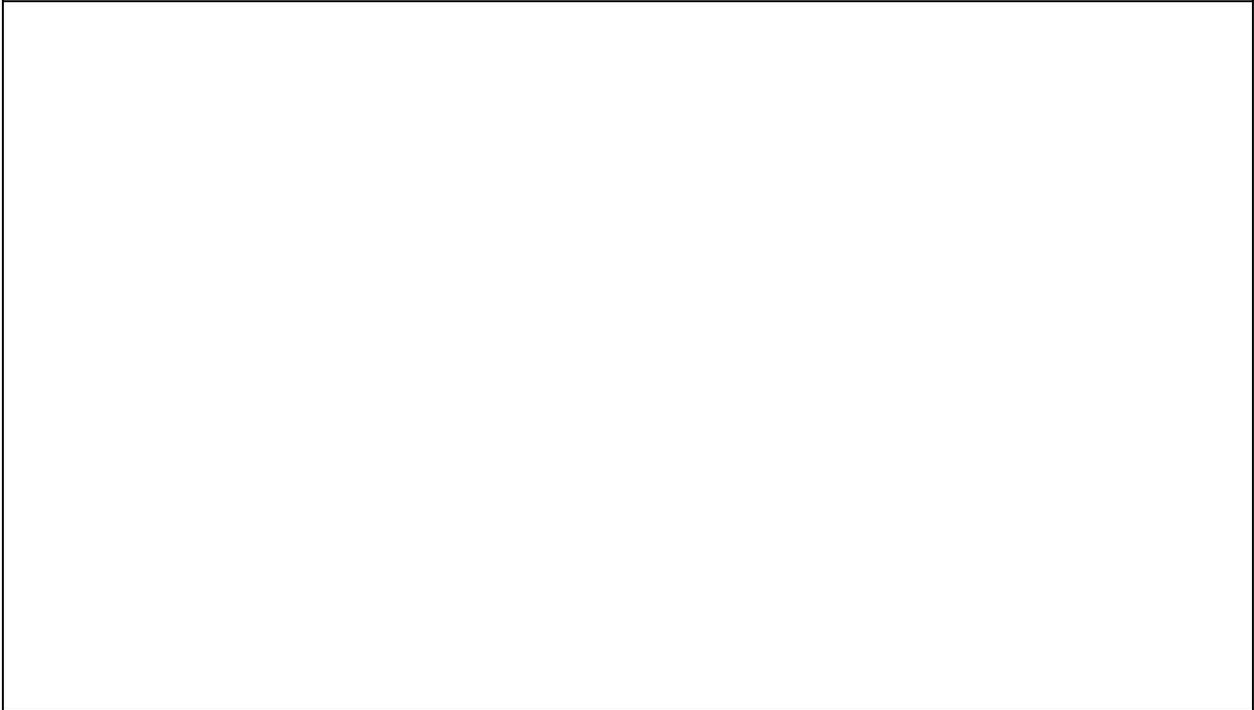
The **Function Block Index** lists and links every function block on a document. You can display it by clicking on any function block on a CAD or CLD. If function block hotspots have been disabled, this index will not be shown. See Enabling function block hotspots for information on how to enable and disable function block hotspots.



2.12.8 Graph of F(x) (Function Code 1)

INFI 90 Function Code 1 implements user-specified piece-wise linear functions. Hyperview displays a sketch of the function in the function box when specs are visible (hint: use the **S** key). Right-click on the block

number and select **Create Live Function Graph for this Block** to get a larger, resizable, movable graph.
Example of a Function Code 1 Graph:

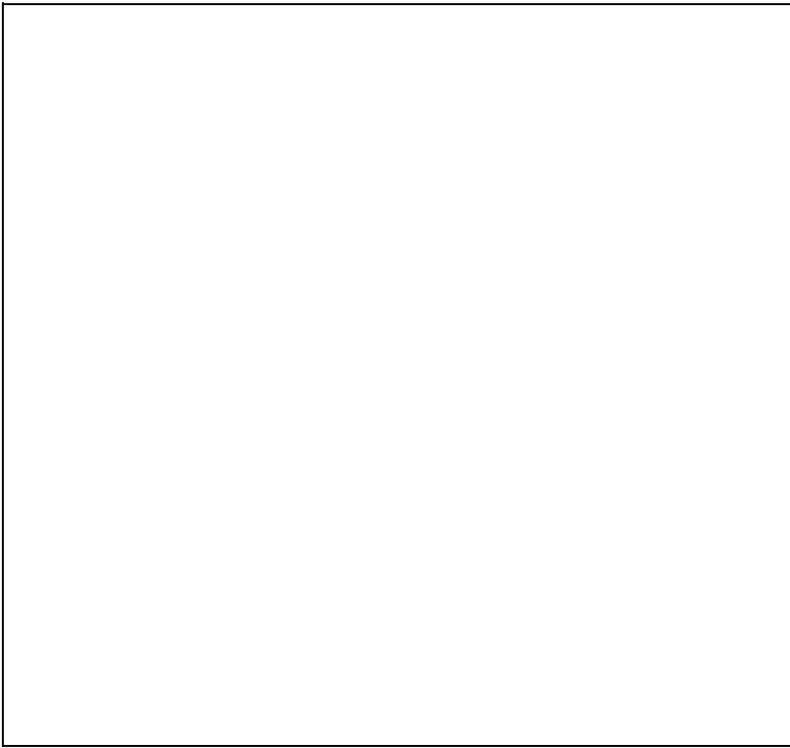


You can print any particular graph by clicking on the print button in the lower right hand corner. The graph will be printed full-page size.

The graph plots the spec values that define an input-output function. In version 10.4 and newer, the input and output values are also plotted directly on the graph, and are updated at the same rate as the underlying document. You can force a data refresh by clicking the refresh button in the graph window.

2.12.9 Adapt Block

Follow the link for Function Code 24 from the FUNCTION CODE INDEX to all the instances of Adapt Blocks in your INFI 90 system. You can easily go from the Adapt Block to the block it adapts (or the other way around), even if they are on different pages. Live monitoring will show the adapted spec in grey on an adapt block. Adapted specs are further marked with a right angle bracket beside the spec (>).

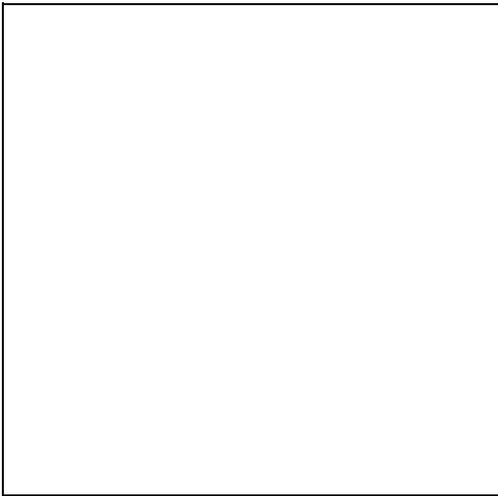


2.12.10 Rung Block Display

Rung blocks are heavily used in some systems: turbine control systems designed by ETSI, for example. When specs are visible, Hyperview displays a ladder diagram that describes the function of the rung block. Live monitoring will show which inputs are open and closed, plus the output.



Forced outputs are shown in Version 9.60 and later.



Hyperview will display 0, 1, H or ? in a red box after the output.

2.12.11 AutoCAD and MicroStation

AutoCAD and MicroStation drawings are useful to have in the DBDOC build. Viewing them in Hyperview is much easier than working with large stacks of sheets. Where the INFI 90 tag name is on the drawing or can be constructed from data in attributes, there will be a direct link to the tag in the CAD/CLD sheet and on graphics. Live information can be displayed on these drawings. Links are also constructed based on a partial text phrase match algorithm that can be activated in the BuildPlus Project Options. We can customize the way we identify tags so if you are not getting the links you want, please contact us.

2.13 Managing Live Loop Connections

2.13.1 About live data settings

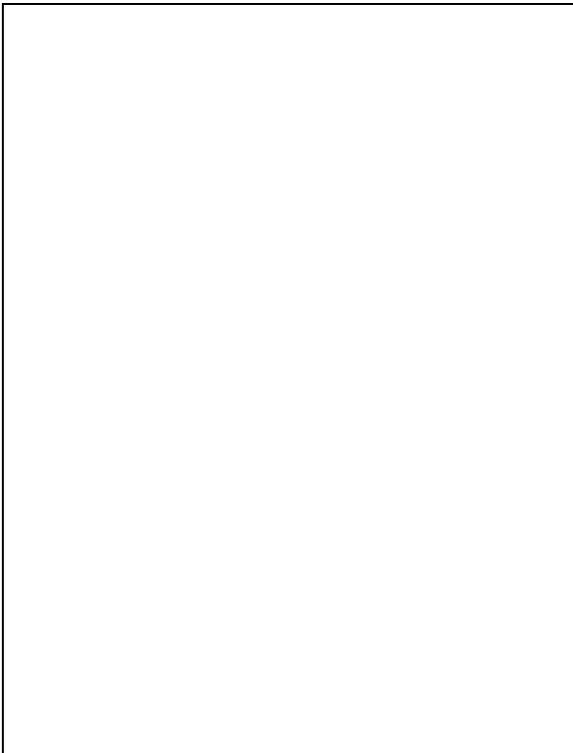
Live data can be displayed on documents in Hyperview and plotted in the Watch Window. This data must be fetched from your system, generally from a CIU. A DBDOC data server called CIUMon talks to both Hyperview and the CIU, and transmits data requests and responses.

Different systems have different configurations for live data. Some have one CIU, and some many. Some have serial port modules available. Some have CIUMon servers running in relay mode located on an intermediate LAN to provide NERC compliant live data to the Business LAN.

Hyperview needs to know what data servers to talk to. Generally this is specified right in the project file (built by BuildPlus during the build process). The live data connections can also be specified from inside Hyperview.

Your live data settings have probably already been configured, built into the project file. If no settings are built into your project file, or you need to change or override the built in settings, you can create custom settings in the **Live Loop Settings** tab in the **Options** dialog.

2.13.2 About the Live Loop Setup dialog



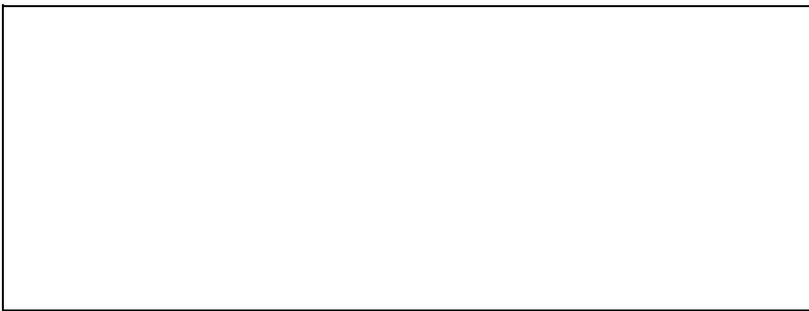
The **Live Loop Setup** dialog is the control center for managing and monitoring your live loop setup. This is where you can see what CIUMons Hyperview is connected to and whether there are any connection issues. Although the current version of Hyperview automatically routes data requests to be best placed CIUMon, it is

also possible to force requests for particular loops or PCUs to be sent to a specified CIUMon. If a serial port module is available for a particular PCU it will be used automatically for requests for that PCU.

In general, your live loop configuration will be determined by your DBDOC administrator and built into your project file. If this configuration has not been defined in this way, or you wish to override it, you can create a custom live loop configuration in this dialog which can include CIUMons of your choice, or force certain CIUMons to be "ignored" by Hyperview.

Note: You can see an icon representing the status of your live loop connections in the **Live Data Status Bar**.

2.13.3 Choosing a settings list



Live loop configuration is handled a bit differently from before starting with Hyperview 10.4. As before, settings can be built directly into the project file by your DBDOC administrator, but now it is possible to group the CIUMons into different lists based on what network context the CIUMon is located in. For example, many sites have one or more CIUMons running on their DCS LAN which are accessible to Hyperviews on the DCS LAN, but not accessible to Hyperviews on the business LAN. Those sites might add a second settings list that included a CIUMon running in relay mode on the business LAN.

You can choose among project (dbdoc file) settings, custom local settings, and demo data.

Use a CIUMon settings list from the project file

The settings lists built into the project file can be seen in the drop-down under **Use a CIUMon settings list from the project file**. Usually there will be only one list, but it is possible to build multiple lists into the project file. Your DBDOC administrator might do this in order to specify different CIUMons to be used in different network contexts. Hyperview will try to choose the correct list for you (based on which CIUMons it can connect to), but you can override this choice manually.

Show demo (fake) data

Use this option to test the visual display of live data when your Hyperview is not hooked up to a CIUMon. You would never use this option under normal circumstances.

Use custom settings

This choice allows you to define a local persistent CIUMon list. Use it if there are no settings built into the project file, or if you want to create different settings. Pressing **Copy from project file** will re-initialize the custom settings to match the currently selected project file settings. If you want to create custom settings that are similar to the project file settings, that's a good place to start.

2.13.4 Understanding your current live loop settings

The list of CIUMon servers that Hyperview is currently connected to (or trying to connect to) are displayed in **CIUMon Settings**. If you select a different server list as described above, a different set of servers will be displayed here. In general this list of servers will have been preconfigured at the build stage, and built into the project file. More than one list may have been built into the project file (typically to support fetching live data in different network contexts). In this case Hyperview will generally have selected the appropriate list automatically.

Each CIUMon server in the list is connected to either a CIU, a Serial Port Module, or another CIUMon server (in relay mode) which is ultimately connected to an actual CIU or SPM.

Unlike in previous versions of Hyperview, it is not necessary to specify in Hyperview which loop a CIUMon server is connected to. CIUMon reports its loop connection directly to Hyperview, and data requests are routed appropriately. The self-reported CIU location is shown in this dialog.



Location and Port

This is the computer and port that a CIUMon server is running on. Hyperview makes a connection to the CIUMon at that location. This is the only information about a CIUMon server that Hyperview needs to know.

Status

This column shows the current status of the CIUMon connection, whether it is connected, not connected, or blocked. If Hyperview is connected to a version 10.4 or later CIUMon, CIUMon will self-report which loop it is connected to, and that information is displayed here. Similarly, if the CIUMon is connected to an SPM to talk to a specific Loop and PCU, the PCU number will be shown here.

Note: If Hyperview is connecting to an older CIUMon (pre 10.4), that CIUMon will not self-report its CIU location. In that case, the CIU location information can be provided manually via **Routing Preferences** (see below). When CIUMon settings are converted to version 10.4, routing preferences are automatically created for unusual cases, and you can also create them manually if needed. This should rarely be necessary.

Local port

This column indicates which local port on the Hyperview computer is being used to make the connection with the CIUMon server. Normally it is chosen automatically from the default port range, but it can be specified when a connection is defined, or a non-default port-range can be specified.

Used for

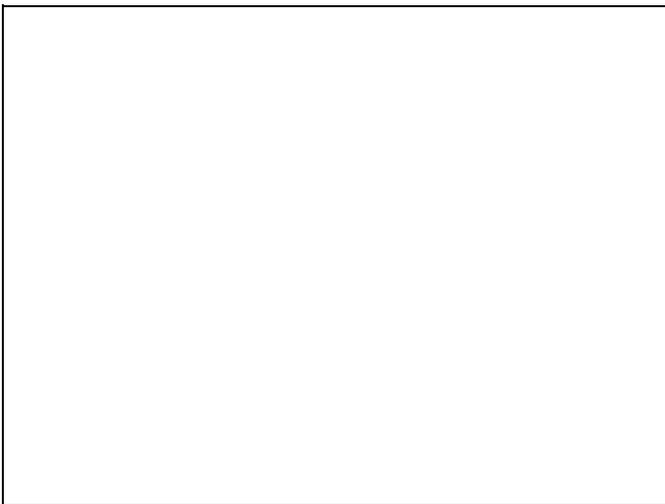
This column shows which loops are having their data fetched via that CIUMon. For example, if "3" is in this column, it means that data requests for Loop 3 are going to that particular CIUMon server.

2.13.5 Adding or modifying live loop connections

2.13.5.1 Adding a connection

To add a new live loop connection to a CIUMon server, press **Add** beneath the server list. The **Add CIUMon** dialog will appear. Specify the location of the CIUMon server Hyperview needs to connect to.

Note: You can only add, edit, and delete servers that are part of a **Custom settings** server list. You can't change a server list that is built into the project file.



Hostname or IP

Specify the name (or IP number) the of the computer on which the CIUMon server you wish to connect to is running.

Note: To indicate that the desired CIUMon is running on the local machine, use **127.0.0.1**. Generally **localhost** will work as well, but this depends on your **hosts** file being configured correctly.

Port

Specify the port that the CIUMon server is listening on. By default this is 5099, but if a CIUMon server on your system has been configured to listen on a different port, you must specify that port here.

Force Hyperview to use a specific local port

Allows you to specify which port will be used locally for communications with the CIUMon server. It is usual to leave this blank, and allow it to be selected automatically from the default port range (or specified port range), but under some circumstances it may need to be set explicitly.

Local port

If you are forcing Hyperview to use a specific local port by checking **Force Hyperview to use a specific local port**, you must specify the port number here. You might do this if your Hyperview can only use one specific port through your firewall.

2.13.5.2 Editing a connection

To edit an existing live loop connection, select the connection in the **CIUMon settings** list and press **Edit**. The **Edit CIUMon** dialog will appear. Make the changes needed and press **OK**.

2.13.5.3 Deleting a connection

To delete an existing live loop connection, select the connection in the **CIUMon settings** list and press **Delete**.

2.13.5.4 Ignoring a connection

To "ignore" an existing live loop connection, select the connection in the **CIUMon settings** list and press **Ignore**. Functionally this is identical to deleting the connection. When routing data requests, Hyperview will act as though that connection did not exist. However, it is temporary, because an ignored connection can be **unignored** and reinstated. If for some reason you want to force Hyperview to never send requests to a particular server, you can "ignore" it.

2.13.6 Specifying a non-default local port range

By default, Hyperview will assign a local port to a CIUMon connect automatically, choosing an available port from within the default port range. Unless you require specific port behavior, for example due to firewall constraints, it is best to stick with this default.

To specify a custom local port range, check **Use custom port range** and then specify the range.

Note: You can only specify a custom port range for a **Custom settings** server list.

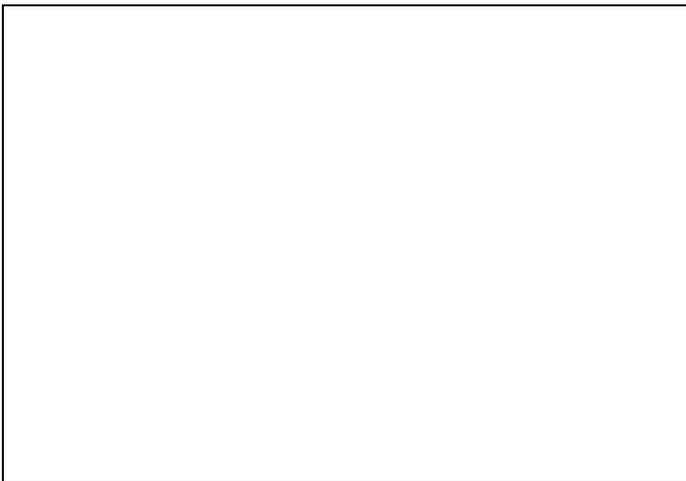
2.13.7 Adding routing preferences

Normally data requests are routing to appropriate servers automatically.

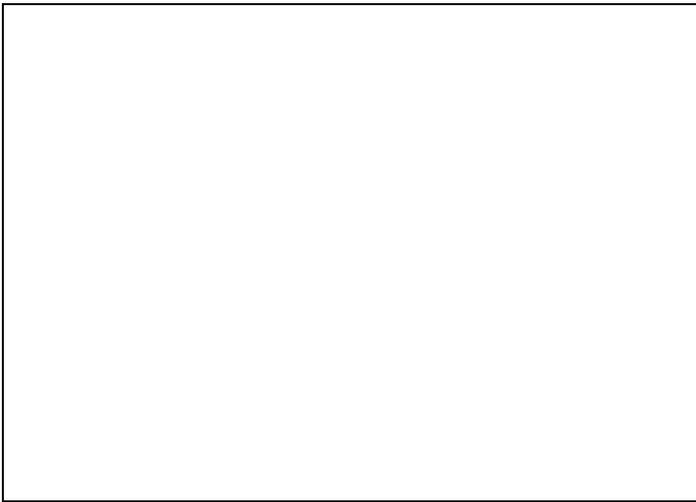
In the event that you want to specify that requests for a particular loop or loop-pcu combination should be routed to a particular server, click on **Routing preferences**.

Note: If Hyperview is connected to a pre-10.4 CIUMon, the CIUMon will not self-report the location of the CIU it is connected to. Use a routing preference to specify this location if needed. The CIUMon will otherwise be assume to be connected to a CIU on Loop 1.

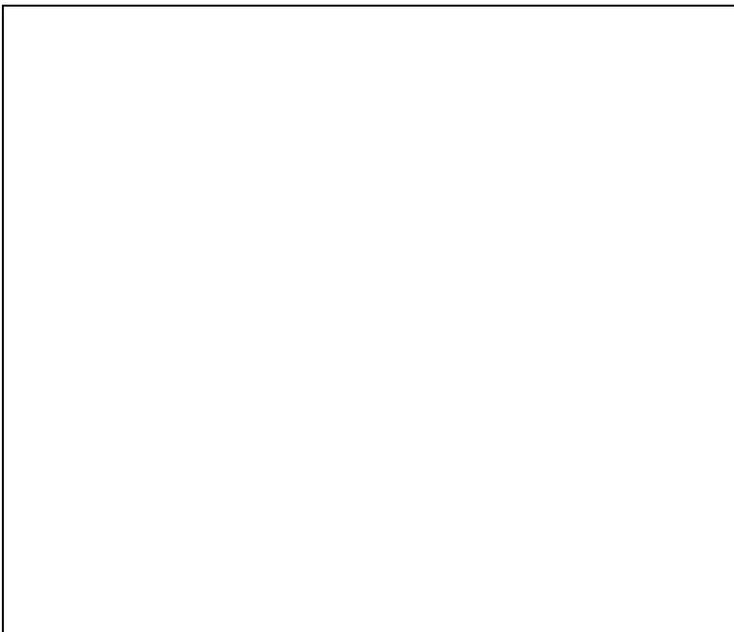
To add a routing preference, click on **Routing preferences** to bring up the **Add preferences** dialog.



Specify the loop (and PCU, if applicable) you wish to make a routing preference for.



Choose the CIUMon server to which data requests for that loop-PCU should be routed.



You can also edit and delete existing routing preferences.

2.13.8 Unavailable data



If data cannot be fetched for a particular loop or PCU, the loop and PCU are displayed here. If live data monitoring is turned on, but there is no CIUMon connection established, there will be items in this box. This could also happen if the only CIUMon available is connected to a serial port module on a certain PCU, and data is being requested for a different PCU.

2.13.9 Status of live data connections

It's easy to see the current status of your live data connections. Just look at the network icon in the Hyperview status bar.

Connection OK (Project Settings)	<input type="checkbox"/>	CIUMon is connected using project file live data settings, communications are happening, user is not blocked
Connection OK (Custom Settings)	<input type="checkbox"/>	CIUMon is connected using local custom settings, communications are happening, user is not blocked
Unable to connect	<input type="checkbox"/>	Cannot communicate with specified connections
No connection specified	<input type="checkbox"/>	No connections have been specified in the Live Loop Setup options dialog
Connection blocked	<input type="checkbox"/>	User is not allowed to collect data on any of the specified connections -- they have been blocked by the DBDOC administrator
Demo settings	<input type="checkbox"/>	Demo data is being displayed, not data from CIUMon

2.13.10 Compression of Hyperview data requests

- Hyperview examines basic Loop, PCU, Module, Block (LPMB) information and combines requests before sending it out to CIUMON.
- If two people are using Hyperview to monitor the same set of points, will CIUMon will only pass on one set of requests to the CIU.

2.14 Managing Live Data Load

2.14.1 Overview of live data bandwidth limitations when fetching live system data

Until recently, the live data fetch rate available in most systems was very low, about 10-20 values per second. DBDOC, along with other system applications, was subject to this very low data bandwidth.

DBDOC 10.4, when used with the RoviSys Turbo, changes this picture. When used in conjunction with the RoviSys Turbo, DBDOC achieves peak data fetch rates of over 100 values per second, and routine data rates of more than 30 values per second when fetching data continuously.

Before DBDOC 10.4, a DBDOC installation might have used a RoviSys Turbo to interface with a Composer CIU, or to multiplex a CIUMon server with the CIU connection of an existing EWS. With DBDOC 10.4 the RoviSys Turbo can still be used for these purposes, but it can also be used with all CIUs (including older ICI01 CIUs) to increase data fetch rates by a factor of five or ten.

Despite this groundbreaking improvement in live data availability, bandwidth is relatively speaking, low. You need to be aware of the very limited live data fetch rate available in most systems, and techniques to avoid overwhelming the limited capability as you use live data display and trending features in Hyperview. When multiple users of Hyperview all fetch live data from the same CIU, they share the bandwidth available, so if that is the case in your system, you need to be doubly aware.

The good news is that the extra bandwidth provided by DBDOC 10.4 and RoviSys Turbo make it much easier to use DBDOC's Watch Window for data trending without overwhelming capabilities of your system.

Read on for critical information about the following:

- How to tell how much data is being fetched by your Hyperview browser.
- How to manage the Watch Window data load in Hyperview.
- How to manage the data load of live data display in Hyperview.

2.14.1.1 Constraints on live data fetch rate

Fast facts:

- A single CIU without RoviSys Turbo gives fewer than 20 values per second to the entire world of DBDOC.
- Multiple CIUs without RoviSys Turbo can give fewer than 20 values per second each.
- A serial port module can give DBDOC as many as 80 values per second from the PCU it is serving, a capability that very few are using.
- The updated RoviSys Turbo uses an optimized fetching and polling strategy, and has been clocked at over 100 values per second in the field, and routine data rates of over 30 values per second. This is a vast improvement, but still quite constrained.

Ways to increase the bandwidth available:

- Use the RoviSys Turbo in conjunction with DBDOC 10.4 in order to increase your data fetch rate by a factor of five or ten.
- Multiple CIUMon processes multiplexed to each of your EWS CIUs will give you more data. Use RoviSys Turbos to multiplex CIUMons to EWS CIUs.
- If you have a small number of PCUs, multiple serial port modules will give you a great deal more data.

CIU setups and how they affect available bandwidth:

- **Single CIU** – A single CIU should probably be on Ring 1, unless it cannot be, or unless the main system is not on Loop 1. Be aware of the extra load of bridging from one loop to another, which has to be done by way of Ring / Loop 1. If you have only one CIU available for DBDOC, you are limited to the base 10-20 values per second, unless you also use the RoviSys Turbo to optimize data fetching. Using this extra software will increase your data rate to 30-100 values per second.
- **Multiple CIUs** – You may have more than one EWS, and each might be on a different Loop. It is easy to take advantage of DBDOC CIUMon and RoviSys Turbo. We encourage you to multiplex any EWS CIUs you have. For 10% to 20% of the cost of a CIU, you can have its spare bandwidth available for DBDOC. It is simple to implement and easy to reverse or suspend.
- **Serial Port Modules** – The Serial Port Modules are of immense value when used with DBDOC. Each is capable of giving up to 80 values per second from the modules in a PCU. Of course, you have to be satisfied that this does not load the PCU too much, but being on top of that is a manageable problem. As all the data from any PCU with its own Serial Port Module is offloaded from your CIUs, the system directly gains throughput, too.

2.14.2 Managing live data load

2.14.2.1 Managing data load for live data on documents

Tips for minimizing your bandwidth needs:

- **Slow** mode should be used (not **Continuous** mode) unless continuous data is necessary.

- Be aware of the **Send/s** and **Recv/s** fields in the **Live Data Status Bar** at the bottom of the Hyperview browser. These indicate how many data requests Hyperview is making per second, and how many data updates it is receiving.
- Be aware that unlike live data on documents, the Watch Window collects data even if it is closed or hidden.

If you use Continuous mode live data display on documents:

- On documents, live data is fetched for all visible points. If you are zoomed out on a graphic, and monitoring in **Continuous** mode, this can place a heavy load on the system. **Zoom in** on graphics.
- Better yet, avoid **Continuous** mode live data on graphics – save it for CAD/CLD diagrams!
- Make sure the default update interval for live loop annotations is no smaller than it needs to be. The default update interval can be specified in the **Options** menu on the Hyperview browser.
- Be aware that your data collection activities affect the bandwidth available to other Hyperview users sharing the same CIU connection.

2.14.2.2 Managing live data load for blocks in the Watch Window

The Watch Window is easy to use, and easy to abuse. Be aware of how much data you are asking for, and be especially careful when the Watch Window is closed. Closing Hyperview stops data collection in the Watch Window but when Hyperview is reopened it starts data collection again without necessarily opening the Watch Window.

Tips for minimizing your bandwidth needs for live data collected by the Watch Window:

- Don't collect data you don't need. **Pause data collection** by clicking on the green clock for blocks you are not actively trending.



- Be aware of the **Send/s** and **Recv/s** fields in the **Live Data Status Bar** at the bottom of the Hyperview browser. These indicate how many data requests Hyperview is making per second, and how many data updates it is receiving.
- Be aware that the Watch Window will collect data even if it is not open.



- The default update interval for Watch Window blocks is 10 seconds. **Increase the update interval** if you don't need that level of detail in your data. You can change the update interval by editing this field in the block list to the right of the plots: .

2.14.3 How to tell how much data is being requested (Live Data Status Bar)



In Hyperview you are now able to tell how much live data a particular Hyperview is requesting per second. This information in the lower right-hand corner of the Hyperview browser is visible at all times in the status bar at the bottom of the Browser, and shows you exactly what data fetch load you are placing on the system.

The **Send/s** field shows how many requests Hyperview is making per second, from all sources (Watch Window, Live Loop Annotations, and live data directly on documents), while the **Recv/s** field shows how much data is actually being received from the CIU or serial port module. If the send rate is approaching the system data rate constraint (10-20 values per second per CIU), you need to take steps to reduce the data load in order to avoid performance degradation.

Keep in mind that the load from all Hyperview users has to be added together to really know what you can get from your CIU or CIUs.

For a global picture of data load, including requests from all Hyperviews and any other applications requesting data from CIUMon, you can view the CIUMon statistics. Using CIUMon options, or using CIUMonController.exe for CIUMon installed as a service, you can turn on statistics showing who is asking for data, and how much data they are trying to get. This can help a lot in sorting out the load on the system.

2.15 Hyperview Keyboard Shortcuts

Display	Topic display	Pane display		
	T (or A)	Toggle attributes/tag names	J	Move window divider
	S	Toggle specs	F	Toggle full screen mode
	V	Toggle Vegas Lights		
	Ctrl+E	Toggle error markers	Document rotation (PDF, AutoCAD, MicroStation)	
	Ctrl+Tab	Highlight hotspots	R	Rotate counterclockwise
	F5	Refresh	Shift+R	Rotate clockwise
	Esc	Abort <i>Draw</i> Operation		

Navigation	Changing location	Zoom		
	Backspace	Back	+	Zoom in
	Shift+Backspace	Forward	-	Zoom out
	Alt+Home	Home	Z	Toggle zoom
	B	Next topic in group	Table of contents	
	Shift+B	Previous topic in group	Y	Synch table of contents
	L	Next index entry	C	Top level table of contents
	Shift+L	Previous index entry	Ctrl+T	Toggle Treeview mode table of contents
	N	Next topic with search hits	Projects and windows	
	Shift+N	Previous topic with search hits	Ctrl+O	Open Project File
	H	Next search result in topic	Ctrl+N	New window
	Shift+H	Previous search result in topic		
	Ctrl+G	Find coordinates		
		Focus		
	F6	Move focus to next pane/window		

Tools	Tool windows	Search		
	W	Watch Window	Ctrl+F	Full text search
	E	Error Browser	Alt+D	Database search
	X	Cross Reference Window	Alt+T	Topic title search
	F1	Help	Printing	
	On-document live data	Ctrl+P	Print	
	M	Turn on or refresh snapshot data	P	Print with last settings

Shift+M Turn off snapshot data

Double M Turn off snapshot data

Bookmarks

Ctrl+D Define bookmark

The clipboard

Ctrl+A Select all

Ctrl+C Copy

Ctrl+V Paste

3 Guide to Hyperview in a Browser

This User Guide is only partially complete. Chapters after **Managing User Accounts** still to be included.

3.1 Hyperview in a Browser

3.1.1 Overview

Hyperview in a Browser is a new version of Hyperview that allows you to access basic Hyperview functionality in a web browser. This offers additional flexibility and device compatibility for using DBDOC's project and live data viewer.

For example with a tablet on your wifi, you could use Hyperview in a Browser to:

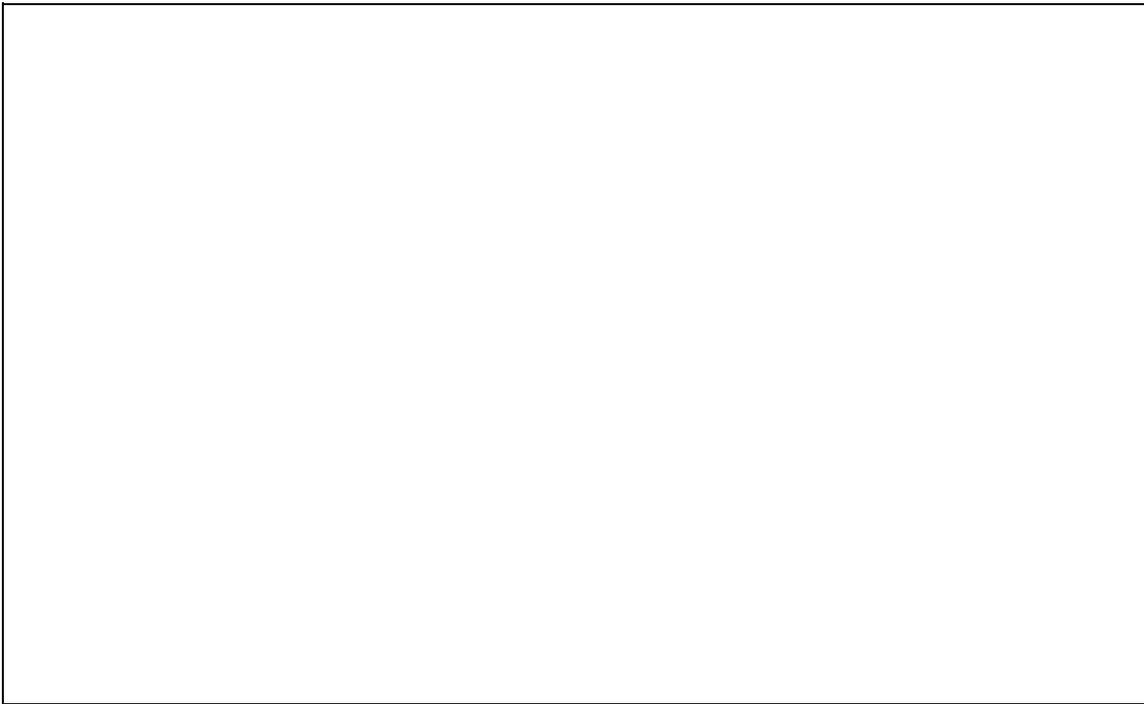
- Monitor configuration and values during commissioning and checkout.
- As a mobile maintenance and troubleshooting tool, verifying the field functionality against the running configuration.
- During installation checkout, to verify wiring against drawings on the tablet.

While a central Hyperview "install" can be accessed via a Windows network share, the web browser interface is touch-friendly interface and suitable for tablets, for systems whose execution or user environment specifics make web interfaces easier (including non-Windows systems), to integrate with an existing webportal, or some scenario we haven't thought of yet.

3.1.2 Platform/Browser compatibility

Hyperview in a Browser has been tested in Edge, Chrome, Safari and Firefox, and should work in versions of these browsers dating back to October 2017. In order to use **Hyperview in a Browser**, you just go to a certain website (and, if logins are required, log into your **Hyperview in a Browser** account). A subset of **Hyperview** functionality is supported in **Hyperview in a Browser**, including basic navigation, live data, full text and title search, and more.

Below you can see **Hyperview in a Browser** running in an Edge web browser.



3.1.3 Setting up a Hyperview Service

In order to use Hyperview in a Browser, you must use the Hyperview Service Controller to start a Hyperview Service.

3.1.4 Managing user accounts

By default, no login account is used, but you can set up accounts to restrict access and to save per-user settings. See our documentation on making and managing Hyperview in a Browser accounts.

3.2 Setting up a Hyperview Service

3.2.1 Overview

As of DBDOC 11.3, it is possible to enjoy basic Hyperview functionality in a standard web browser. How it works is that the Hyperview application (hyperview.exe) can be run as a service, and in this mode, it runs an embedded web server. This "Hyperview Service" loads a project file, and serves it as a web application. From the point of view of the user with the web browser, Hyperview is a website.

A new application, **Hyperview Service Controller** is required to run Hyperview as a service. Using Hyperview Service Controller, you can specify which Hyperview executable to run as a service, which project file to serve, what port to serve the project file on, and various other configuration details. Note that you must have administrator privileges in order to run services with Hyperview Service Controller.

The Hyperview website is only available on a specified port of a specified machine if suitable firewall accommodations have been made. Generally it would only be available inside a network, not to the outside world.

Running the Hyperview Service Controller

In order to create a Hyperview Service, you must run the **Hyperview Service Controller**. The Hyperview Service Controller can be found in the Programs folder of a DBDOC 11.3 or later installation (HyperviewServiceController.exe). Running Hyperview Service Controller with full functionality requires administrator privileges.

Configuring your Hyperview Services setup

Before you can create a Hyperview Service, you must set up a few things that will apply to all the services. Press **Setup** on the main interface of the Hyperview Service Controller. This brings up the **Configure Hyperview Services** dialog.

Adding a new Hyperview Service

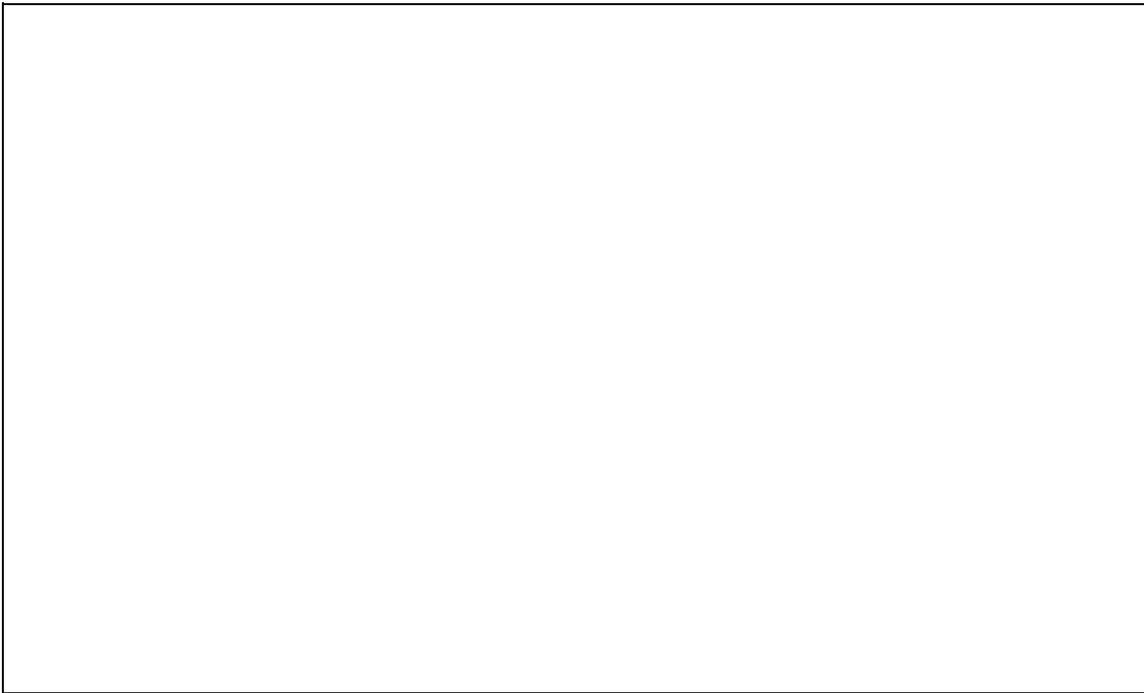
To add a Hyperview Service, press **Add Service...** in the main window of the Hyperview Service Controller. This will bring up the **Add a Hyperview Service** dialog.

3.2.2 Running the Hyperview Service Controller

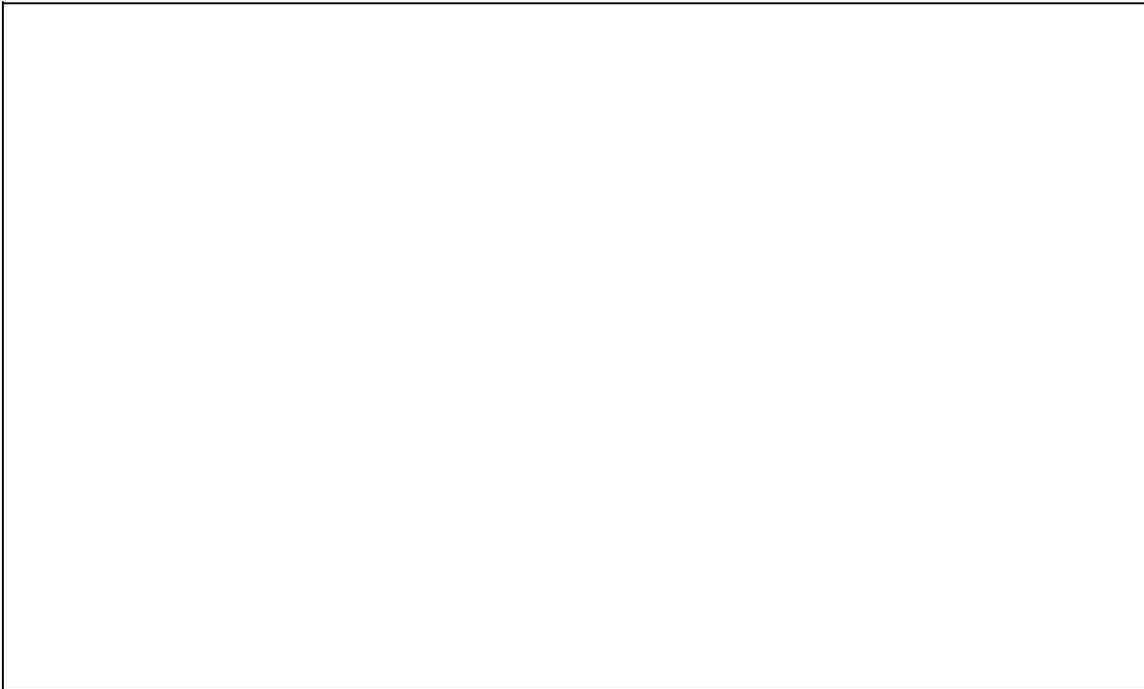
The **Hyperview Service Controller** can be found in the Programs folder of a DBDOC 11.3 or later installation. It is called **HyperviewServiceController.exe**. Running Hyperview Service Controller with full functionality requires administrator privileges.

Note: Hyperview Services should be run on the same machine as the project files they are serving. The Hyperview Service Controller will need to be run on this same machine too. This may mean that you run Hyperview Service Controller on your build machine, where there is local access to project files, or it may mean that you run it on the machine with the shared folder that your project files are copied to.

When the Hyperview Service Controller is first started, it will look like this:



When services have been defined, they are listed in on the left.



You can create new services, modify existing services, start and stop services, and delete services. By default, once created, services will automatically restart when the machine they are on is rebooted, so you only need to start a service once. Each Hyperview Service will serve exactly one project file, which you must specify. You can, however, specify the project file name with wildcards in the same manner as on the command line, in order to server the most recent build of that project.

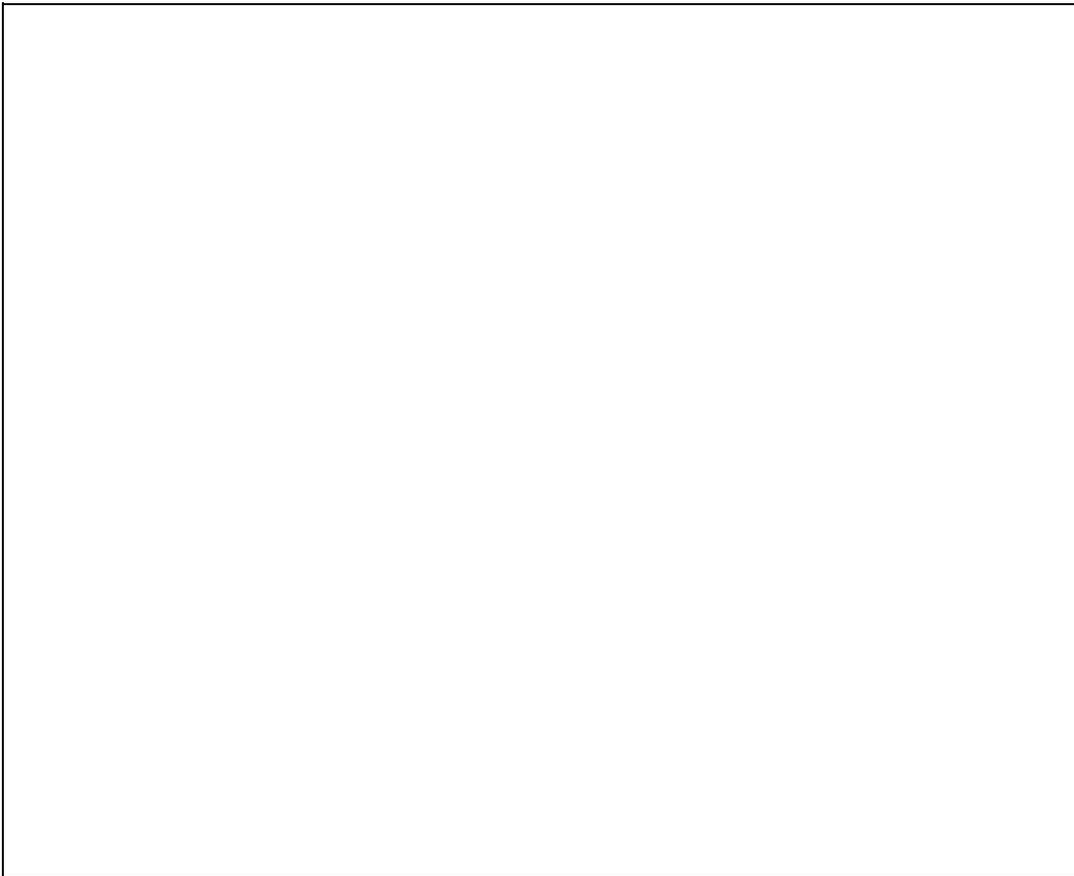
Note: Even if you use "*" to denote the most recent build, the Hyperview Service must still be restarted in order to actually load the newest project file. It will not automatically detect that a new matching project file has appeared.

- **Hyperview Services:** A list of currently configured Hyperview Services.
- **Add Service...:** Opens a dialog for creating a new Hyperview Service.
- **Modify Service...:** Opens a dialog for modifying the currently selected service in the **Hyperview Services** list.
- **Delete Service:** Deletes the currently selected service.
- **Current service settings:** Shows configuration details for the currently selected service in the **Hyperview Services** list.
- **Modify Logging Options...:** Opens a dialog for configuring logging settings for the currently selected service.
- **Start Service:** Starts the currently selected service.
- **Stop Service:** Stops the currently selected service.
- **Start All Services:** Starts all the services. If all the services are already running, this button will be disabled.
- **Stop All Services:** Stops all the services. If all the services are already stopped, this button will be disabled.
- **Launch Website:** Launches a browser showing the Hyperview website for the currently selected service.
- **About:** Shows version information for this Hyperview Service Controller.
- **Setup:** Opens the **Configure Hyperview Services** dialog for general setup that pertains to all services.
- **Explore to Log Files:** Opens the folder containing Hyperview Service Controller and Hyperview Service logs.
- **Refresh:** Refreshes the **Hyperview Services** list.

3.2.3 Configuring your Hyperview Services setup

3.2.3.1 General setup for all Hyperview Services

Before you can create a Hyperview Service, you must set up a few things. Press **Setup** on the main interface of the **Hyperview Service Controller**. The following dialog appears:



3.2.3.1.1 Hyperview server setup

The following two items need to be set appropriately in order to create Hyperview Services.

- **Shawl Executable:** This is a wrapper application that allows Desktop Hyperview to run as a service. It should be located in your DBDOC Programs folder. Make sure that the proper path to shawl.exe is specified here.
- **Hyperview Server Port Range:** This is where you specify the ports that a Hyperview Service can use. You can choose one port, or a range of ports. Each service can only serve one project file on one port, so if you want to serve more than one project file simultaneously, more than one port should be specified in the range here.

Note: The ports specified here constrain where **Hyperview Service Controller** can put Hyperview services. However, you will additionally need to make appropriate holes in your machine's firewall for the ports specified here -- this is not done automatically.

3.2.3.1.2 HTTPS Settings

If you would like to serve your Hyperview Service over HTTPS you can configure the service controller to do so here. You can either specify your own certificate and key files, or use the option to generate self-signed ones.

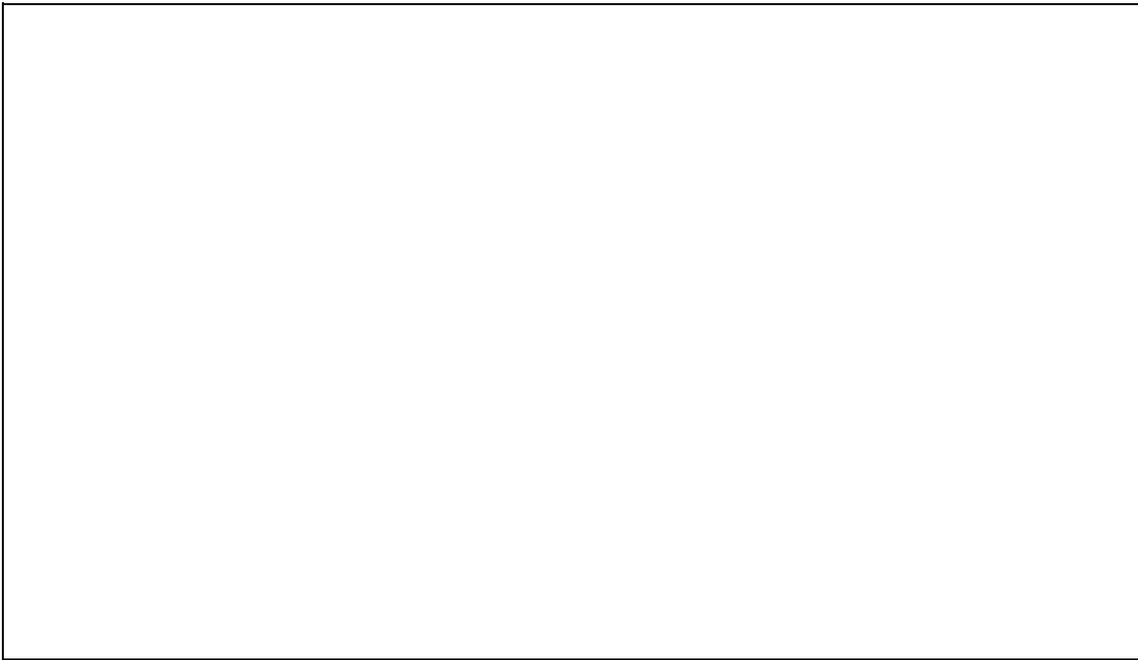
- **Private HTTP Port Range:** This is where you specify the ports that a Hyperview Service can use when HTTPS is enabled.
- **FQDN:** This lets you choose a domain name other than the default. The default domain name will be used if the field is left empty, and is shown below.
- **Certificate File:** This is where you must specify the location of your certificate file if you choose to use one.
- **Key File:** This is where you must specify the location of your key file if you choose to use one.
- **Generate self-signed certificates:** Use this option if you want to skip using your own certificate and key files and have them generated for you. When this option is checked, the certificate and key fields will be disabled.
- **Start Caddy:** This button starts the Caddy service for HTTPS. If Caddy is already running, it will be disabled.
- **Stop Caddy:** This button stops the Caddy service for HTTPS. If Caddy is not currently running, it will be disabled.

3.2.3.1.3 Other setup information

- **Directory for log files:** This is where you will find Hyperview Service log files. You can specify a different location if you wish.
- **Explore to Hyperview UserData Folder:** When Hyperview is running as a service, it stores its datafiles in this location. This location is analogous to the user data folder used by Desktop Hyperview (usually `C:\Users\\AppData\Roaming\GMCL\DBDOC` for desktop users), but stores information for all the users connected to a particular Hyperview Service.
- **Reset Admin Password:** Revert the Administrator password to its installation setting. When you view the Hyperview website (**Hyperview in a Browser**), there will be a login. There is a preset Administrator account with an installation password. Once logged in, the Administrator user will be able to change the Administrator password, and create other user accounts via the **Hyperview in a Browser** interface. If the Administrator password is lost, it can be reset to the initial installation password here. Please contact GMCL for the reset installation Administrator password.

3.2.4 Adding or modifying a Hyperview Service

In order to add a new Hyperview Service, press **Add Service...** on the main interface of the **Hyperview Service Controller**. The following dialog appears:



In order to add a service, you must specify a project file to serve, a Hyperview executable to run as a service, and the port on which the project file is to be served by the service.

- **Service Name:** This is the name that will be given to the service. You will be able to see it in (for example), the Windows **Services** utility. The name is automatically generated from the project filename and port that you choose.
- **DBDOC File to Serve:** This is the project file that is to be served by the Hyperview Service. Note that in general, the project file must be on the same machine as the Hyperview Service (i.e. the machine you are currently on). There are usually restrictions on allowing the **Local System** user (the virtual user that runs services) from accessing network filesystems. Moreover, for best performance, it makes sense to have only one network hop (the one between Hyperview in a Browser and the Hyperview Service) in the system.
- **Use Most Recent:** If you check this box, this service will serve the most recent DBDOC build for the specified project. You should check this box if you build new project files regularly, and want to be serving the most recent one. Note that the Hyperview Service will need to be restarted in order to load up the newest project file. There is a command line interface for doing this, if you wish to incorporate it into your regular build process.
- **Hyperview Executable:** This is the hyperview executable that will be acting as a service. A hyperview.exe from at least DBDOC 11.3 is required.
- **Service Startup Type:** These are the options as to how the service should be handled when the machine restarts. The default, **Automatic**, causes the service to be automatically restarted if the machine reboots. **Manual** would mean that the service would not be automatically restarted -- you would need to restart it explicitly from the Hyperview Service Controller. **Disabled** means that the service cannot be started. There should be little reason to choose anything but **Automatic** here.
- **No Login Required:** If this is checked (the default), then there is no login required to view a Hyperview page served by the Hyperview service. An account based on IP is automatically created, and used to keep track of things like whether Specs are shown, which are particular to each user. This **No Login Required** mode is convenient, because it allows links to particular Hyperview "web

pages" to be shared and used generally within a particular network context. If **No Login Required** is not checked, then users must log in with username and password in order to view Hyperview pages.

In order to modify an existing Hyperview Service, select a service and press **Modify Service...** on the main interface of the Hyperview Service Controller. After the service is modified, it will automatically be restarted so that any changes take effect.

3.2.5 Starting and stopping Hyperview Services from the command line

Although generally Hyperview Services are managed with the Hyperview Service Controller, it is possible to start and stop Hyperview Services from the command line. This would be useful if you had a Hyperview Service configured to load and serve a "most recent" project file. If a new matching file appeared, the Hyperview Service would not notice. It would need to be stopped and restarted in order to load the updated build.

With the command line interface, you can include a script in your build routine that automatically restarts some or all of your Hyperview Services, causing them to load the most recent project file, if they were configured that way.

3.2.5.1 Command Line

To start or stop Hyperview Services, use this command:

```
"C:\Program Files (x86)\GMCL\DBDOC\Programs\HyperviewServiceController.exe" /servicecmd=[cmd<:port>]
```

where **cmd** can have the following values:

- **start**: Starts all the Hyperview Services that have been created. If a service is already running, has no effect.
- **start:N**: Starts the Hyperview Service running on port N (assuming it exists).
- **stop**: Stops all the Hyperview Services that have been created. If a service is already stopped, has no effect. Note that stopping a service may not be instantaneous. It may take 30 seconds or so.
- **stop:N**: Stops the Hyperview Service running on port N (assuming it exists).

You must run the script with Administrator privilege in order to be able to control Hyperview Services in this way.

3.2.5.2 Example

For example, if you wanted to automatically stop and restart all your Hyperview Services, a sequence of commands like the following could be used.

```
"C:\Program Files (x86)\GMCL\DBDOC\Programs\HyperviewServiceController.exe" /servicecmd=stop  
[WAIT 30 SECONDS]  
"C:\Program Files (x86)\GMCL\DBDOC\Programs\HyperviewServiceController.exe" /servicecmd=start
```

Or perhaps only one service on a known port requires this treatment. If the service question was running on port 8001 (run the Hyperview Service Controller to find the port), these commands could be used.

```
"C:\Program Files (x86)\GMCL\DBDOC\Programs\HyperviewServiceController.exe" /servicecmd=stop:8001  
[WAIT 30 SECONDS]  
"C:\Program Files (x86)\GMCL\DBDOC\Programs\HyperviewServiceController.exe" /servicecmd=start:8001
```

3.2.6 Accessing Hyperview Services through a firewall

As of DBDOC 11.3, you can run Hyperview as a service. This Hyperview Service behaves as a web server, and serves a specified project file on a specified port. This port will not be accessible to web browsers on other machines unless it is unblocked for TCP communication. For example, if your Hyperview Service is running on port 8001, you would need to make sure there was a hole in the firewall for port 8001.

Hyperview Services access live data in exactly the same manner as desktop Hyperview, i.e. by connecting to a CIUMon. The CIUMon that a Hyperview Service will connect to must be specified in the project file that it is serving, and must be accessible to the machine that it is running on.

3.2.7 Managing Hyperview Service log files

3.2.7.1 Log file location

In the **Hyperview Service Controller**, click on **Explore to Log Files** to explore to the folder where log files for Hyperview Service Controller, and also for individual Hyperview Services are kept.

You can change this location in the **Configure Hyperview Services** dialog.

3.2.7.2 Log file names

The Hyperview Service Controller log is called

```
HyperviewServiceController_<date>.log.
```

Hyperview Service logs are given a unique name based on the project the service is service, the port it is

serving it on, and the date and time:

```
<project>_<port>_HyperviewServer_<date>-
```

For example,

```
Test project_8888_HyperviewServer_2022-01-17-16-20-26.log
```

3.2.7.3 Logging options

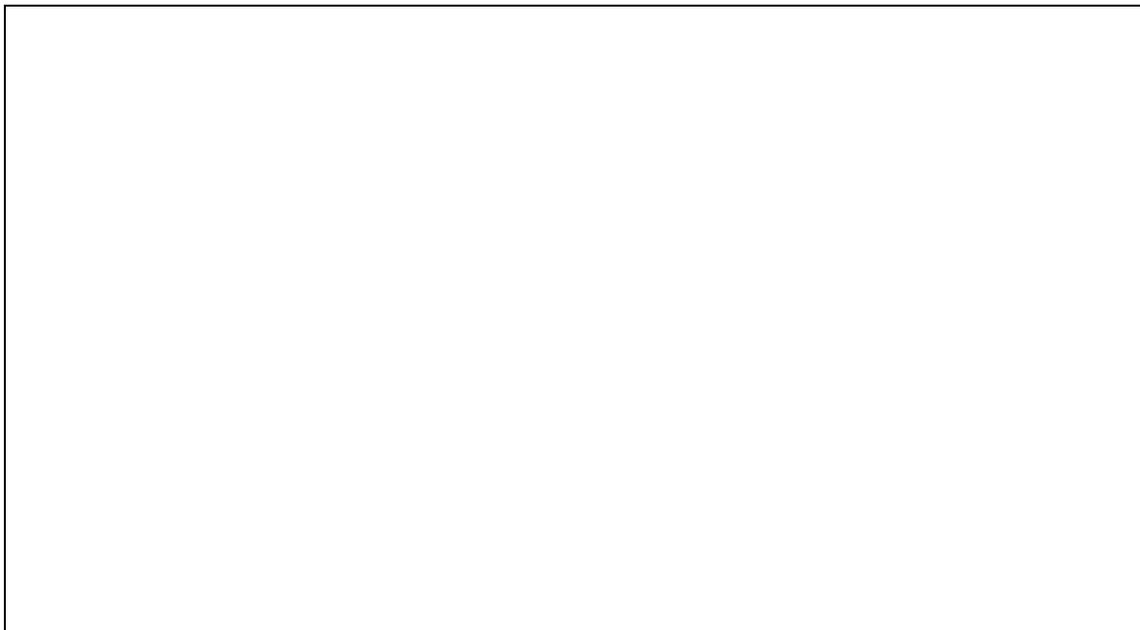
3.2.7.3.1 Hyperview Service Controller logs

The log for the Hyperview Service Controller has no optional settings. One new log starts on each day that Hyperview Service Controller runs.

3.2.7.3.2 Hyperview Service logs

You can choose logging options for Hyperview Services by clicking on **Modify Logging Options...** in the main Hyperview Service Controller window, or in the dialog for adding or modifying a Hyperview Service. The options shown will apply to the currently selected Hyperview Service (in the **Hyperview Services** list).

The options for each service are specified individually. So, for example, you could have a debugging Hyperview Service that logged all data communications, and a regular service that did not.



Log Contents

By default, most Hyperview Service activities, such as logins, page requests, making and breaking live data connections, and errors are logged.

Optional logging

- **Detailed live data communications:** Check this box if you want all data communications between a Hyperview Service and CIUMon to be logged. By default, only connection status change and periodic status updates are logged.

Log Management You have several options about when new Hyperview Service logs should be started.

Start a new log

- **When the current log is full and on service restart:** This mode will only create a new log file when the Hyperview Service is restarted, unless a maximum size has been reached, in which case a new log file will be created sooner.
 - ◆ **Restrict log files to N MB each:** Specify the maximum log file size here.
- **On service restart only (no size limit):** A new log file will be created only when a Hyperview Service is restarted. This is probably not a mode to use if **Detailed live data communications** is checked.
- **Daily and on service restart (no size limit):** A new log file will be created every day, as well as when a Hyperview Service is restarted.

Log File Retention

- **Keep up to N log files:** Specify how many log files to keep. Especially if a new log file is being started daily, you may want to prevent hundreds of files from being created over time. If the limit is exceeded, the oldest log file will be automatically deleted.
- **Keep all log files:** If this is specified, no log files will be automatically deleted.

Live Data Options

- **Force Snapshot Mode Live Data:** Checking this option disables continuous mode (clock) for live data, and only allows Web Browser Hyperview to use Snapshot/Slow mode (camera) for live data. The clock icon on the toolbar will be greyed out, indicating that this option is inactive and continuous data is turned off.

3.2.8 Finding Hyperview Service user data files

Like desktop Hyperview, Hyperview Services keep a number of data files. You can explore to them by clicking on **Explore to Hyperview UserData folder** in the Configure Hyperview Services dialog. In general there should be no call to look at these files.

The exact folder where the data files are located depends on the operating system, but on Windows 10, for

example, it is

```
C:\Windows\SysWOW64\config\systemprofile\AppData\Roaming\GMCL\DBDOC
```

In this folder are data files associated with Hyperview Services on this machine. For example, the users.db file contains account information for all the user accounts for the Hyperview Services on this machine.

Information related to individual users of a Hyperview Service are located in the DBDOC\UserData folder, in <username> subfolders. For example, the home page for each user and project is stored there.

3.3 Using Hyperview in a Browser

Once a Hyperview Service is running, using Hyperview in a web browser is just a matter of going to the Hyperview website. Once you have logged in, you can browse a dbdoc file using an interface very similar to that of desktop Hyperview. You can search the dbdoc file, show live data, and make bookmarks. The Administrator user can create new accounts, and all users can change their passwords.

3.3.1 Hyperview website location

To view the Hyperview website, visit

```
http://<host>:<port>
```

where <host> is the machine that is running the Hyperview Service, and <port> is the port that it is using. For example,

```
http://myhost.internal:8001
```

To see what Hyperview Services are running on what ports, run the Hyperview Service Controller (requires Administrator privilege). You can also launch the Hyperview website from the Hyperview Service Controller interface.

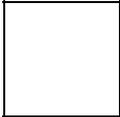
3.3.2 Hyperview in a Browser functionality

Hyperview in a Browser supports a subset of desktop Hyperview functionality, including basic navigation, searching, and live data. This documentation will provide you with detailed information about all of Hyperview in a Browser's capabilities.

3.3.3 Basic functionality

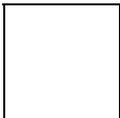
Hyperview in a Browser supports much of the same functionality as **Desktop Hyperview**. The following chapters cover some of the basic functionality.

3.3.3.1 Displaying the main Table of contents

On toolbar, click the  button.

This brings up the top level table of contents for the project file.

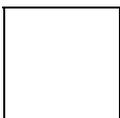
3.3.3.2 Synchronizing the index pane to the pane with focus

On toolbar, click the  button.

Every document and index in Hyperview is listed in some index in the project. **Synch Table of Contents** causes this index to be displayed for the topic which currently has focus (shown with a magenta outline).

Usually the topic you are synchronizing will be in the content pane, and the index will appear in the index pane, but a sub-index in the index pane can also be synchronized. In this case, the index one of whose links is the name of the subindex will be shown in the index pane, replacing the original sub-index.

3.3.3.3 Opening the menu

On toolbar, click the  button.

Opens the main browser sidebar menu. Click on the menu button again to close it.

3.3.3.4 Creating and managing bookmarks

On the main menu, click **Bookmarks**.

A panel appears that lets you configure add, modify, and delete bookmarks. Click on any bookmark in the **Bookmarks**: table to display it in the browser. You can also "export" the bookmark by clicking **Show in Tab**. This will open a new tab with the URL to the bookmark in question.

Bookmark URLs can be copied to other browsers and machines, included in emails etc, and provide a convenient way to communicate a particular location in a DBDOC file.

A bookmark can be made as a "System" bookmark so that other users will be able to access and use it. See Defining bookmarks centrally with System bookmarks.



- **Add Bookmark**: Add the current browser contents as a bookmark.
- **Name**: Specify the name for the bookmark to be added.
- **Bookmarks**: Shows a list of all the bookmarks that have been added. Click on a bookmark entry to display it in the browser.
- **Show in Tab**: Opens a new tab with the contents of the currently bookmark.
- **Edit**: Modify the name of the currently selected bookmark. A text editor will open in place, in the table.
- **Delete**: Delete the currently selected bookmark.

3.3.3.5 Getting online help

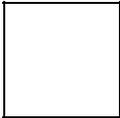
On the **Help** Menu, click **Help Website**.

Opens the main Help page on the GMCL website in another tab.

3.3.4 Navigation and signal tracing

Hyperview in a Browser supports much of the same functionality as **Desktop Hyperview**. The following chapters cover navigation and browsing related topics.

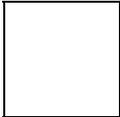
3.3.4.1 Browsing to the previous topic in a topic group

On toolbar, click the  button (shortcut **Shift+B**).

Browses to the previous topic in the chapter containing the current document.

See Browsing to previous topic in topic group in the Desktop Hyperview documentation for more information.

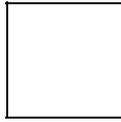
3.3.4.2 Browsing to the next topic in a topic group

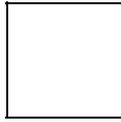
On toolbar, click the  button (shortcut **B**).

Browses to the next topic in the chapter containing the current document.

See Browsing to next topic in topic group in the Desktop Hyperview documentation for more information.

3.3.4.3 Browsing to the previous entry in an index



On toolbar, click the  button (shortcut **Shift+L**).

Browses to the previous link in the index pane.

See Linking to the previous index entry in the Desktop Hyperview documentation for more information.

3.3.4.4 Browsing to the next entry in an index



On toolbar, click the  button (shortcut **L**).

Browses to the next link in the index pane.

See Linking to the next index entry in the Desktop Hyperview documentation for more information.

3.3.5 Zooming and scrolling

Hyperview in a Browser supports much of the same functionality as **Desktop Hyperview**. The following chapters cover functionality related to scrolling panes and zooming in and out on them to view their content.

3.3.5.1 Zooming on a graphical topic

On graphical topics (graphics, PDFs, CAD sheets, etc), use the scroll wheel or a pinch gesture to zoom the content pane in and out.

3.3.5.2 Scrolling and panning

Text documents in **Browser Hyperview** can be scrolled like normal webpages using the scroll wheel, scrollbars, or a scrolling gesture.

Graphical documents in the content pane can be panned by dragging them and do not have scrollbars.

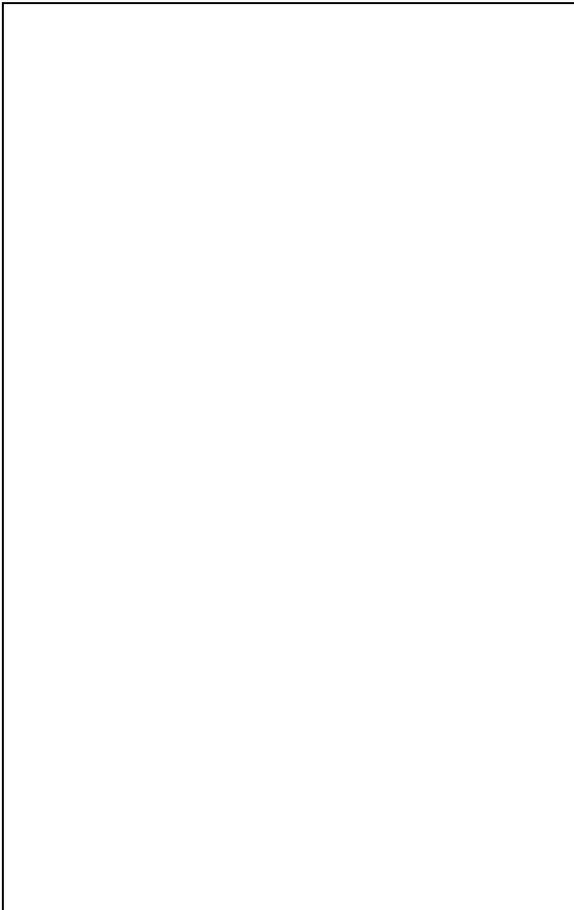
3.3.6 Searching

Hyperview in a Browser supports much of the same functionality as **Desktop Hyperview**. The following chapters cover searching. Currently text search and title search are supported.

3.3.6.1 Searching for a word or phrase

On the **Search** Menu, click **Full Text Search**, or click the  button on the toolbar.

3.3.6.1.1 Overview of Full Text Search



Search for an arbitrary word or phrase, including wildcards.

3.3.6.1.2 Search options

The search specifics are virtually identical to those in desktop Hyperview. Please see Searching the system for a word or phrase for full details on the various search options, as well as information on search operators, wildcards, and handling of non-alphanumeric characters.

3.3.6.1.3 Search results

When you search for a word or phrase, all documents containing the word or phrase are listed in a **Search Results** pane (see Displaying the results of the last search). Within each document, all occurrences of the word or phrase are highlighted to stand out. Use the **Next Highlight** button and the **Previous Highlight** button on the tool bar to step through all of the search hits in any document.

3.3.6.2 Searching topic titles

On the **Search** Menu, click **Topic Title Search**.

A panel appears that lets you search topics by title. Just start typing, and all topics the beginnings of whose titles match what has been typed will show up in the results.



- **Type a word:** Enter a keyword to be used to be matched to the beginning of document title keywords. As you enter characters, the **Topic Title** text box displays the title that most closely matches the characters typed so far. Click on any result to display it in the browser panes.

3.3.6.3 Clearing search highlights

On the **Search** Menu, click **Clear Search Results** to remove all highlight marks from search hits. This will also clear the saved search results.

3.3.7 Showing live data

3.3.7.1 Overview

Live data in Hyperview in a Browser is an extension of desktop Hyperview live data. The Hyperview Service, which is just a Hyperview application running as a web server, is what communicates directly with CIUMon to fetch live data. When Hyperview in a Browser requests live data, the request goes to Hyperview Service, which passes it on to CIUMon. From CIUMon's perspective, Hyperview Service is the same as any other Hyperview.

It is thus necessary that Hyperview Service be able to connect to CIUMon, but it is not necessary that the machine running the browser containing Hyperview in a Browser be able to connect to CIUMon, so long as it can connect to a Hyperview Service.

3.3.7.2 Live loop settings

Unlike in desktop Hyperview, live loop settings cannot be configured on the fly for Hyperview in a Browser or Hyperview Service. In order to get live data, live loop settings must be built into the project file. The Hyperview Service will automatically connect to the CIUMon(s) specified in the project file, and all Hyperviews in web browsers connected to this server will use these connections.

3.3.7.3 Updating live data settings

On the main Menu, click **Live Data Settings**.

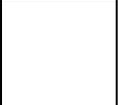
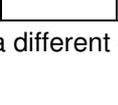
A panel appears that lets you configure your Hyperview in a Browser live data.

For more information on how live data works in Hyperview in a Browser, see [Showing live data](#).



- **Update interval (in seconds):** Specify the interval at which Hyperview Service will request data from the CIU on behalf of Hyperview in a Browser. You can enter in fractions of seconds, setting the interval down to as low as 0.1 seconds.

3.3.7.4 Displaying live data on a document (updated periodically)

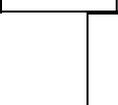
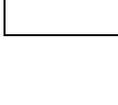
-  Click the  button on the toolbar to start fetching live data.
- Click the  button again to stop.
- Viewing a different document will also stop data collection.

3.3.7.4.1 Behavior

When **Clock Mode** / **Continuous Mode** live data is turned on, displayed live data will be updated at the rate specified in the **Live Data Settings** panel. To conserve bandwidth, only data for items which are actually visible on the display will be fetched. Data for offscreen values will not be fetched. It is usually a better idea to use **Snapshot Mode** live data, which updates less frequently.

Use this feature to retrieve live block values from your ABB INFI 90 process control system. Data is fetched from the Hyperview Service that Hyperview in a Browser is connected to. Hyperview Service fetches the data from a CIUMon that it is connected to. See Showing live data for more information.

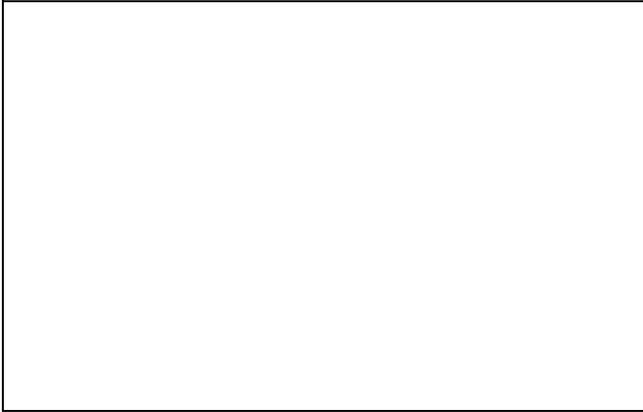
3.3.7.5 Displaying live data on a document (updated when needed)

-  Click the  button on the toolbar to start **Slow Mode** / **Snapshot Mode** live data collection.
- Double click the  button to stop.
- A single click on  will refresh the displayed live data.

3.3.7.5.1 Behavior

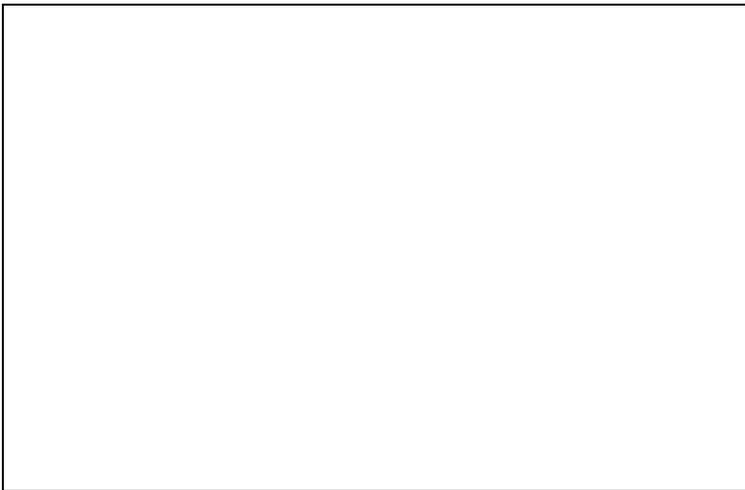
Slow Mode fetches live data for the currently visible hotspots once when you turn it on, and again each time you view a different document, zoom, or scroll. If you do none of these things, **Slow Mode** will fetch fresh data every ten seconds.

Use this feature to retrieve live block values from your ABB INFI 90 process control system. Data is fetched from the Hyperview Service that Hyperview in a Browser is connected to. Hyperview Service fetches the data from a CIUMon that it is connected to. See Showing live data for more information.



When you first view a document that is fully zoomed out, so that the entire document is visible, **Slow Mode** will delay requesting data for five seconds. This is so that if you are immediately zooming in, data for the entire document won't have been fetched unnecessarily. You can cause data to be requested immediately by clicking the **Slow Mode** button in the toolbar.

If more than the default eleven seconds passes before data is updated (the default Hyperview stale data interval), the data will display as "gray" so that you can tell it is old. You can fetch fresh data immediately by clicking the **Slow Mode** button on the toolbar.



3.3.8 Customizing the display

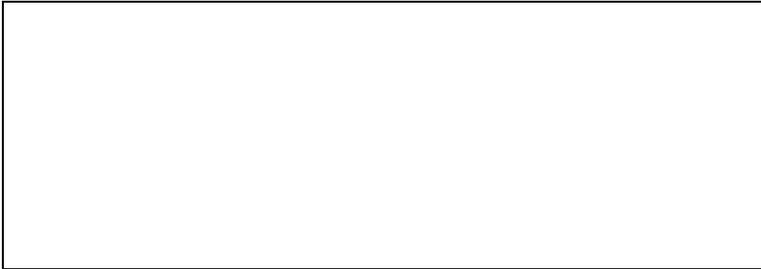
Hyperview in a Browser supports much of the same functionality as **Desktop Hyperview**. The following chapters cover ways to customize the display, such as toggling the display of specs and attributes.

3.3.8.1 Showing or hiding function code specs

On the **Display Settings** Menu, click **Specs**. A panel appears with the options of **Specs Hidden**, **Regular Specs**, **Layered Specs**.

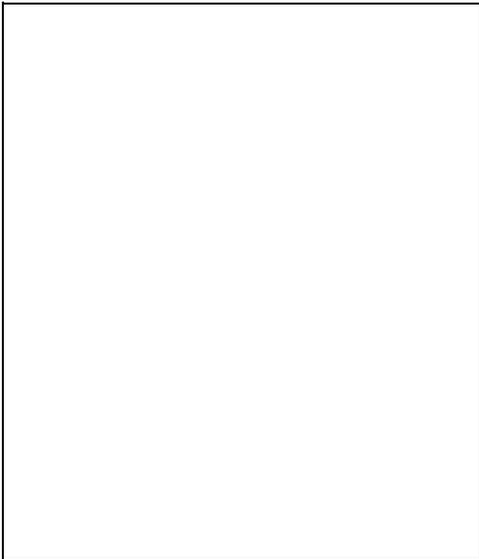
You can also cycle through these options when the menu is not active by repeatedly pressing **S**.

3.3.8.1.1 Spec Display Modes

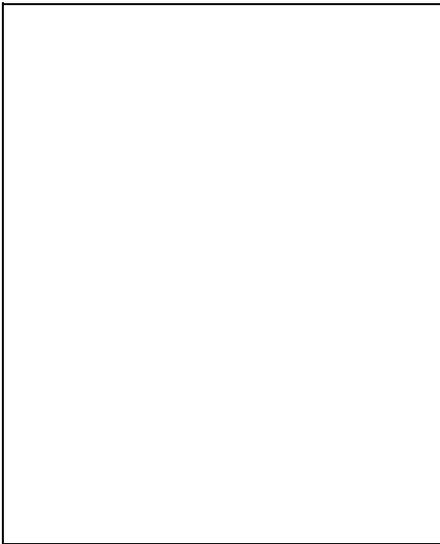


There are three Spec display modes.

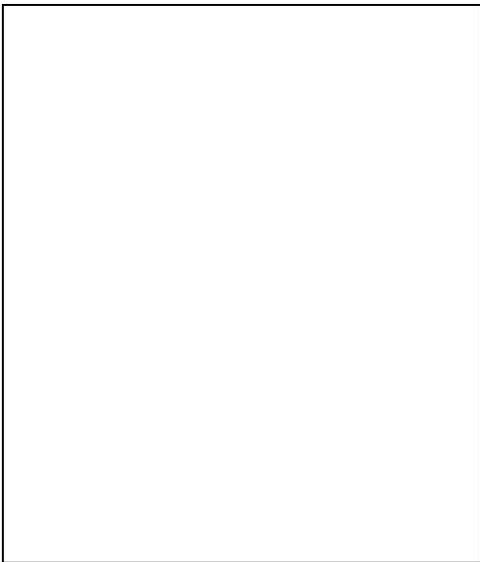
- **Regular Specs:** Displays Specs for a particular block in plain text.



- **Layered Specs:** Displays Specs for a particular block with a gray background. In layered mode you are also able to drag the Specs anywhere on the drawing.



- **Specs Hidden:** Turns the Specs display off.



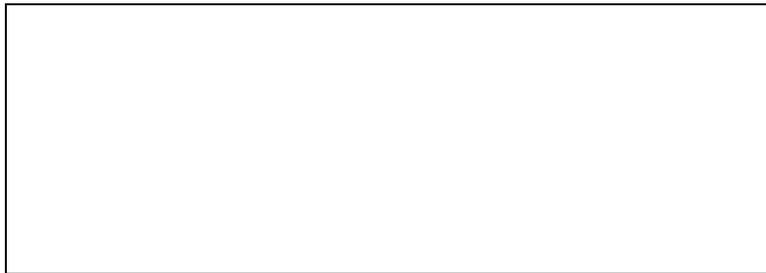
3.3.8.2 Showing or hiding tag names and attributes

On the **Display Settings** Menu, click **Attributes**. A panel appears with the options of **Attributes Hidden**, **Regular Attributes**, **Layered Attributes**.

You can also cycle through these options when the menu is not active by repeatedly pressing **A**.

3.3.8.2.1 Attribute Display Modes

There are three Attribute display modes.



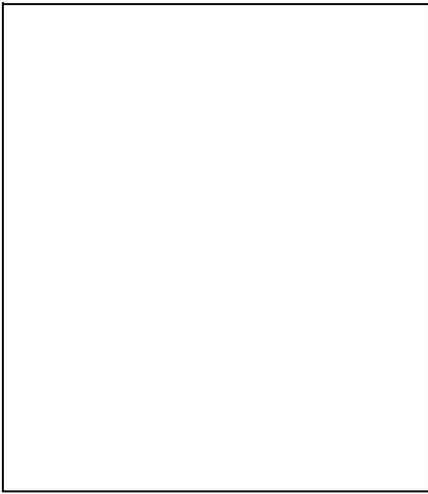
Regular Attributes: Displays Attributes for a particular block in plain text.



Layered Attributes: Displays Attributes for a particular block with a gray background.



Attributes Hidden: Turns the Attributes display off.



3.3.9 Finding version information

Hyperview in a Browser is in communication with a regular Hyperview application which is acting as a web server, and displays a DBDOC file built by Hyperlink. The versions of the server Hyperview and of Hyperlink can be viewed from the online interface.

3.3.9.1 Showing DBDOC file build information

On the **Help** Menu, click **Build Information**.

Displays information about the version and build date of Hyperlink, and the code page that was used to build the currently open project file, as well as the date the project file was built and the computer used to build it. The location of the project file on the server is also shown.

3.3.9.2 Showing Hyperview Server information

On the **Help** Menu, click **About Hyperview**

Displays the current version of the Hyperview being run as a web server, as well as the command line that was used to run it.

You can also see the version of the Hyperview in the status bar at the bottom of the browser. The current date and time is at the far left, with the current Hyperview version just to the right.

3.3.10 User accounts

Hyperview in a Browser can be configured to require user accounts for access, or for these to be optional. For more information, see the **No Login Required** option for Adding or modifying a Hyperview Service.

By default, Hyperview in a Browser services allow access without user accounts, i.e. in **No Login Required** mode.

However, if the **No Login Required** option is not chosen when a Hyperview Service is configured, users are required to have accounts in order to access Hyperview in a web browser, and logging in is required.

Even when logging in is not required, it is still possible to log in, if user accounts exist. Since user data like bookmarks and home pages are associated with user accounts, logging in makes these available from browsers on different computers. When Hyperview in a Browser is used without the user being logged in, these items can still be created, but they are associated with the IP address of the moment, rather than a user, and thus will not be available from other computers.

Users may update their own passwords. Only the Administrator user may create, modify and delete user accounts.

3.3.10.1 Logging in to your Hyperview in a Browser account

When you navigate to the Hyperview in a Browser website, you will see a login page, assuming the Hyperview Service has been configured to require logins.

If logins are required, in order to connect to a Hyperview Service and use Hyperview in a Browser, you must have a user account. The Administrator user can create an account and set an initial password, which you will be able to update as needed.

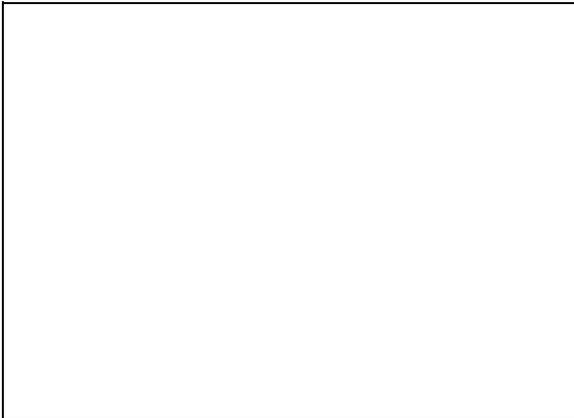
It is possible to configure Hyperview in a Browser to not require logins and user accounts -- see Adding or modifying a Hyperview Service for information on the **No Login Required** option. If Hyperview in a Browser is running in **No Login Required** mode, it is still possible to log in via the **Sign In** option on the menu. The Administrator can log in there, and manage user accounts in the usual manner, and individual users can log in if they have accounts.

User data such as bookmarks and Home pages are generally associated with specific user accounts. However, if Hyperview in a Browser is being used in **No Login Required** mode without a user logged in, this persistent user data will be associated instead with the IP address that the browser is running on, and will not be accessible if accessed from a web browser on a different machine.

3.3.10.2 Changing your password

On the main menu, click **Change My Password**.

A panel appears that lets you modify your user password. You must know your current password. If you don't know your password, the Administrator user can update your password.



3.3.10.3 Signing out of your Hyperview in a Browser account

On the main menu, click **Sign Out**.

You will be signed out of your Hyperview in a Browser account.

3.4 Managing user accounts

3.4.1 User accounts

Hyperview in a Browser comes with a default Administrator account with username **admin**. The "admin user" can add new accounts. Regular users can update their passwords, but cannot add new accounts.

To obtain the default password to the Administrator account, please contact GMCL.

Note that it is possible to run Hyperview Services without requiring logins or accounts. See the **No Login Required** option for Adding or modifying a Hyperview Service

For more information see:

3.4.1.1 User actions

- Logging in to your Hyperview in a Browser account
- Changing your password

3.4.1.2 Administrator actions

- Creating, modifying, and deleting user accounts
- Resetting the Administrator password

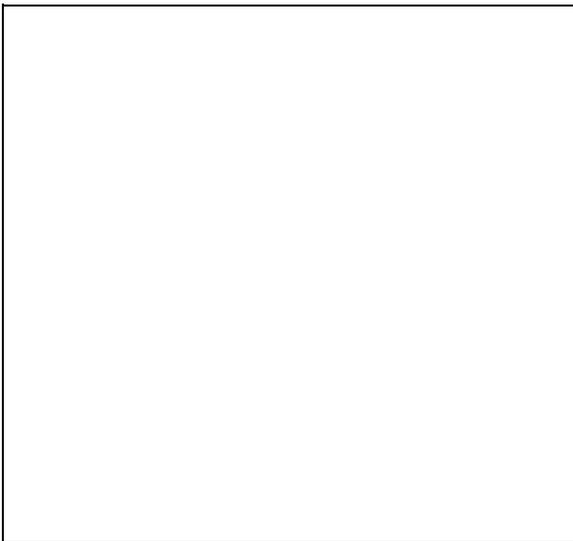
The user accounts apply to all Hyperview Services running on a machine. If there are multiple Hyperview Services serving different DBDOC files as different "websites", for example, the same set of users will be able to log into all of them.

3.4.2 Creating, modifying, and deleting user accounts

On the **Admin** menu of Hyperview in a Browser, click **Manage Users**.

Note: Only the Administrator user will see the **Admin** menu.

A panel appears listing all the users who have Hyperview in a Browser accounts (the account file is stored with other Hyperview Service user data files).



- **Edit:** Opens an interface for modifying or deleting the user account. The user password can be changed this way.

- **Add User Account:** Opens an interface for adding a new user account.

3.4.3 Resetting the Administrator password

Only the Administrator user can create new user accounts and change user passwords. To do so, log into Hyperview in a Browser with username **admin** and the Administrator password. If the Administrator password has been lost, it can be reset via the **Configure Hyperview Services** dialog of the Hyperview Service Controller.

4 Overview of BuildPlus

BuildPlus is used to build your system's configuration, graphics and databases into a Snapshot file that can be viewed using the Hyperview browser, providing a read-only snapshot of your process control system. Once you have configured the first build for your system, you can rebuild very easily, and even run builds automatically from the command line.

4.1 Getting Started with BuildPlus

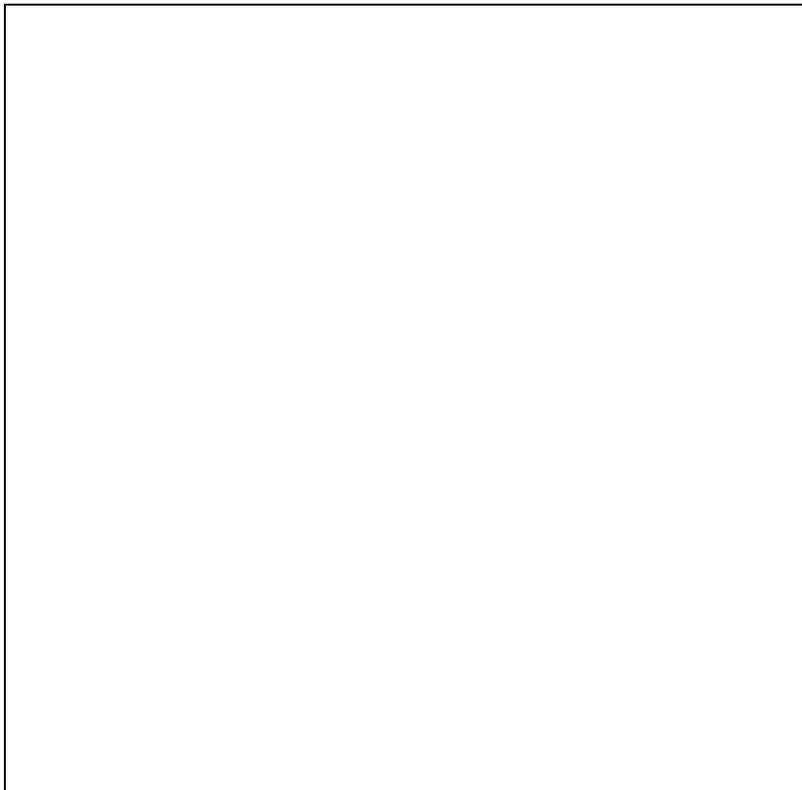
For information about installing DBDOC (including BuildPlus), see the **DBDOC Installation and Setup** section of this manual.

4.1.1 DBDOC dongle

A USB dongle (security key) is required to run a build to completion with BuildPlus. A warning will appear if the dongle is not present when you start BuildPlus, and also if you get to the Hyperlink compile stage. The build will not continue and run Hyperlink until the dongle is present.

Note:

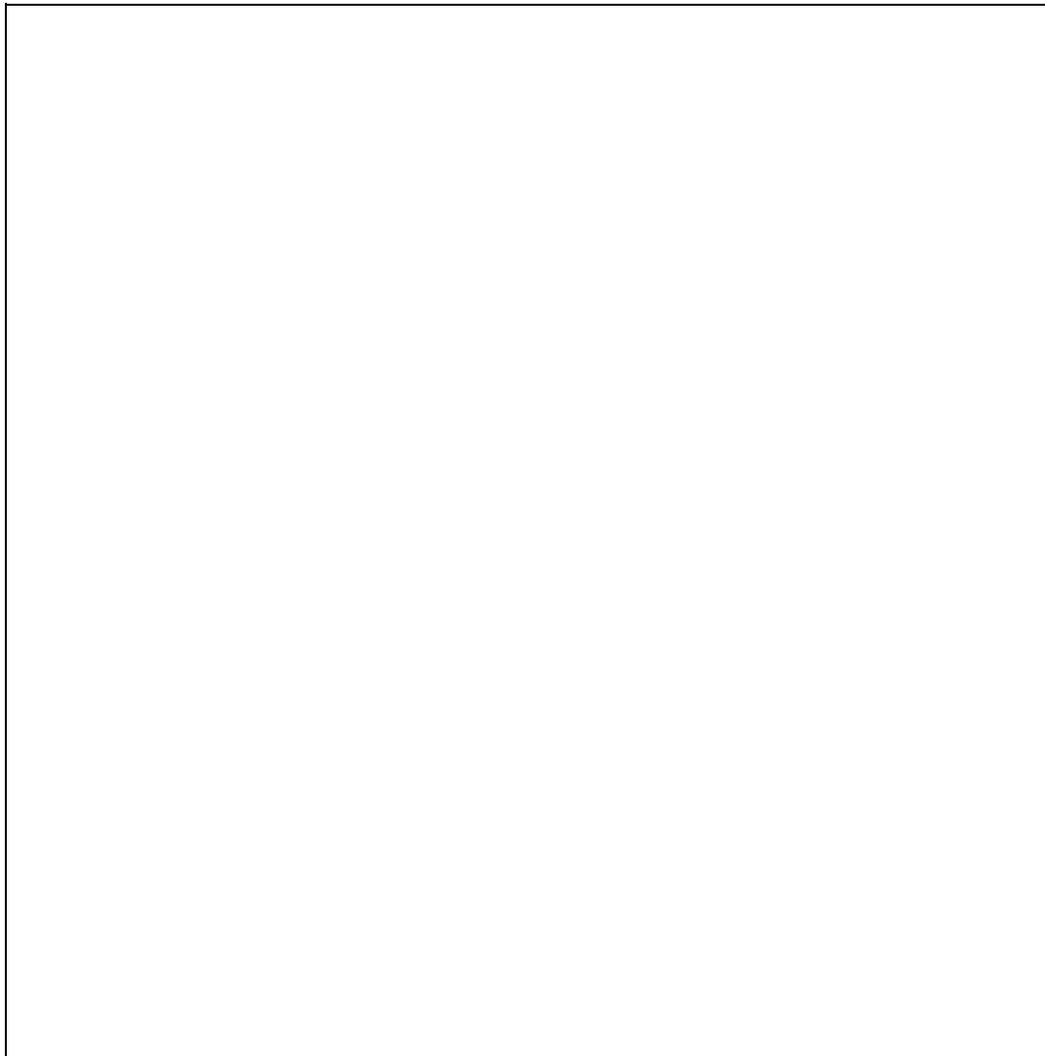
As of DBDOC 11.0, new *Sentinel* dongles are required. The old *MicroDog* dongles used with earlier versions of DBDOC will not work and BuildPlus will just give the *License Issue* warning shown here.



Please contact GMCL for assistance with dongle-related issues.

4.1.2 About your DBDOC license and computer key

On the **Help** menu, click **DBDOC License** or on the **Tools** menu, click **DBDOC License....**

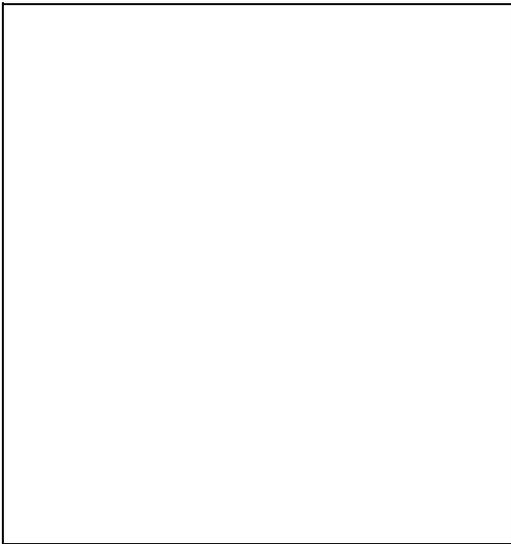


4.2 Adding or changing license information

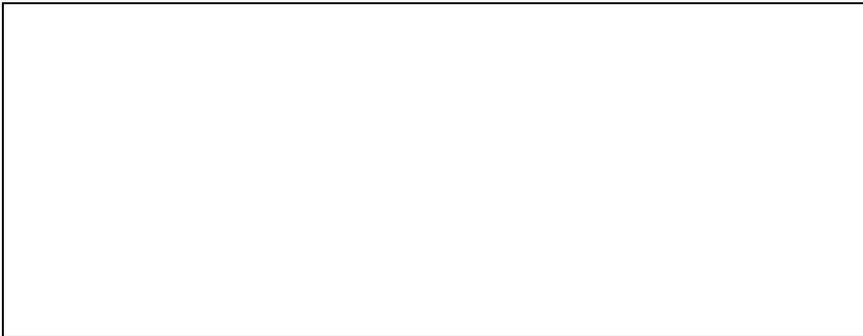
See Licensing DBDOC.

4.2.1 Opening BuildPlus

- BuildPlus can be opened using the **Start** menu - (in some versions of Windows it may be under **Programs**) look for a **GMCL DBDOC** folder and click **BuildPlus**.



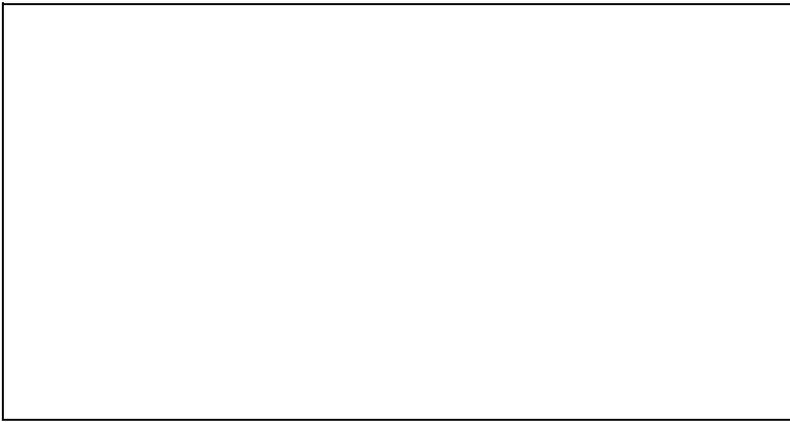
- BuildPlus can be opened from DBDOC program file, located by default at C:\Program Files (x86)\GMCL\DBDOC\Programs, with the file BuildPlus.exe.
- If the project is already open in another copy of BuildPlus, the following dialog will appear. Once you click **OK**, BuildPlus will start and allow you to open a different project.



4.2.2 About BuildRoots

Creating a BuildRoot actually creates a new working directory, where projects will be stored.

- The first time BuildPlus is run, it will automatically ask you to create a BuildRoot. BuildPlus creates several fairly large working files, so put your BuildRoot on a disk with plenty of space.
- On running BuildPlus for the first time, BuildPlus will suggest a BuildRoot for you – X:\DBDOC_BUILDS, where X: is the local disk with the most free space. You may also browse to choose a different directory, or type in the directory of your choice.

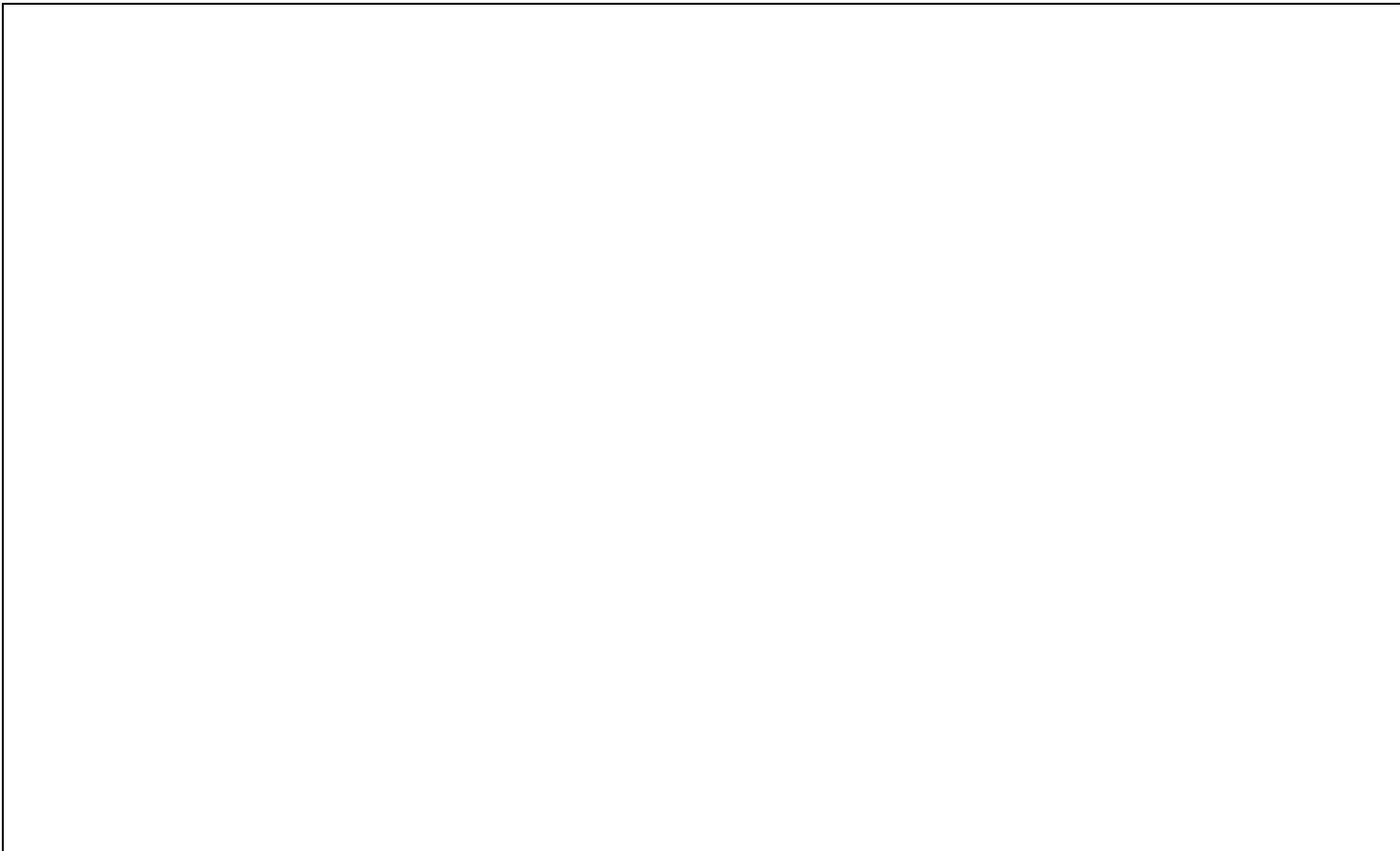


- The code page for the BuildRoot is the same encoding that is used for your INFI 90 files. The default already selected is the code page your system uses, and is usually correct.
- Clicking on the **Options** button allows you to change BuildRoot Options.

Changing BuildRoot options

BuildRoot Options apply to all projects in the BuildRoot. (Global options apply to all projects on the build computer, and Project Options apply only to the open project.)

Other than the **Create BuildRoot** dialog above, to access BuildRoot options, on the **Tools** menu click on **BuildRoot Options**.



Usually the default selections should be used. For information about each option, look in BuildPlus help documentation under BuildRoot, changing options, and select the link for that option.

If a project has not been created yet, BuildPlus will launch the **System File Specifications** dialog. See **Doing Your First Build** to continue.

4.2.3 Getting Help with BuildPlus

There are many ways to access DBDOC help from BuildPlus.

Contacting GMCL

E-mail	support@gmcl.com
Phone	866 438-2101 Toll Free in North America
Phone	(+1) 780 438-2101
Fax	(+1) 780 437-3367
Web	http://gmcl.com

We provide virtually 24x7 support coverage.

Choosing Help from the BuildPlus main menu

- On the **Help** menu, click **Contents** to go to the contents (or first) page of the help file.
- On the **Help** menu, click **Index** to go to the index page of the help file.
- On the **Help** menu, click **Search for Help on** to go to the search page of the help file.
- On the **Help** menu, click **GMCL Help** to see options to either launch your default browser and go to the Help page on the GMCL website, or send an email to GMCL support.
- On the **Help** menu, click **Display BuildPlus .nod file location...** to display the location of crucial files pertaining to projects and BuildRoots. This is also the location of DBDOC_Components.txt, when the Tool **Enumerate DBDOC Components** is applied.

Using ? button on dialog boxes

At any time it is available, contextual help can be obtained by clicking on the ? box in the top right corner of the dialog box, and then clicking on the subject you are wondering about.

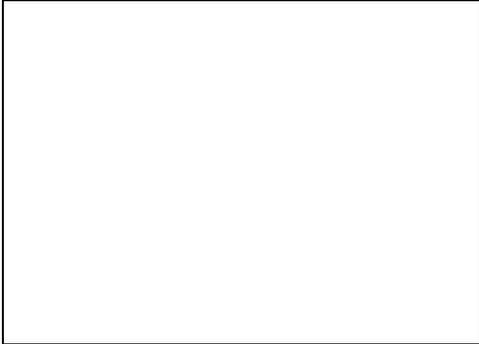
Using "What's This" from the toolbar

On the BuildPlus toolbar, click on , then click on the item you need help with.

4.2.4 About your DBDOC version

When contacting GMCL for assistance or updates, you may be asked for the versions of your DBDOC components.

- To find the BuildPlus version, on the **Help** menu, click **About BuildPlus**. This displays the current version of BuildPlus as well as the version of the currently installed kit.



- To find other DBDOC component versions in the project directory, use the **Tools** menu to **Launch Windows Explorer** to the project directory and open the file named **DBDOC_Components.txt**
- To create a file listing all DBDOC components in the Application Data folder, on the **Tools** menu, click **Enumerate DBDOC Components**. This creates text file DBDOC_Components.txt in one of the following locations:
 - ◆ C:\Documents And Settings\All Users\Application Data\GMCL\Projects, for users who have write privileges to this location, or
 - ◆ C:\Documents And Settings\\Application Data\GMCL\Projects.

or equivalent directories depending on the operating system. The file lists the versions of all DBDOC programs and any mismatches relative to the kit version of DBDOC installed.

4.3 Doing Your First Build

To configure your project in preparation for a build, the following steps will be completed:

1. Naming Your Project and Choosing Database Files to Build
2. Choosing Configuration, Graphics and Other File Locations with the Configuration Wizard
3. Changing Project Options, if desired
4. Resolving any other questions asked by BuildPlus

4.3.1 Naming Your Project and Choosing Database and 800xA Files to Build In

The **System File Specifications** dialog is launched automatically for the first build.

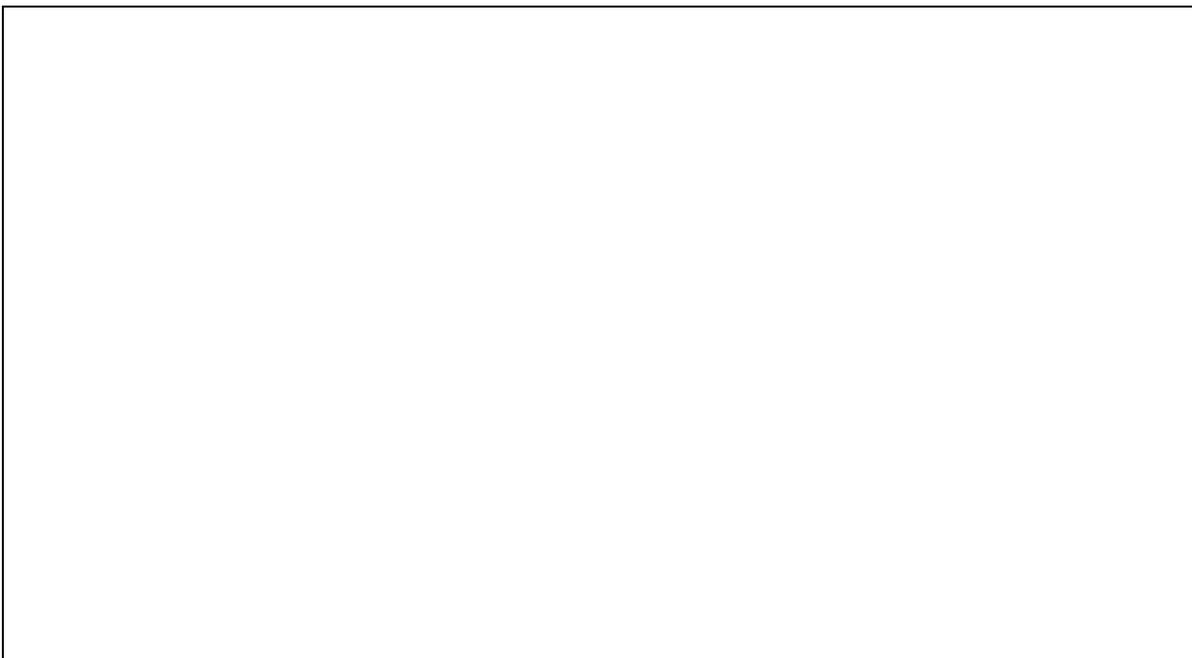
At other times, it can be retrieved from the **Build** menu, by selecting **System File Specifications**, or by

clicking the  button on the BuildPlus toolbar, to change the databases and/or 800xA graphics included in the build.

At any time, contextual help can be obtained by clicking on the ? box in the top right corner of the dialog box, and then clicking on the subject you are wondering about.

Naming your project

In the **System File Specifications** dialog, enter a unique and meaningful name for your project (e.g. a few words that identify the function, location, plant or company).



Choosing database (and 800xA graphics) types

Check the boxes that apply to database (and 800xA graphics) types in your project:

- Composer systems – **.EBP** databases
- Access **.MDB** databases – Process Portal B / Operate IT / Conductor NT systems / 800xA. (See also instructions to extract an Operate IT Database for 800xA and for PPB)
- Process Portal **.XML** Databases – for some Process Portal B systems
- WinCAD/WinTools systems – **.PTT** Relational databases
- Generic tag and trend databases: **.DBF**, **.TTG**, **.TTR** and **.UCF**, also **OPC-90 .CSV**, **Citect Console Databases**, plus any other database types that you can export into **.DBF** format.
- S+, PI, generic - **.XLS** and **.XLSX** Excel Workbooks - This reads in data stored in spreadsheets, including OSISoft PI configuration documents and Symphony Plus graphics databases
- 800xA **.AFW** Backup Folders (non-ASSYST **.AFW** dir) - **.AFW** directories Contact GMCL for information about ASSYST extraction of your graphics. (See also instructions to extract 800xA PG2 graphics.)
- 800xA BBA/ASSYST Files Folders (PNG dir) - (directory that contains **.png** graphics files in folder BBASnapshot or AssystSnapshot) (See also Extracting 800xA graphics using the Assyst tools.)
- **ABB Function Code PDF Files** (any form) can be built in checking this box.

Specifying where to search for your database, 800xA graphics and Function Code files

(Note that the databases, graphics and function code files do not have to be located on the same computer, or on the build computer, and DBDOC access is read-only. But if you don't have access through a network to the computer that contains the source files, you may have to copy them to your local computer. In this case, leave the directory structure the same. Use UNC naming convention when possible.)

Either: Enter the folder name in which the file(s) reside, and BuildPlus will find the files, or you may include the name of the file.

Or: Click on **Browse** to select a directory or file. (This choice will be appended to the end of any search paths already entered.) You can **Browse** for the directory in which the database file(s) reside and BuildPlus will find the files, or you may include the name of the database file.

- Once a path has been entered, the drop-down arrow can be used to select the same location for other database types, or you can copy and paste from one box to another.
- More than one path will be searched if multiple paths are listed for a database type, separated with a "|" character

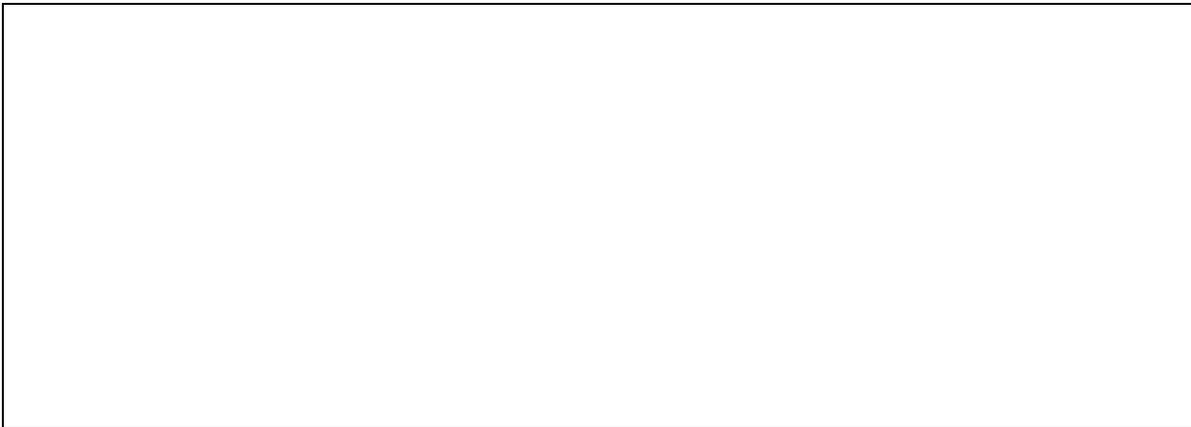
After specifying the locations of your files, click **Next** so BuildPlus can find your files.

Selecting specific databases and 800xA graphics locations

- The left panel displays "databases to select from"; all databases of the specified type found by BuildPlus in the specified search paths. If you wish to change the search path to find additional

databases, use the **Back** button.

- The right panel displays "databases to be included in the build", including any database names entered previously (provided they were located in the search). If only one database was found during the search, it will be automatically moved into the final list.
- Move a database from one list to another by selecting it from the initial list and clicking the enabled **Add** or **Remove** button as required. Using the **CTRL** or **SHIFT** key in combination with the mouse can make multiple database selections.
- If a database has "Unimplemented Type" listed for the file size, BuildPlus has identified that it cannot process the file, and that file cannot be selected.
- The **Next/Finish** button will be enabled when there is at least one database in the final list
- To proceed with the build without selecting any databases of a given type, use the checkbox **Check here to continue without selecting any databases**. You may also use the **Back** button to omit that type of database from the build.



- Composer EBP databases are each shown with a checkbox in the final selection list. Unchecking the box means that databases extracted from EBP file will not be used or included in your final Snapshot. Generally, you would only uncheck the EBP database if it is unused, obsolete or in some other way invalid, and a better database can be built in.
- Locations of Function Code .pdf files are listed for information, and BuildPlus will select the appropriate files. To change which locations of Function Code .pdf files are included, use the **Back** button to the **System File Specifications** box, find your preferred file location and check the **Redo** box, then click **Next** to proceed.

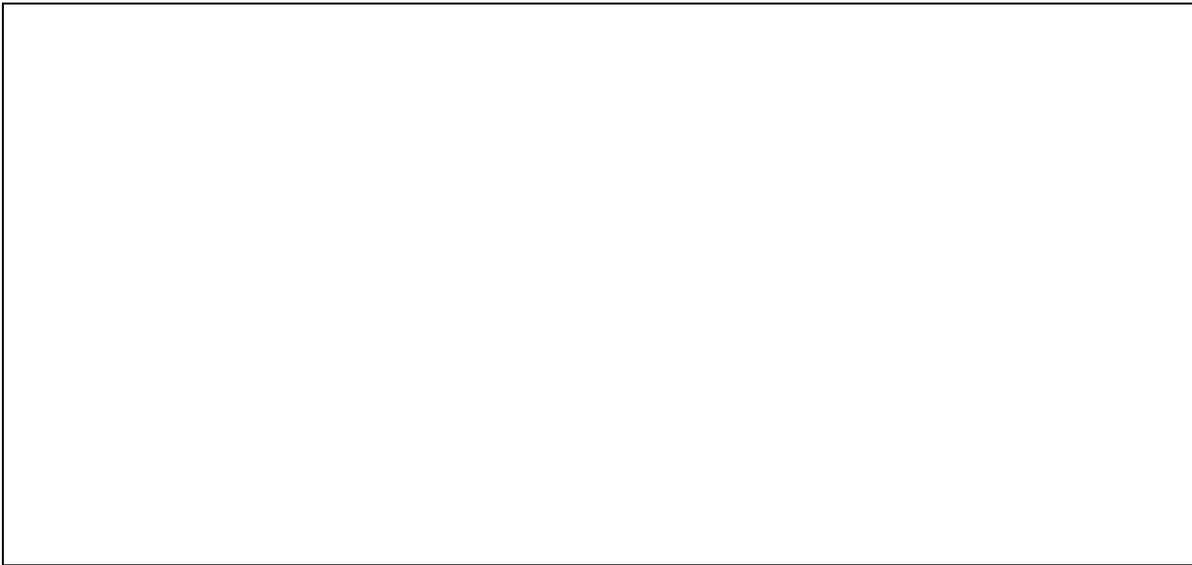
System File(s) Chosen lists all your selections so far. If corrections need to be made, press the **Back** button to review the system file selection panels. Otherwise, press the **Finish** button to continue. After some database processing, the Configuration Wizard dialog will appear.

4.3.2 Specifying Configuration, Graphics and Other Files with Configuration Wizard

- For a new project, the Wizard is only accessible after the project name and databases have been selected in the System File Specifications dialog.
- Some Wizard selections may be made automatically based on Database types already chosen.
- The Wizard only needs to be run once unless file selections need to be changed. To access Wizard

later to make changes, on the **Build** menu of BuildPlus, click **Project Wizard**, or click the button on the toolbar.

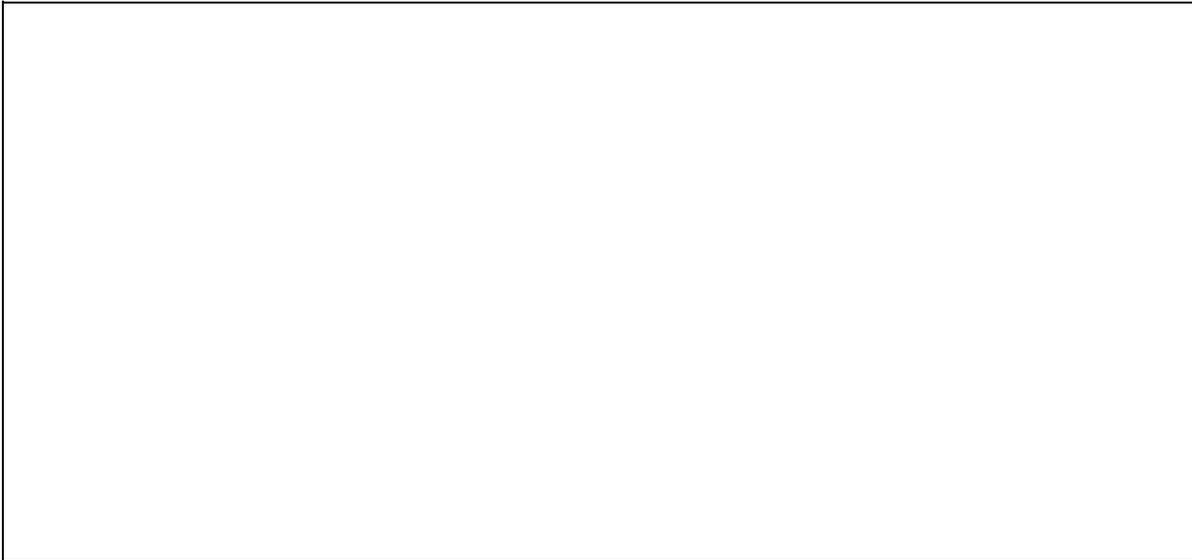
- At any time, contextual help can be obtained by clicking on the ? box in the top right corner of the dialog box, and then clicking on the subject you are wondering about.



Within the Wizard,

- Choose one configuration type with a round checkbox, or choose Other/None and select other types of configuration documents: Batch (.b90), Ladder (.lad) or .AB and .RTU files.
- Choose one type of SODG graphics with a round checkbox, or choose Other/No SODG Graphics and specify S+/PGP, Process Portal B and/or Conductor NT graphics as appropriate. (The 800xA checkbox will be checked based on if 800xA PG2 files were included in **System File Specifications**. (You may also choose a type of SODG graphics plus 800xA, Conductor NT and/or Process Portal B graphics in the same build.)
- As of DBDOC 10.7, you may specify a location for .jpg, .jpeg, .png, or .gif images to be included in S+ or PGP graphics. If your project already included S+/PGP graphics, a **S+/PGP Graphics Detected** dialog may appear, allowing you to specify the location of these images. These selections can also be modified when rebuilding from Wizard.
- Specify other file types to be included in the build by checking appropriate optional square checkboxes.
- As of DBDOC 10.7, it is possible to have pdf files built-in for display within the Snapshot file. If your project previously included externally-linked pdf files, a **PDF Documents Detected** dialog may appear, allowing you to build-in any or all of your folders of pdf files. These selections can also be modified when rebuilding from Wizard.

- If you have AutoCAD **.PRO** format files, check the box for **AutoCAD (.DWG)**. The text box to the right of this will open, allowing you to enter **.PRO**.
- ORCAD files saved with extension **.dxf** can be built in by selecting Exported CAD (**.DXF** only)
- C-programs with a **.txt** extension can be included by selecting Text
- Click **Next** to continue once all selections have been made.



"Please specify the directory where most of your project source files are located, then press the Apply button"

- When most directory paths are empty, specifying one directory where most files are located, then pressing **Apply** fills in all directory search paths with that one choice as a starting point
- Otherwise, the **Disable** box should be checked.

Entering paths to files

- Enter the folder name in which the file(s) reside, and the wizard will find the files
- Once a path has been entered, the drop-down arrow can be used to select the same location for other files **OR** you can copy and paste from one box to another.

Browsing to choose paths

- Clicking on Browse will bring up the Windows "Browse for Folder" dialog, where a directory can be selected, or a folder name can be copied in. This choice will be appended to the end of any search paths already entered.

Adding more than one path

- More than one path will be searched if multiple paths are listed, separated with a "|" character

- Multiple paths can be quickly added by clicking the **Browse** button and selecting additional paths.

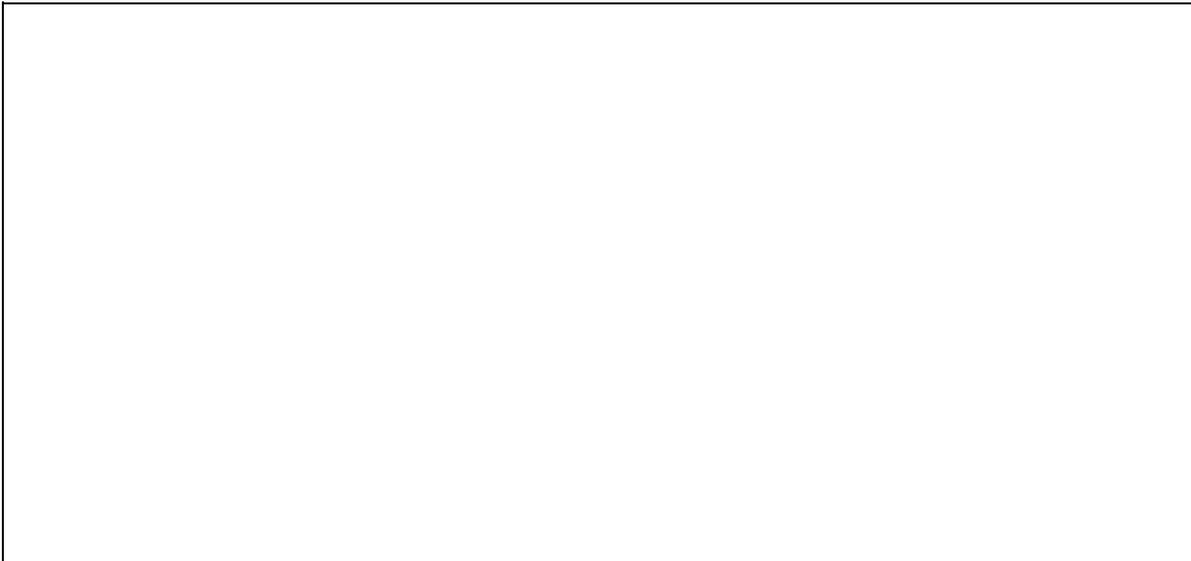
Red arrow alert

When you see a red arrow there may be invalid path information, which must be corrected to proceed.

(If files are located on a remote machine you can't access, it may be necessary to copy them to your local computer. If you do, don't change the directory structure as it helps BuildPlus make some automatic associations.)

- When DECOMPOSER folders within the DBDOC_BUILDS directory are listed, they contain files created by DBDOC and should usually be left as the directory for Wizard to search.
- The CAD user libraries are usually omitted. For Composer projects, BuildPlus extracts a MACROS.LBR file. To specify a different user library, uncheck this Omit checkbox. This allows you to build with what you consider to be the master copy. (WinCAD systems often have multiple copies that must eventually be consolidated and united.)

After specifying the locations of files on each page, click **Next** to have Wizard find your files.



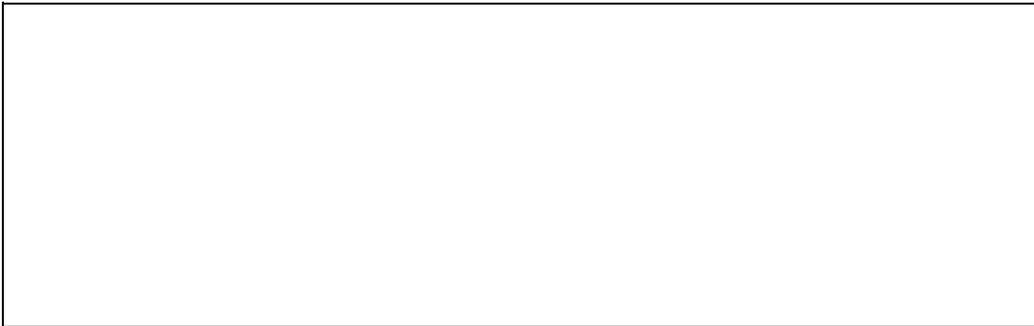
Confirming File Selections

- The left panel lists folders identified within the specified directory which contain each file type, with the currently selected folders highlighted. The right panel lists files contained in the last selected folder.
- To assist with choosing files, the lists can be sorted alphabetically or numerically by directory name, size or date modified by clicking on each heading.
- Using the **CTRL** or **SHIFT** key in combination with the mouse can make multiple file selections.
- The **Explore** button opens the specified directory to view the folders and files within.

- If the expected files aren't listed in the left panel, use **Back** to change where the Wizard should look for files, and ensure the **Redo** box is checked for a new Wizard search.
- Checkbox **Non-Optional: You should select an item. Otherwise, check box** can be checked to skip certain files or groups. This is not a recommended practice since your build results may not be valid.
- **Single MHD: Only one MHD has been selected, check to continue** should be checked if you wish to proceed with only one module in the build.
- To see which .ebp database file was used for extraction into each **DECOMPOSER** folder, move the mouse over the name of the folder and a datatype will be displayed.
- For unit console graphics, to see which console is involved with each **EBYxxx.TMP** folder, move the mouse over the name of the folder and a datatype will be displayed.

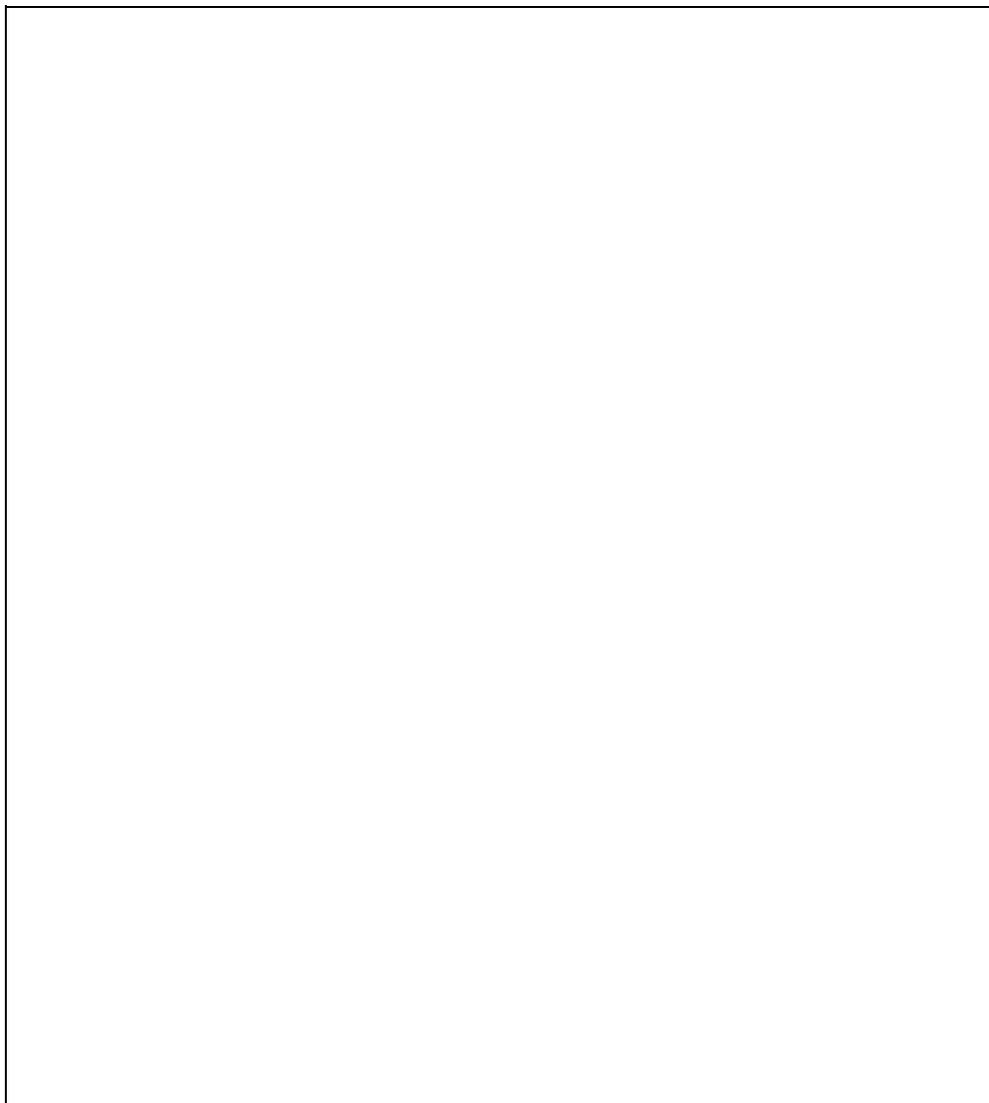
After selections have been made, click **Finish** to proceed with the build.

- The **Run Project Options** dialog may appear next, giving you an opportunity to change project options. Click **Yes, edit settings now** if you wish to build error files into your Snapshot and/or build in settings for live data. (See below.) For information about other project options, see BuildPlus Help:project options, changing. Click **No** to continue with the build using default project options.

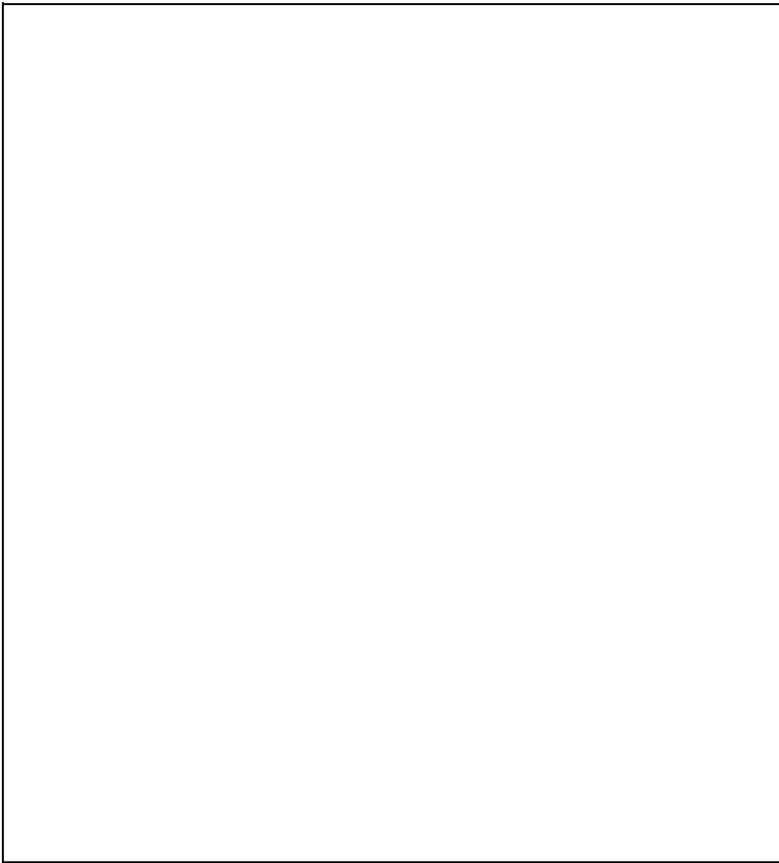


4.3.3 Building in Settings for Live Data

You may choose to **Run Project Options** after Wizard, or on the BuildPlus **Tools** menu, click **Project Options** and navigate to the **Live Loop/CIUMON Settings tab**.



1. Choose to either **Show real data from CIUMon** (default) or **Show demo (fake) data**
2. To add CIUMon settings, click on **Create list**



1. Change the **Name** for the list if you wish (this is helpful when multiple lists may be created)
2. To enter CIUMon information, click on **Add** under the CIUMon box. Enter the **Hostname or IP** address of the CIUMon, and the **Port**, if not 5099 (default). Click **OK** to continue.



1. To edit or delete a CIUMon from the list, first select the location to be edited or deleted, then click **Edit** or **Delete**
2. It is occasionally necessary to enter **Routing preferences**, for example to identify a specific CIU for an individual's testing. To enter routing preferences, click on **Add** under the Routing preferences box and select the Loop for which the preferred server will be requesting data, or uncheck the **All** button and enter the specific PCU. Click **OK** to continue.
3. To change the Hyperview local port autobind range from the default (5100 to 5200), choose **Use custom port range** and enter your preferred settings.
4. When finished adding and editing CIUMon lists, click **OK** to see your selected CIUMon settings lists



1. To create another list, click on **Create list**. To edit or delete a list, select the list, then **Edit list** or **Delete list**.
2. Click on **Apply** if you wish to save your lists and change other project options. Click on **OK** to continue.
3. If you are adding CIUMon settings after a build is complete, choose **OK** in the CIUMon settings changed dialog.

4.3.4 Resolving Other BuildPlus Dialogs

4.3.4.1 Duplicate File Choices

BuildPlus reports any duplicate module headers for CAD or CLD files so the correct entries may be selected for the build:



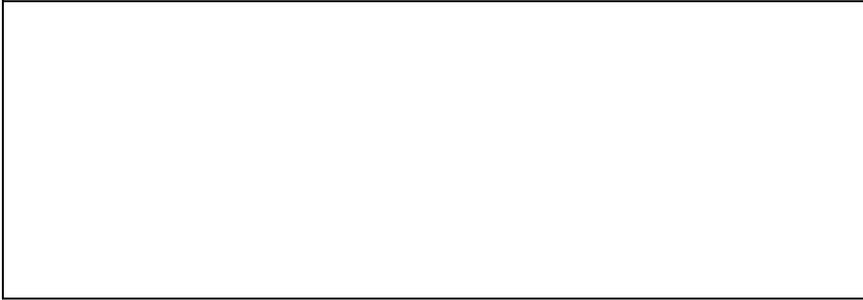
Modules that are unique have been placed in the final selection list on the right-hand pane. The duplicate modules that need to be resolved are listed at the bottom of the dialog box, with the choices listed in the left-hand pane. (Files are grouped together based on a common root directory and are displayed in a treeview.)

- Clicking the **Check Source** button lists which Composer database (.ebp) file was used to create each Decomposer directory
- Select individual modules or the full root path to move into the final list; check the box beside the appropriate files and/or roots and click the **Add** button (or use **ALT+A**).
- To remove from the final list, select the appropriate files and/or roots and click the **Remove** button (or use **ALT+R**).

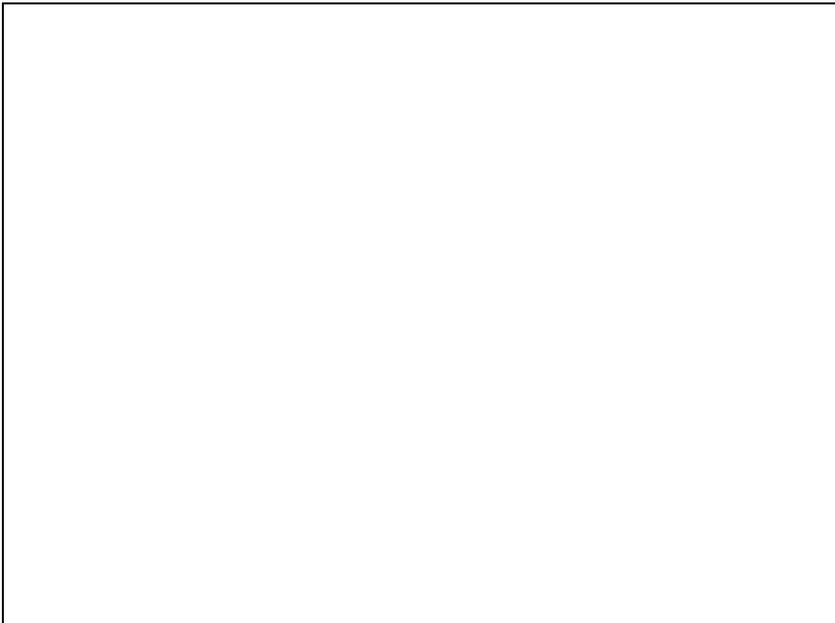
Once the correct selections are made, this dialog will no longer appear, and the final selections can be viewed within the Wizard.

4.3.4.2 Conductor NT or PPB Console associations

When multiple consoles are present, you may see:



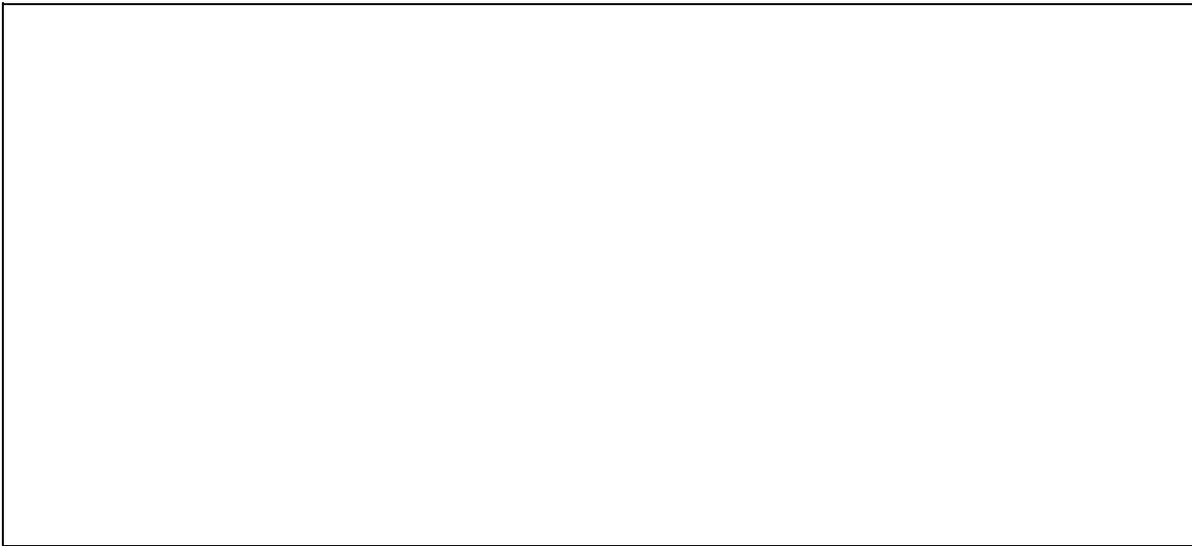
When you click OK you will have the chance to review and change associations made by BuildPlus. To check associations at other times, on the **Tools** menu, click **Conductor NT/PPB Console Associations**.



In this dialog, BuildPlus presents a list of the consoles that were created for the project.

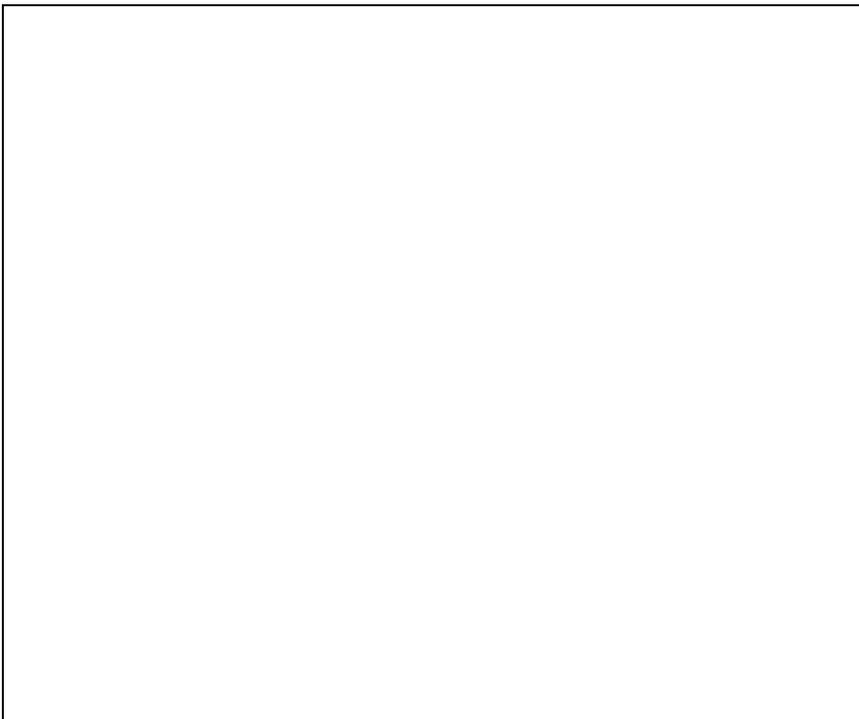
- Check the boxes to choose which consoles to include in the build
- Check the box to define an additional console if required.
- If you click **Skip and Continue**, the build will continue without saving any of the changes so far.
- Pressing the **Cancel** button will cause BuildPlus to exit out of the associations and stop the build.
- Click **Next** to continue, to change a console name or change associations for a console.

The Conductor NT/PPB Console Associations window will appear for the first console:



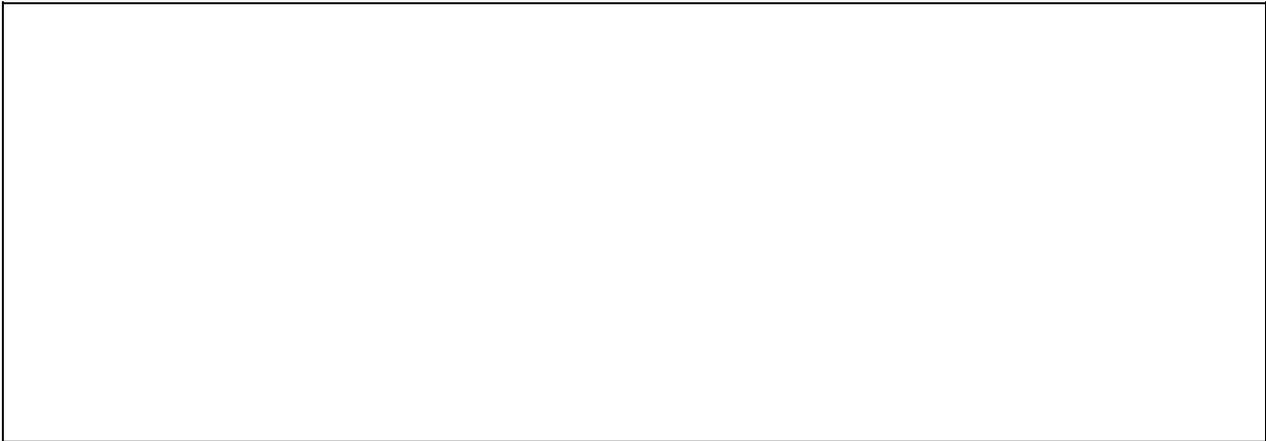
To change the console name from the default Console1, select the console name and enter a more meaningful one. (Each console must have a unique name.)

To change Databases associated with this console, click **Change Database Selections**, and the database selection window will appear:



In this window, BuildPlus presents all of the databases that are included in the project, along with databases that may have been extracted from the original choices (e.g. Relational .PTT, Composer .EBP). BuildPlus will also show the number of records contained within each database. Select all databases required for the console; you may also select no databases if that is applicable for your case. Press **OK** when you are satisfied with the selections.

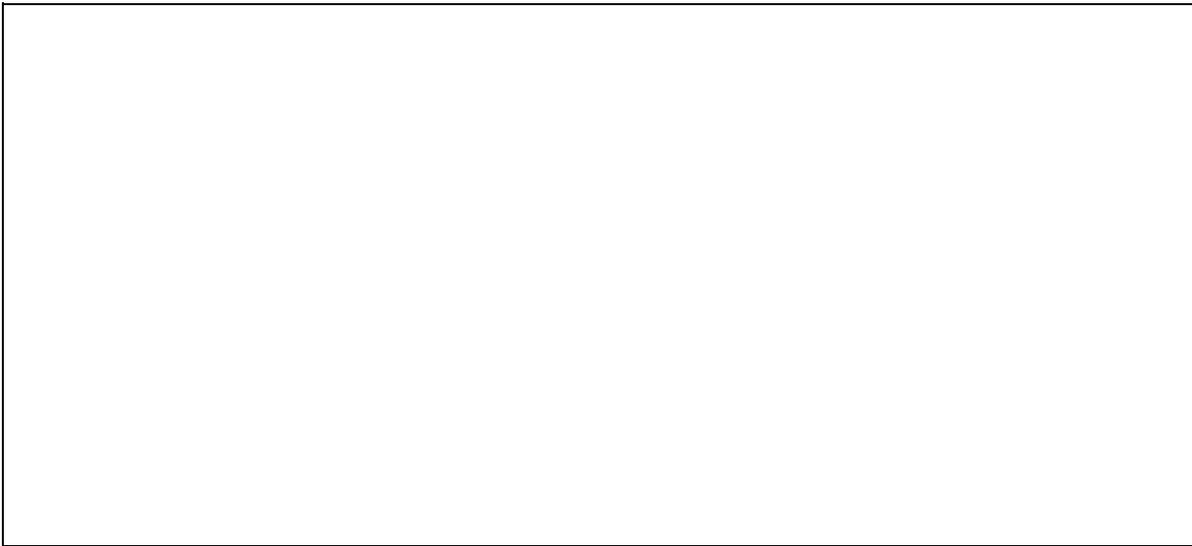
To change **XDC, colordef.dat or fontdef.dat selections**, in the Conductor NT/PPB Console Associations dialog, click **Change XDC/DAT File Selections**, and the following dialog will appear:



Shown in each list are the files selected through the project wizard. Select the correct entry for the current console in each list. (Note that .xdc files are only required for Conductor NT graphics. See also About .XDC files).

To change **graphics selections**, look at the bottom of the Conductor NT/PPB Console Associations dialog.

- **Global Graphics** lists all graphics paths not currently associated with a console. Any graphics left in this list after all console associations are completed will be available to all consoles. Each path is described as a model or submodel path, and only M1 paths are listed (not M2).
- **Graphics to be built with this console:** lists graphics associated with the currently displayed console.
- to remove graphics or submodels selected for this console, select any graphics listed on the right side under **Graphics to be built with this console:**, and click on **Remove**
- to add graphics or submodels for this console, select them from the list on the left **Global Graphics**, and click on **Add**



Once the correct database, graphics, and other files are selected for the current console, click **Next** to move onto the next console. Clicking on **Back** at any time allows you to check selections for a previous console.

On the last console to be defined, the button **Define another console** at the bottom of the association panel will become available. Click this to create another console.

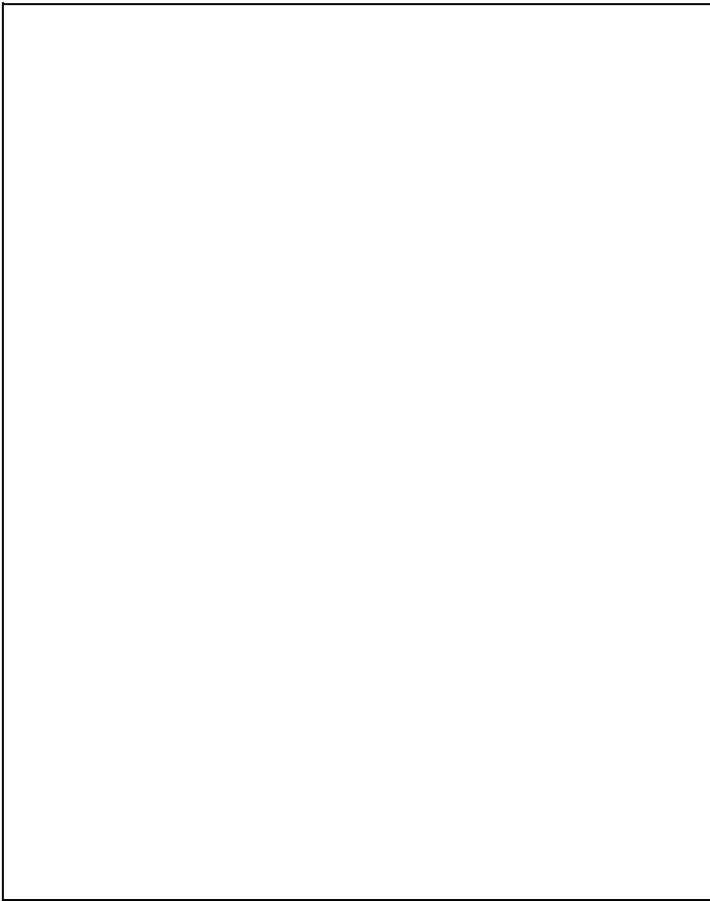
When these associations are done in-line with a build, the build will automatically continue on until completion. If the console associations were triggered from the **Tools** menu option then on the last panel, the **Rebuild project immediately upon finishing** checkbox will become enabled. If this is checked when you click **Finish**, BuildPlus will immediately start the build of the project again to incorporate any changes that may have been made in the associations. If this option is left unchecked, any changes made will not be seen in the Snapshot until the next time the you build the project.

4.3.4.3 Symbols-Graphics Association

This dialog may appear during the build if automatic associations cannot be made, or

On the **Tools** menu, click **Project Options** and navigate to the **Symbols-Graphics Association** tab.

- When a build contains multiple SODG graphics and symbols directories, it may be necessary to specify which set(s) of symbols to use for each graphics directory. This tool allows you to check and change any automatic associations that BuildPlus has made or to make manual associations.



Launch Windows Explorer button: If you want to check out what graphics are contained in a particular folder, select the path from the list and click the button. A Windows Explorer window will be launched, automatically opened to the folder in question.

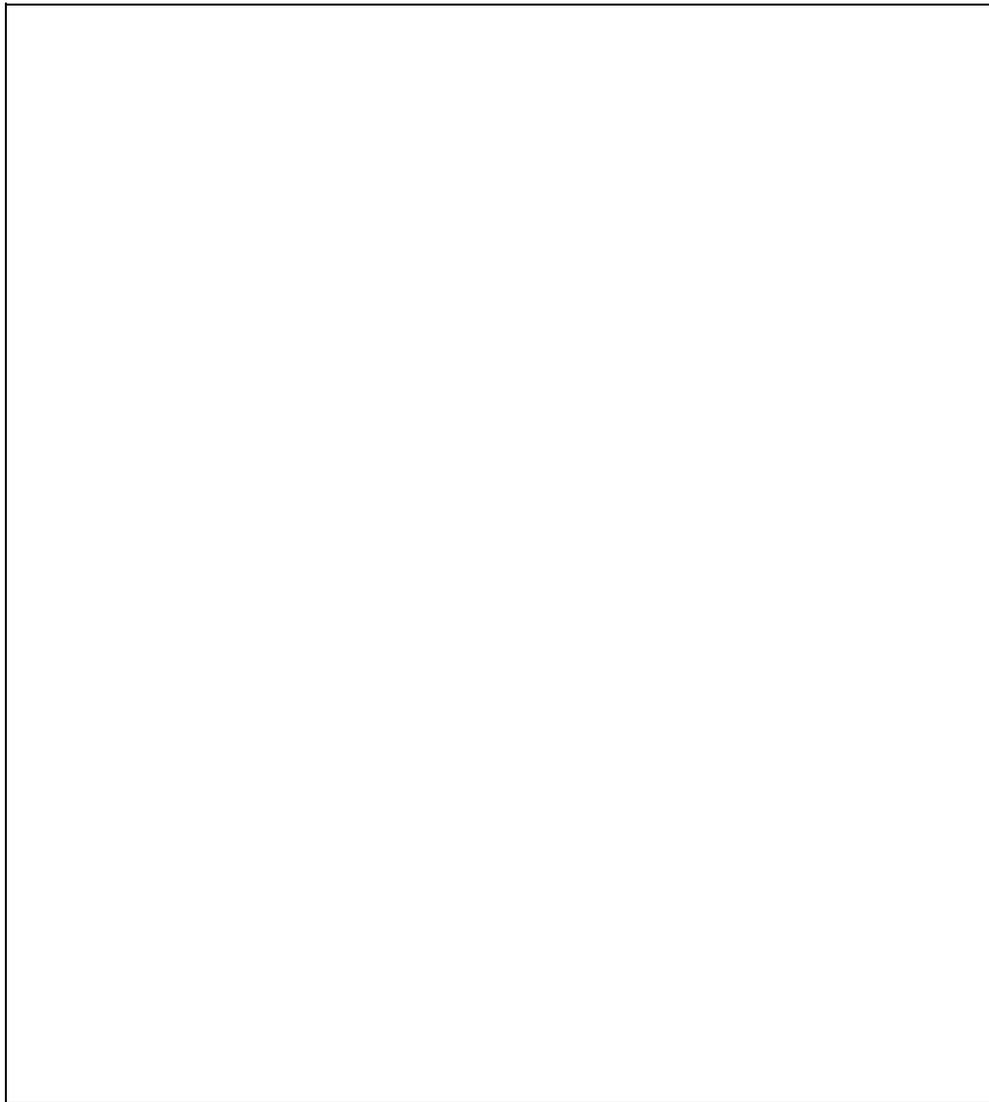
The **Skip and Continue** button will only appear if this option was launched during an in-line build. Pressing this button will tell BuildPlus to discard any changes the user has made and continue on with the build. The **Cancel** button will exit the dialog and stop the build.

To see all of the manual user associations and automatic BuildPlus associations, select **Tools | Project Options** from the menu and click on the **Symbols-Graphics Associations** tab. If you make changes to the associations, rebuild the project after closing the **Project Options** to see your changes in the Snapshot. The associations are stored in the registry for future builds.

4.3.4.4 Database-Graphics Association

This dialog may appear during the build if associations cannot be made using console.cfg files, or

On the **Tools** menu, click **Project Options** and navigate to the **Database-Graphics Association** tab, by clicking on the tab or using **Alt D**.



Any graphics or symbol paths that require association appear in the **Graphics Path** list at the top of the dialog. Symbols paths will be prefaced by "(Symbols)".

To associate databases with a graphics path:

1. Click on a graphics path in the **Graphics Path** list to select it.
2. Select the correct tag and trend databases in the **Tag Database** and **Trend Database** lists. Multiple tag database selections can be made (click-and-drag, CTRL or SHIFT with mouse click) but only one trend database selection.
3. Modify the "Description" (optional) for the Table of Contents entry in the finished Snapshot.

4. Click **Add to List**.

To associate the same databases(s) with multiple graphics paths:

1. Select the correct tag and trend databases in the **Tag Database** and **Trend Database** lists. Multiple tag database selections can be made (click-and-drag, CTRL or SHIFT with mouse click) but only one trend database selection.
2. Click **Add All**.

To view an existing association:

- Expand the entry for the graphics path in the **Final Selections** list.

To edit an existing association:

1. Select the graphics path in the **Final Selections** list.
2. Click **Edit Selection**.
3. Make the changes.
4. Click **Add to List**.

To edit all associations:

1. Click **Edit all**
2. Make the changes.
3. Click **Add All**.

Note that if this dialog was automatically launched during a build, it will time-out after 5 minutes of inactivity and

1. Any user made associations in the final list will be retrieved and stored to the registry.
2. Any graphics paths left unassociated at the timeout will have some automatic associations made:
 1. If a path was originally matched up with a console.cfg file, then the databases listed in the file will be used.
 2. If it is a symbols path, BuildPlus will look for a graphics path that uses that symbols path and use the databases specified for the graphics path.
 3. If it is a GLOBAL graphics path, BuildPlus will use databases in a console.cfg file that is in the same directory path as the global graphics.
 4. Use the first valid pair of tag and/or trend databases listed in the console.cfg files or the user made associations.
 5. Use the largest tag and/or trend databases that have been built.
3. A message will displayed in the main message window stating that default associations have been made and the associations will be shown for the graphics paths that were done automatically.

4.3.4.5 Database Selections - Changing included databases

This dialog may appear during the build if there are multiple tag and/or trend lists, or

On the **Tools** menu, click **Project Options** and navigate to the **Database Selections** tab.



Check those databases that you wish to have included in your build and deselect those that are not relevant to your build.

Check the **Don't show this during the build again** box to prevent the appearance of this dialog.

4.4 After Your First Build

Each time a project is successfully built, you will have

- A DBDOC file, which will open automatically using the Hyperview browser
- Build reports or error files
- Other project files

4.4.1 About Your DBDOC File

A project file is created from each successful build, and can be opened with DBDOC Hyperview.

To open a Snapshot file from BuildPlus, on the **Build** menu, click **View .DBDOC Document**, or click the  button on the Toolbar.

- If there is only one Snapshot, Hyperview is launched to display it, or
- A list of Snapshot files is displayed for the current project. When one Snapshot file is selected, Hyperview is launched to display it.
- If you do not have a project open in BuildPlus, this option will be unavailable.

Alternatively, on the **Tools** menu, select **Launch Hyperview**

- Hyperview will open the most recent Snapshot file, whether or not a project is open.

For more information, please refer to the Hyperview Browser section of this manual.

To share Hyperview and the project file over a network refer to the DBDOC Installation and Setup section of this manual.

A number of options are available within BuildPlus related to Snapshot files, as listed below:

4.4.1.1 Changing how the Snapshot file will be named (BuildRoot option)

On the **Tools** menu, click **Build Root Options**.



Automatic rename of SNAPSHOT.DBDOC

When unchecked, the file will be named SNAPSHOT.DBDOC. When checked, the Snapshot file will be automatically renamed at the end of the build so that it will not be overwritten by future builds. The default is to rename it <Project_name>_<Build_date>.DBDOC.

Omit Project Name (use build date only): With the main renaming option checked, you can choose to leave off the <project_name> and use the date for the *.DBDOC name.

Use Hyperlink version in name: If you are building with different versions of the DBDOC programs, you may want to indicate in the Snapshot filename which version of Hyperlink was used to build it. This will help to clarify which features/fixes were built into that Snapshot when switching between different Hyperlink versions.

Select the date format: This option changes the date format used when creating Snapshot file names.

Default: automatic rename on, with project name and build date but no Hyperlink version

4.4.1.2 Choosing locations for copying the Snapshot after the build

On the **Tools** menu, click **Project Options** and navigate to the **Processing Options** tab.

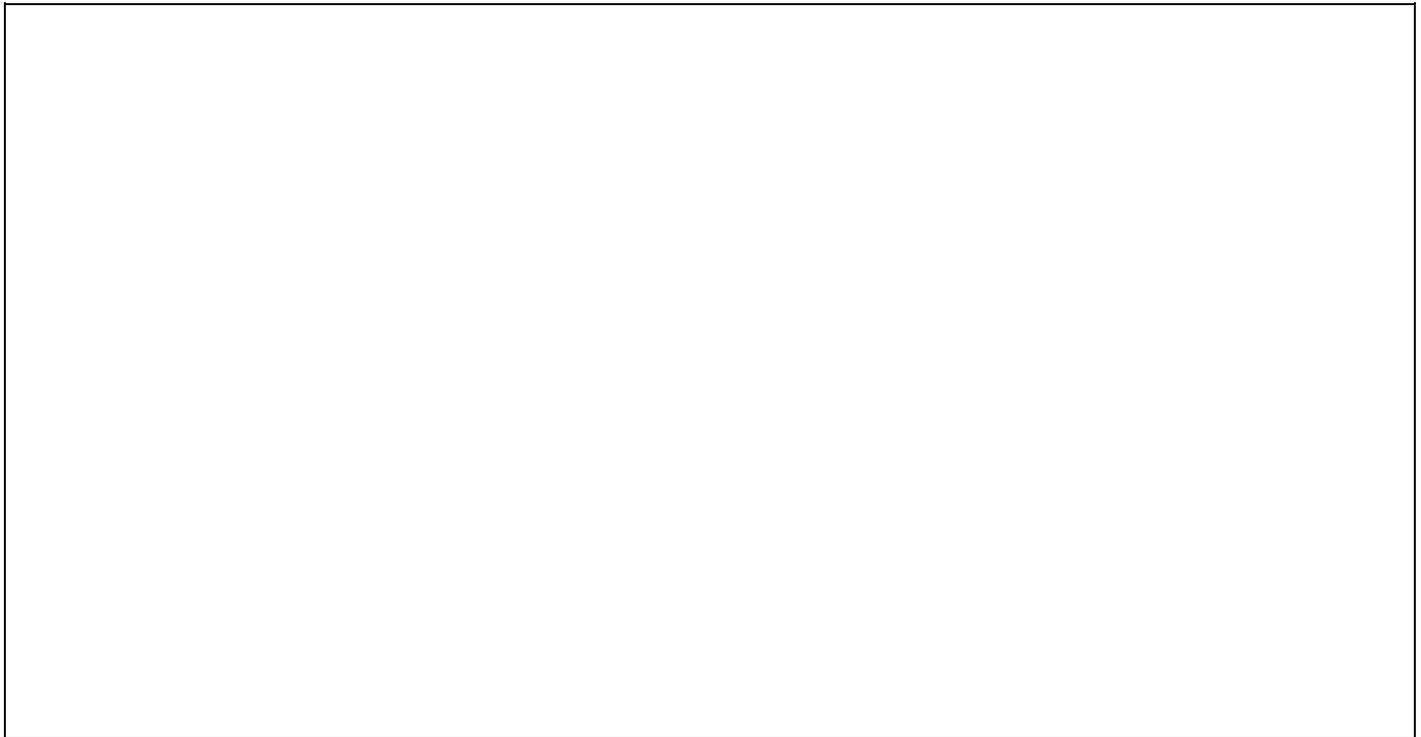


Copy the final Snapshot to the following location

At the completion of a successful build, BuildPlus can automatically transfer a copy of the newly built Snapshot to locations of your choice. Click **Browse** to add each destination directory. To remove a destination, select it from the list, then click **Delete**.

4.4.1.3 Renaming a Snapshot from BuildPlus

On the **Tools** menu, click **Rename .DBDOC/.M14 Document**.



Existing Snapshot files are listed at the top. Choose the file you wish to rename by clicking it, and make changes to the name in the bottom box.

It is recommended that you use this tool when you rename Snapshot files, so that the associated files (including Management of Change and error files) will be updated to refer to the new name.

4.4.1.4 Building bookmarks and annotations into your project file

On the **Tools** menu, click **Project Options** and navigate to the **Hyperview/.Dbdoc Options** tab.



Selecting **Use system settings for Home Page, Text Annotations, Live Loop Annotations, and Bookmarks** allows some forms of "user data" to be included in the .dbdoc project file. In this way, a central "admin" can create bookmarks, home page, and annotations which are automatically available to all regular users of that dbdoc file.

This capability is new in DBDOC 11.6.

4.4.1.4.1 "User" vs "System" Hyperview

There is a new concept of "System" vs "User" Hyperview. The Hyperview launched by BuildPlus is considered to be the "System" Hyperview. All other Hyperviews are "User" Hyperviews. The idea is that the DBDOC admin who builds projects will naturally run the "System" Hyperview, since this is the one which is launched from BuildPlus after builds (and by launching from the BuildPlus toolbar).

If you, as the "System" user/admin, create bookmarks, text or live loop annotations, or a home page for the project viewed in this "System" Hyperview, and then rebuild the project, these user data files will be built into the subsequent build of the project file (in a manner analogous to built-in Live Loop Settings).

When you are in "System" Hyperview, a gear icon will be added on top of the regular Hyperview icons. This will be visible on the desktop toolbar, on the title bar of Hyperview (this will also show "SYSTEM HYPERVIEW" before your project name), and in the "About DBDOC Hyperview" dialog.



4.4.1.4.2 Types of "System" data

4.4.1.4.2.1 System annotations

Creating text or live loop annotations for a project viewed in "System" Hyperview and rebuilding the project will allow for copies of these annotations to show up for all users when they view the corresponding .dbdoc file. System annotations can be viewed in one place using the Annotation Manager in Hyperview. The System annotations will appear on documents in "User" Hyperview as cyan in colour, and are not able to be edited or deleted by users.

4.4.1.4.2.2 System bookmarks

If bookmarks are created for a project viewed in the "System" Hyperview, they will be built into the dbdoc file on the next build. Thereafter, they will appear in the Bookmarks menu of regular users viewing that dbdoc file. The user can choose to hide the System bookmarks, if desired.

4.4.1.4.2.3 System home page

If a home page is defined for a project viewed in the "System" Hyperview, it will be built into the dbdoc file. Regular users viewing the dbdoc file will have this home page automatically set. The user can override the

System home page with a personal local home page if desired.

4.4.2 Launching Windows Explorer to Your Project Files

To access these files from BuildPlus:

- On the **Tools** menu, click **Launch Windows Explorer**.

This will launch Windows Explorer in your project directory, if you have a project open. If no project is open, the directory containing DBDOC_Builds will be opened.

- To find the most recent **.ERR** files, open the **ERROR_Files** folder, then the **MOST RECENT** folder.

4.4.3 Help with Error Files

After a build, open your **ERROR_FILES** folder, then **MOST RECENT**.

For assistance understanding your build reports, please email the **error.7z** file to GMCL for analysis.

If you have any questions, please don't hesitate to contact us!

Email: support@gmcl.com

Toll Free: 1-866-438-2101 in U.S. and Canada

Phone: +1 780 438 2101 outside U.S. and Canada

4.4.4 Sending an Error.7z File to GMCL

After completing a new build, you can email us your zipped files - we'll interpret the files and let you know if DBDOC has identified any potential problems in your system. Since we are constantly refining and expanding DBDOC's error checking capabilities, we strongly encourage our clients to send us a file at least once a year. The latest DBDOC can detect system issues undetectable with earlier versions, and not checked by ABB tools.

After a build, launch BuildPlus, and follow the following steps:

- On the **Tools** menu, choose **Launch Windows Explorer**.
- In the **ERROR_FILES** folder, find "error.7z".
- Attach the file to an email and send it to support@gmcl.com

4.4.5 Exiting BuildPlus

On the **File** menu, click **Exit**. You can also click the Close button (**X**) in the upper right-hand corner of the window.

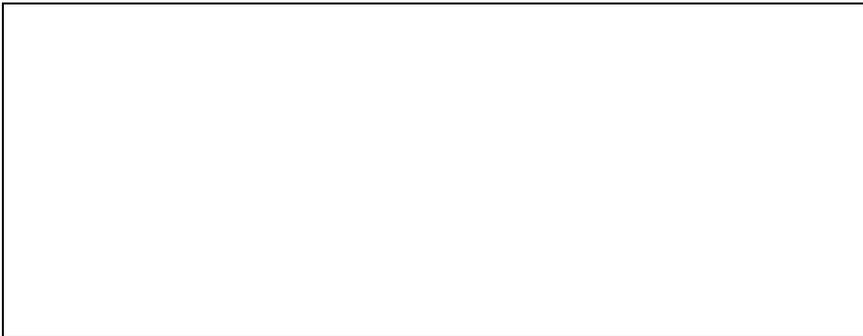
If a build is underway, an Exit Query will appear, and you can choose if you wish to exit or let the build finish.

4.5 Rebuilding

You may choose to rebuild your project

- when you know your files have been modified
- to change the data files you build in
- after a DBDOC update, or with new versions of some DBDOC components
- with different BuildPlus options
- regularly, using scheduled builds

To prevent multiple simultaneous builds of the same project, you may see the following dialog when you open BuildPlus to rebuild. Click **OK**, then find the original BuildPlus window or open a new project.

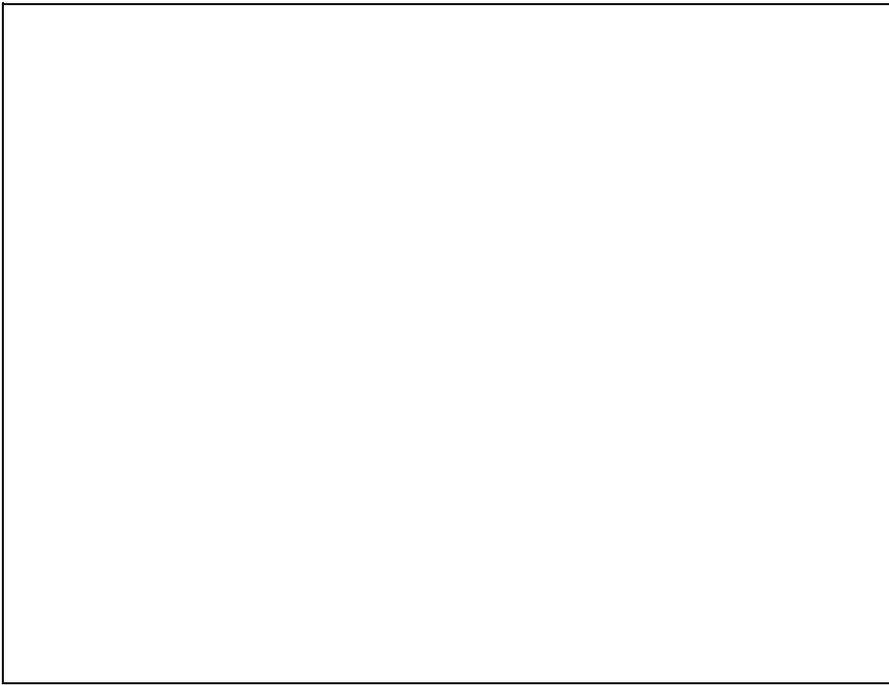


After rebuilding, you may want to compare builds to see the effects of the changes. This can be done using Winmerge, as described below.

4.5.1 When your files have been modified

When you want to rebuild your project just click the **Build** button and BuildPlus will detect any changes in your databases and find any new files in your existing search directories.

If new module header files and CAD or CLD files are detected, they will be listed for you and you will be asked if you wish to include them in your present build.



All new files/folders will be automatically checked for inclusion in the build. Uncheck the box to the left of the selection to exclude them from the build.

If no action is taken within five minutes, all newly found modules folders will be automatically included in the build. For new CAD or CLD modules, BuildPlus will again check for any duplicate module headers that may conflict with previous module choices and ask that they be resolved before the build continues.

BuildPlus will also check for new graphics folders. If new files or folders are detected, they will be listed for you and you will be asked if you wish to include them in your present build.



All new files/folders will be automatically checked for inclusion in the build. Uncheck the box to the left of the selection to exclude it from the build.

If no action is taken within five minutes, all newly found graphics folders will be automatically included in the build. If you wish to make changes to your MHD or SODG graphics selections after finishing the Auto-Detection process, you will need to restart the build from the Wizard stage.

When new 800xA folders have been extracted, you may choose to include them in the build and whether they are graphics or submodels:



The top window lists any 800xA folders found, that need to be categorized as graphics or submodels. The bottom two windows list BuildPlus automatic choices about graphics and submodels folders.

- To modify the choices in the lower two boxes, select a folder by clicking on it, and click **Remove** to move it temporarily to the top box.
- To move a folder from the top box, select it, then you may **Explore** to the source of the folder, or **Add as a Graphic** or **Add as a Submodel**.

This box will time out after 5 minutes of inactivity, and proceed with the build. 800xA folders may be selected or deselected later using Project Wizard.

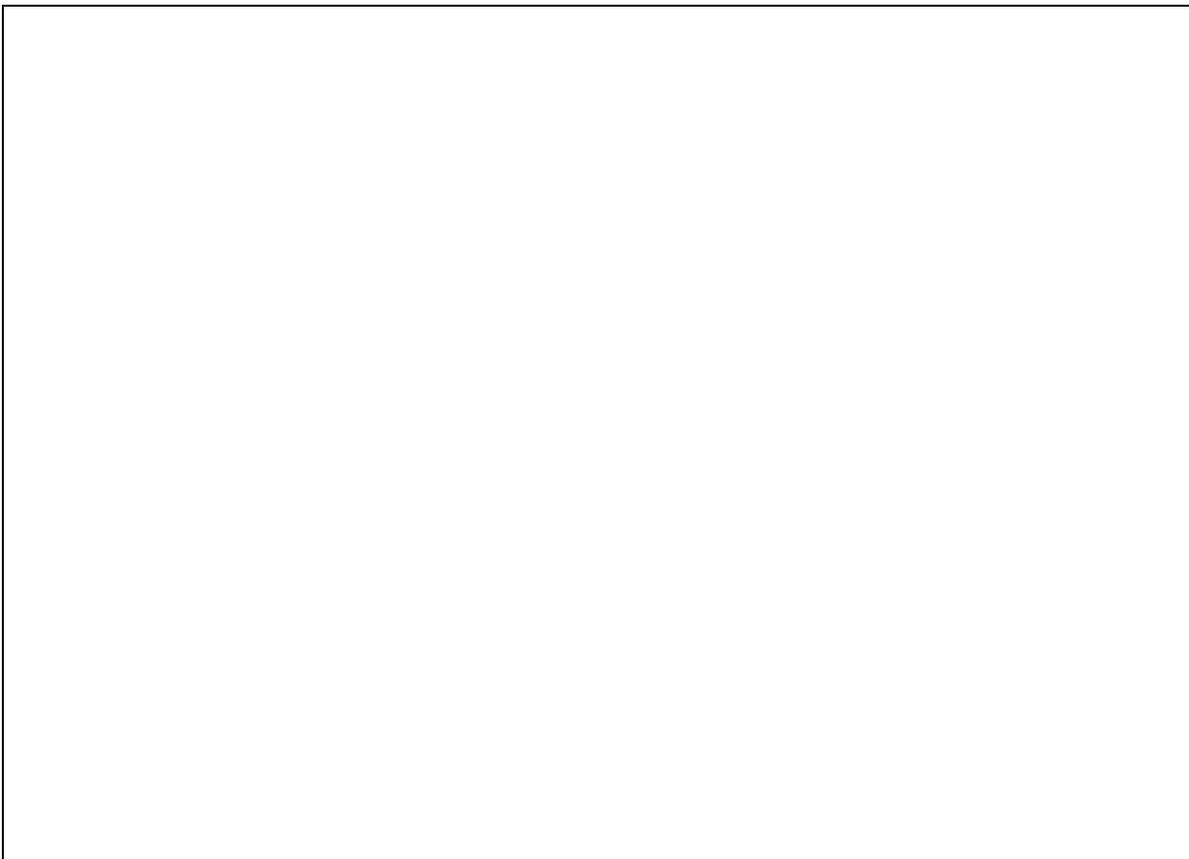
4.5.2 Changing Your Data File Sources

To modify your database selections or the location of database or 800xA graphics files, use the **System File Specifications** button  on the toolbar or **System File Specification** under the **Build** menu.

To change or add other files that you have chosen to include in your build, use the **Wizard**  button on the toolbar or **Project Wizard** under the **Build** menu.

If your data files have been moved to a new computer, go to the **Tools** menu, and choose **Revise Source Locations...**

Choosing to Revise Source Locations opens a dialog listing the current (original) paths to each of your file types.



To change the location of one file type, click on the path, then choose **Modify**. This opens the **New Source Path dialog where you can enter the new source path, or browse to it, then choose OK. Once a new source path is specified, it is added to the drop-down list in case you'd like to use it for another path replacement.** To change the locations of all file types, right-click on one path, and choose to Select all, and **Modify**. The New Source Path dialog will open, attempting to list the common prefix of the path to be replaced. If all paths are to be replaced with the new source path specified, you may check the box **Replace the entire path with this one**. Once some changes have been initiated, they are listed under New Path, with green text for valid paths and red for paths that do not exist. To remove entries in the New Path column,

select those paths then click **Clear Changes** and to keep all new paths, click **Apply** to apply the changes.

4.5.3 After a DBDOC Update

To rebuild when you have received newer versions of DBDOC program components, on the **Build** menu, click **Build**, or click the button on the toolbar. This starts a complete build of your project, without changing any of the files or options already selected.

4.5.4 Changing Options

Within BuildPlus, there are

- **Project options**, that are applied to the current project;
- **BuildRoot options**, that apply to all the projects in the BuildRoot, and
- **Global options**, that apply to all projects in all BuildRoots.

Each set of options can be accessed from BuildPlus under the **Tools** menu. For more information see BuildPlus help.

4.5.5 Scheduled Builds

Running scheduled builds allows builds to occur during your off hours. Automated builds on a regular basis ensure that the most recent files are included. Interactive message boxes are suppressed on scheduled builds and files are cleaned up automatically. To learn about scheduled builds, see BuildPlus Help or contact GMCL.

4.5.6 Command line builds

To learn about building from the command line, see BuildPlus Help or contact GMCL.

4.5.7 Comparing Build Results with Winmerge

WinMerge is an open-source tool that allows you to specify two files or two directories and do a byte-for-byte file comparison. The result of the comparison is displayed in a window, showing you any changes that may exist between the two files or directories. This is a useful tool to compare differences in your error files between builds.

If you have WinMerge installed on your computer, BuildPlus will enable this option for you. You are given the choice to either Compare Files or Compare Directories. In either case, a dialog will appear where you can either type in or browse for the appropriate files/directories for comparison.

Note: Earlier versions of BuildPlus use **WinDiff** instead of **WinMerge**.

- To compare files, on the **Tools** menu, select **Compare with WinMerge** and click **Compare Files**.

(The default selections are the DBDOC_SUMMARY.ERR files of the two most recent builds.) For each file you wish to compare, click **Browse**, navigate to the file, and click **OK**. Once both file paths have been entered, click **OK**. This will launch WinMerge to show differences between the files.

- To compare directories, on the **Tools** menu, select **Compare with WinMerge** and click **Compare Directories**.

(The default selections are the error file folders for the two most recent successful builds.) For each directory you wish to compare, click **Browse**, navigate to the folder, and click **OK**. Once both directory paths have been entered, click **OK**. This will launch WinMerge and show which files are different within this folder. Double-clicking on the file name will show the differences within individual files.

4.6 More About BuildPlus

4.6.1 Opening a recent project

On the **File** menu, click **Recent Projects**

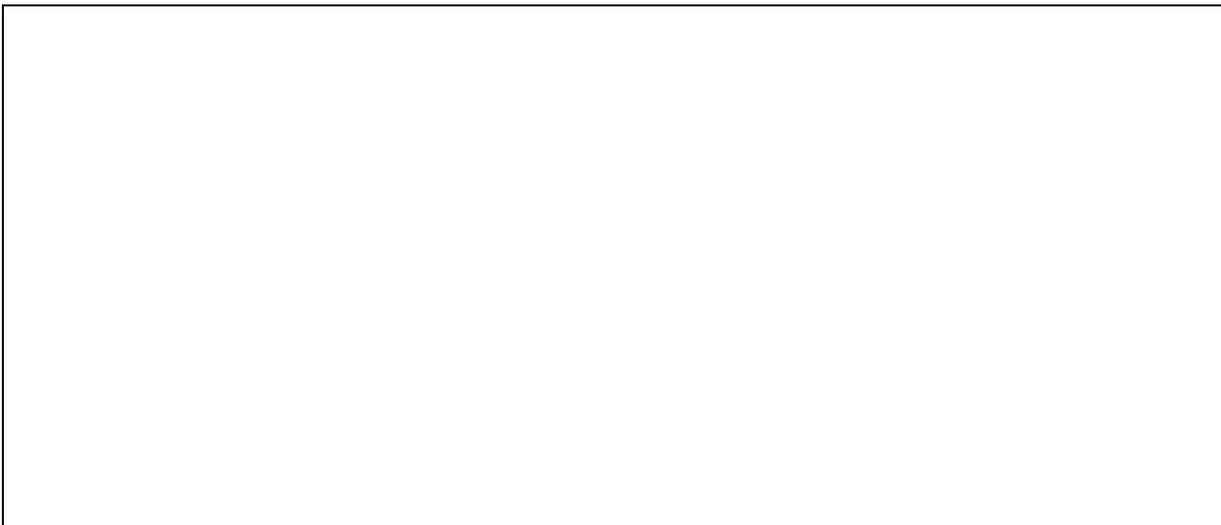
This lists your 20 most recent projects, so you can click on your project name.

Note that when BuildPlus starts, it opens the last project you had open. If you only have one project, it will always be opened automatically.

4.6.2 Renaming a project

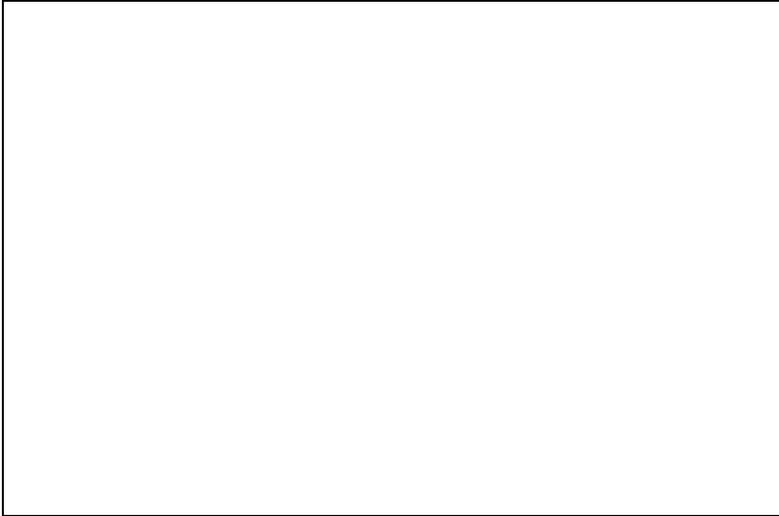
On the **Tools** menu, click **Rename Current Project**.

This allows you to rename the current project, and move it to a different build root if you choose.



Enter the changed name and build root (the same or different) for the project. The current build root will be selected when the window opens.

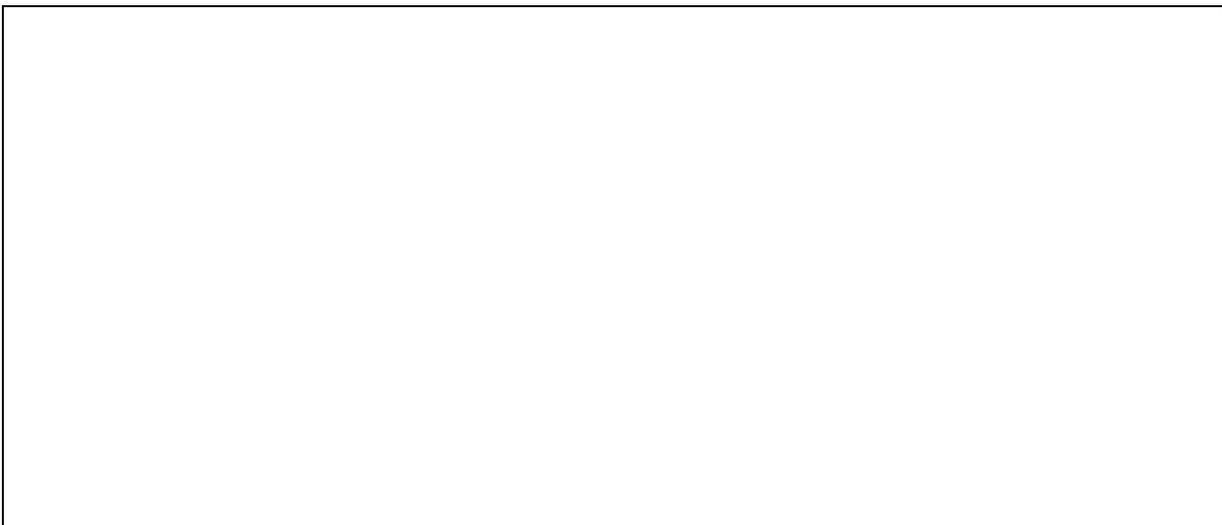
- A project can also be renamed by pressing the **System File Specifications** button and in the **System File Specification** dialog type in the new name of the project in the **Enter name of the project** edit box. Click the **Next** button (or **Finish**, if applicable) when you are satisfied with your new name. You will then see the following dialog: To rename the project, select **I want to rename my existing project to the new name I entered** and click **OK**.



4.6.3 Moving a project

On the **Tools** menu, click **Move Current Project**.

This allows you to move the current project to a different build root, and rename it if you choose.



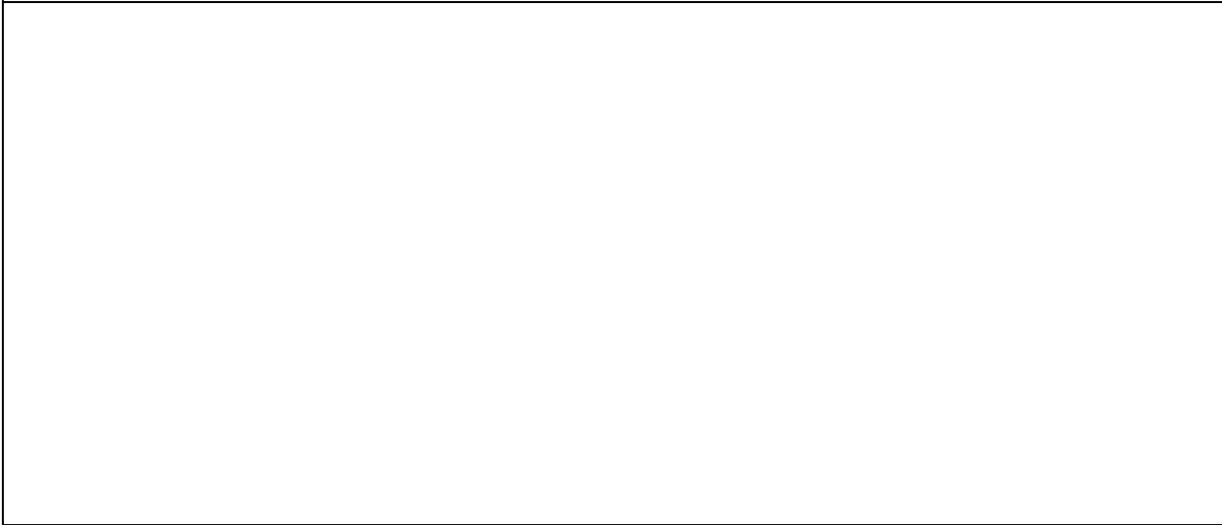
Enter the build root where the project should be moved, and the name for the project (the same or different). The current build root will be selected when the window opens.

4.6.4 Cloning a project

On the **Tools** menu, click **Clone Current Project**.

This allows you to make a copy of the current project either

- in a different build root.
- or with a different name in the same build root.



Enter the build root and name for the copy. The copy must have a different name, different build root, or both.

4.6.5 Deleting a project

On the **File** menu, move to **Delete Project**. Select the build root and project you wish to delete.

When asked "Are you sure?" if you click **Yes**, the project will be deleted.

Most of the files created by BuildPlus will be deleted, and the project will be removed from the BuildPlus menu. A message will appear in BuildPlus advising you to manually delete the project folder and remaining files.

4.6.6 Exporting a project

On the **Tools** menu, click **Export Project**.

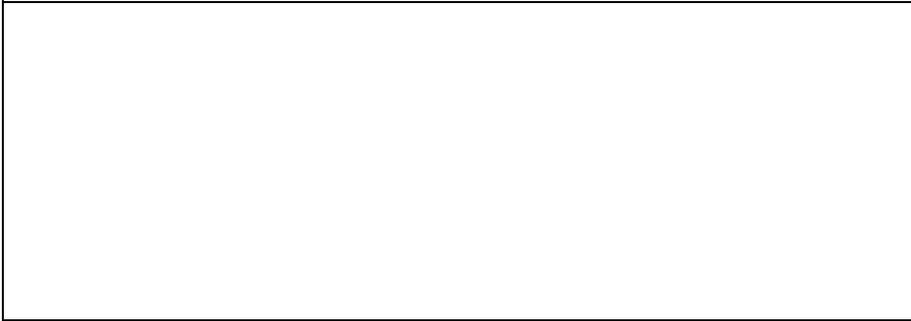
This option allows you to export the current project so it can be imported into another installation of BuildPlus, for example on another computer.

A BuildPlus import file (.bpi) is created for the current project. The import file will be named [project name] .bpi and it will be created in the project's working folder.

This works best if the search paths used when creating the project were in UNC form (eg `\\datacomputer\sharedfolder`), so that the other installation will be able to find the files, even if you import the project on another computer. (A message will appear explaining this the first time you export a project.)

4.6.7 Importing a project

On the **Tools** menu, click **Import Project**.



This option creates a new project from a BuildPlus import file (.bpi), as created with **Export Project**.

You will need to enter the location of the import file, the build root to create the project in, and a name for the project.

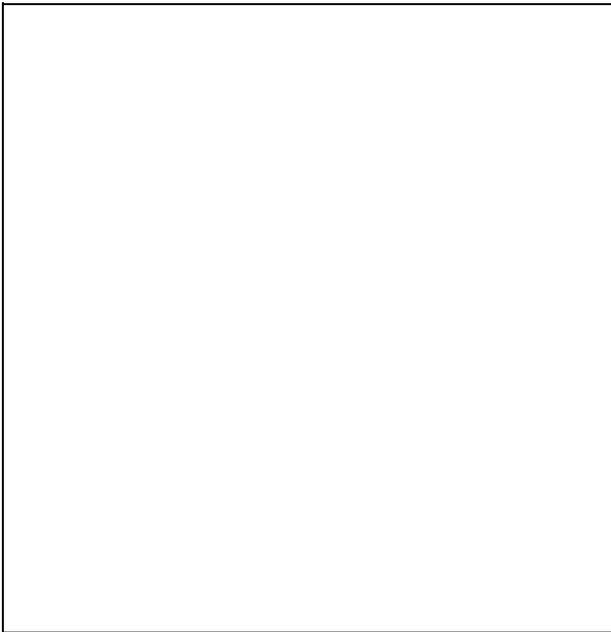
If the project that was exported used UNC paths (eg `\\datacomputer\sharedfolder`), then you should be able to build the new project without any problems by opening it and clicking the **Build** button.

4.6.8 Reverting to a previous build

On the **Tools** menu, click **Revert to a Previous Build**.

This allows you to load file choices and project options used in a previous build.

Choosing one of the folders named `[project name]_[date]` allows BuildPlus to find the `[project_name].nod` file within that folder, and revert the project to file choices and project options from that date.

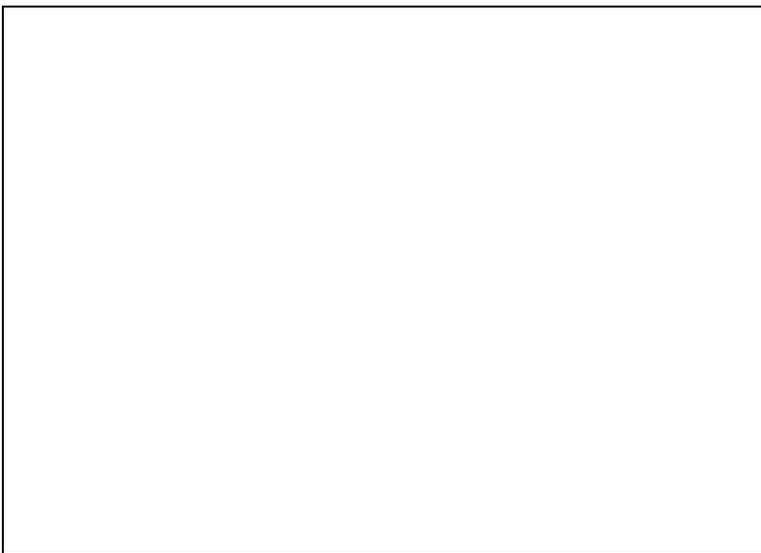


BuildPlus will find the .nod file within that folder (or notify you if it doesn't exist).

4.6.9 Cleaning up temporary/intermediate files

By default, BuildPlus deletes temporary, intermediate files at the completion of each build.

- To change this default setting, you can choose whether to automatically clean up temporary files for all projects in a build root: On the **Tools** menu, click **Build Root Options**, and select **Automatically clean up files upon completion of build**. When this option is on, BuildPlus deletes temporary files automatically when the build finishes. If this option is off, BuildPlus will ask whether to delete the files after each build.

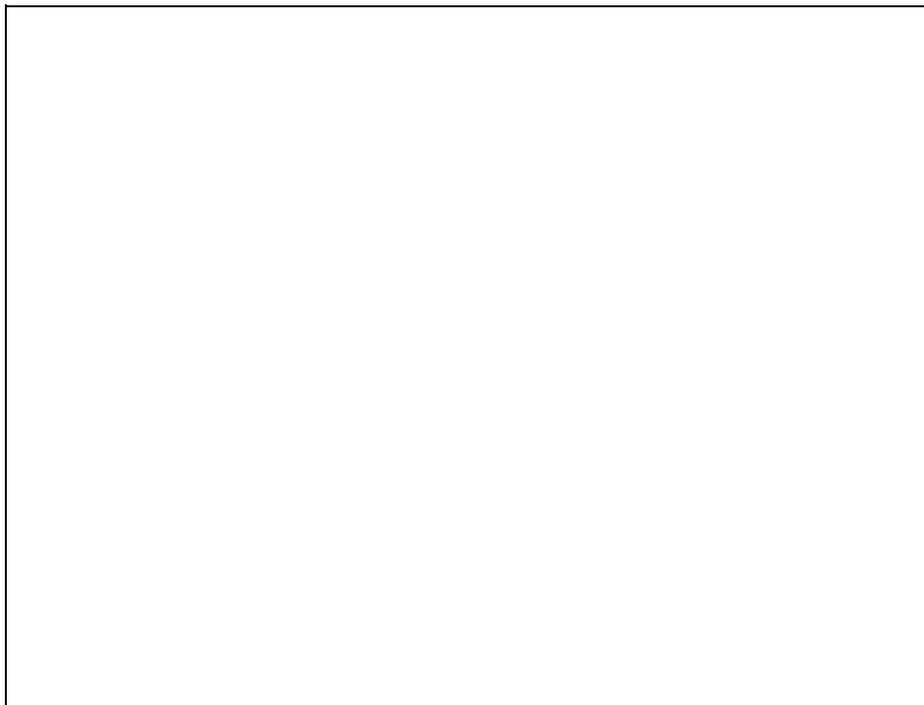


- If files were not cleaned up by default, you can clean up temporary/intermediate files for the current project: On the **Build** menu, click **Clean Intermediate Files**. This deletes temporary files that are not

needed after the DBDOC build is complete. (Typically, the temporary files will exceed the size of the Snapshot file, so this is worthwhile.)

4.6.10 Sibling Projects

If you have more than one project in a BuildRoot, the Sibling Projects dialog may appear, where you can select related projects to check their databases for additional sources for tags.



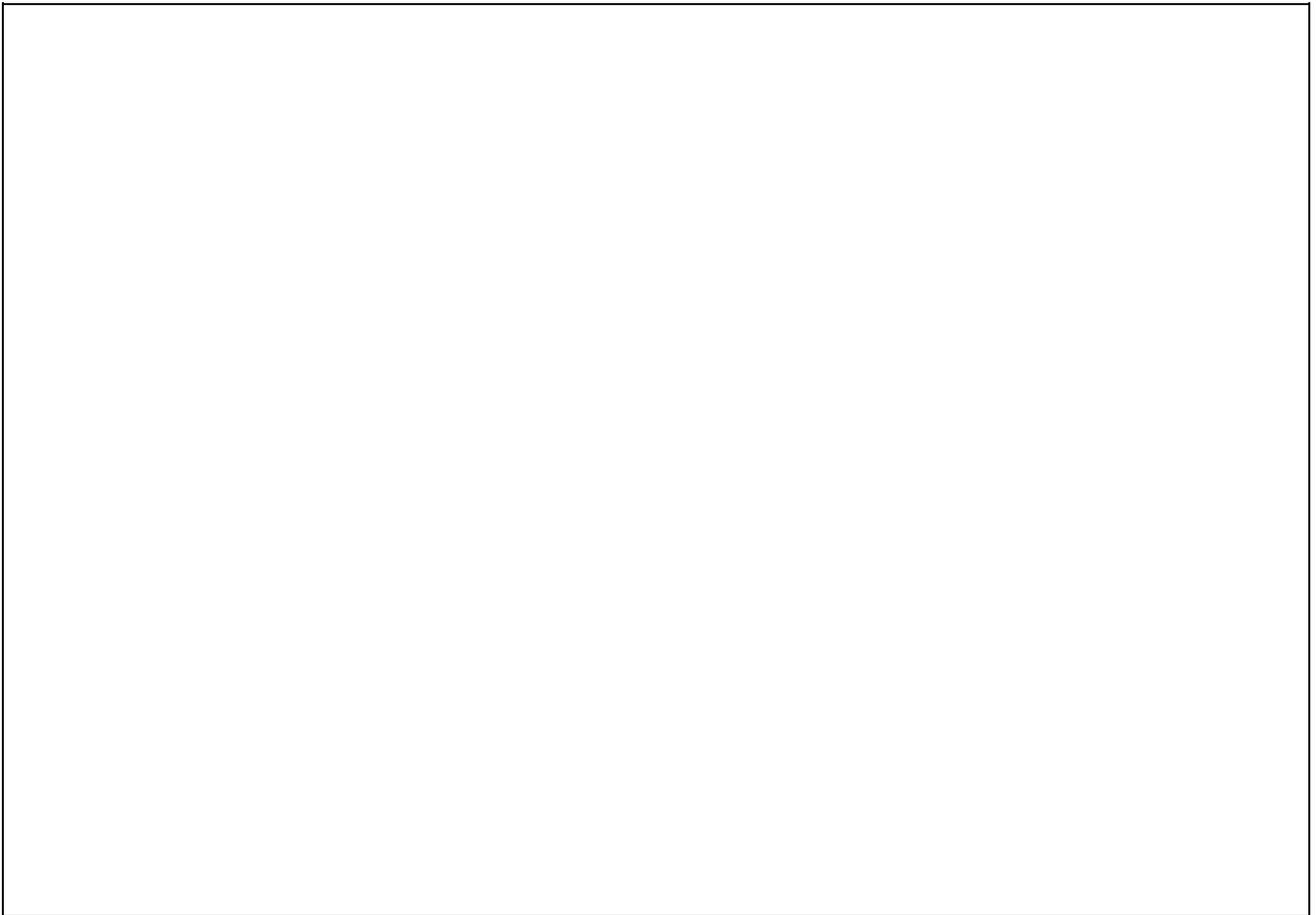
Check or uncheck the boxes beside the project names to include or exclude those projects as sibling projects for future builds of the current project.

If you prefer to no longer be asked about sibling projects during a build, check the box for **Don't show this again**. To change sibling project choices at another time, on the **Tools** menu, click **Project Options** and navigate to the **Define Blocks With External Sources** tab. More information is also available in BuildPlus help documentation at Sibling Projects (Global option).

4.6.11 BuildPlus User Interface

BuildPlus has two sub windows:

- **The Main Message Window** (usually on top): Displays information about processes BuildPlus is executing or has just completed and any errors that may have occurred. Text appearing in red may indicate problems with the build.
- **The Sub-Process Message Window** (usually on the bottom): Displays the messages generated by an external program that has been launched by BuildPlus.



To change how the windows are displayed, click on: **Window** on the BuildPlus main menu. To reopen a closed or minimized window, click on: **View** on the BuildPlus main menu. Your settings (viewing preference, size and position) are stored in the registry for the next time BuildPlus is opened.

The **BuildPlus status bar** shows basic information about what BuildPlus is doing:



- **BP Active:** BuildPlus is running an internal process.

- **EXT Active:** BuildPlus is running an external process.

5 Basics of CIUMon

5.1 Overview



With a connection to your CIU, Hyperview can:

- display current values on CAD sheets and graphics
- display selected block values in annotations on any document
- track selected block values and graph them
- display the status of selected modules
- display values from any part of your system with a connection to a CIU on any loop

Hyperview gets live data through CIUMon. CIUMon handles all communication with the CIU. It provides live data over your network, and can provide live data to multiple users at the same time.

CIUMon cannot provide live data from a PI system, although information about your PI system can be included in your project.

5.1.1 Technical requirements

CIUMon is an interface program that provides read-only access to a CIU interface, via TCP/IP. This program conforms to the specification provided by the *Enhanced Computer Interface Unit Programmer's Reference Manual*, INFI 90 E93-905-9, with extensions that we have discovered. CIUMon can run on Windows 7, Windows 2008 R2, as well as all later versions of Windows. (For older versions of Windows such as Windows XP, Windows NT4, or Windows 2000, we can provide archived versions of CIUMon that will work but do not have the latest features.) The requirements to run CIUMon are as follows:

- The host and client computers must be running a TCP/IP stack.
- For Windows PCs, the preferred operating system and stack is the native Microsoft Network that is provided with Microsoft Windows operating systems.

CIUMon allows you to collect live data from your plant loop:

- CIUMon uses the maintenance interface available to your EWS, which is polled. It can fetch 10-20 values per second.
- CIUMon cannot replace a console, as it has no access to exception reports, alarms or history.
- You can increase throughput with multiple SPMs, or by using RoviSys Turbo.
- CIUMon keeps track of the incoming requests from all clients. If a data request comes in while there is still an outstanding one on the CIU for the same block, CIUMon will not send the request again. Instead, when CIUMon receives an answer to its original request, it will then send that value back to all clients that requested that particular piece of information.

CIUMon communicates via UDP (User Datagram Protocol)

- UDP is a data unreliable connectionless protocol
- Its robustness and tolerance of network connection problems that may be encountered make it ideal for use in CIUMon
- It reduces the likelihood of errors that may come with more complicated communication protocols

5.2 Methods of connecting CIUMon to a CIU

Once CIUMon is installed, decide which method of connecting to a CIU is appropriate for your setup. Then, connect to your CIU either directly, through RoviSys Turbo, or through Harmony OPC Server. Finally, set up CIUMon to run automatically.

CIUMon can be connected directly to some types of CIU.



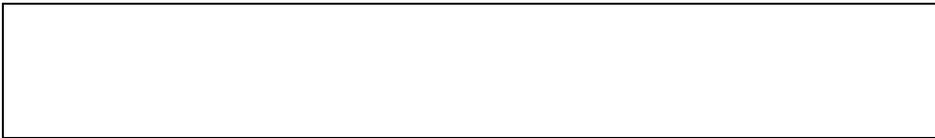
For other types of CIU, CIUMon must be connected through RoviSys Turbo.



Connecting through RoviSys Turbo also allows a CIU to be shared between CIUMon and other programs, such as WinCAD or Composer.



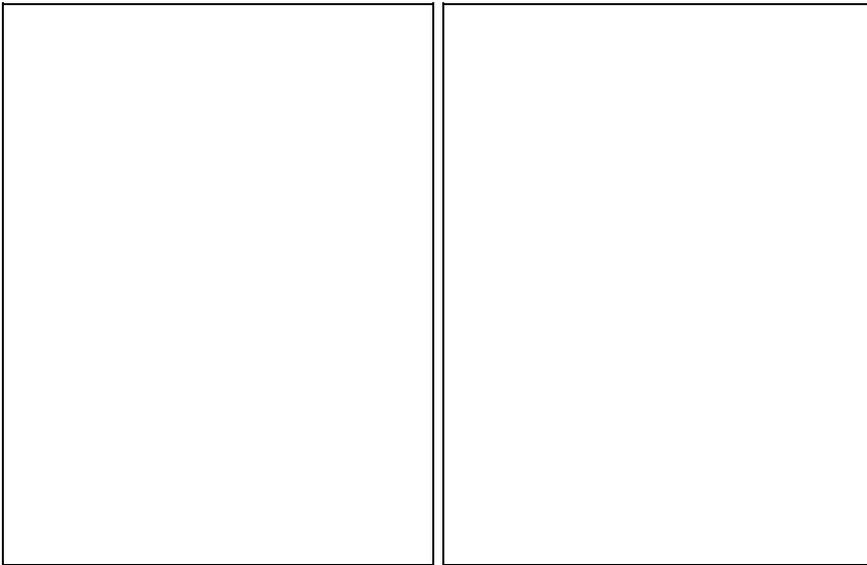
RoviSys Turbo can be connected via **serial port** or via **ethernet** to an IET800 or PNI800 ethernet CIU. This lets DBDOC take advantage of RoviSys Turbo's turbo mode features. Contact us for instructions on how to set this up.



Alternatively, you can create special DBDOC blocks in Harmony OPC Server to allow DBDOC to connect to an IET800 ethernet CIU over your network. (However, because this method does not allow you to use RoviSys Turbo's turbo mode features, it will typically provide less throughput than using RoviSys Turbo to connect either to the ethernet CIU or to another serial or SCSI CIU.)



5.3 Connecting to a CIU



To connect directly to a CIU:

1. On the **Options** menu, click **CIUMon Settings**. Choose a UDP port, and select your CIU type.
2. Click the **Communication Modes** tab, and click **Connect directly through serial port**.
3. Click the **Comm and Flow Control Settings** tab, and select the serial port connected to the CIU in the **Serial Port** list.

Note: If EWS software, such as Composer or WinCAD, is set up on the same computer, then you may wish to move the serial connection to a different COM port. This way you do not have to change any settings in the EWS software, but that software and CIUMon are not trying to access the same port.

4. If you know that any of the other settings need to be changed for your setup, change them now. Usually the defaults are fine.
5. Click **OK** to close the Options window. Then, make sure you can get data from your CIU.

5.3.1 Choosing a port to get data from CIUMon

1. On the **Options** menu, click **Settings**. Click the **Configuration** tab.
2. In the **UDP Port** box, type the UDP port for CIUMon to use.

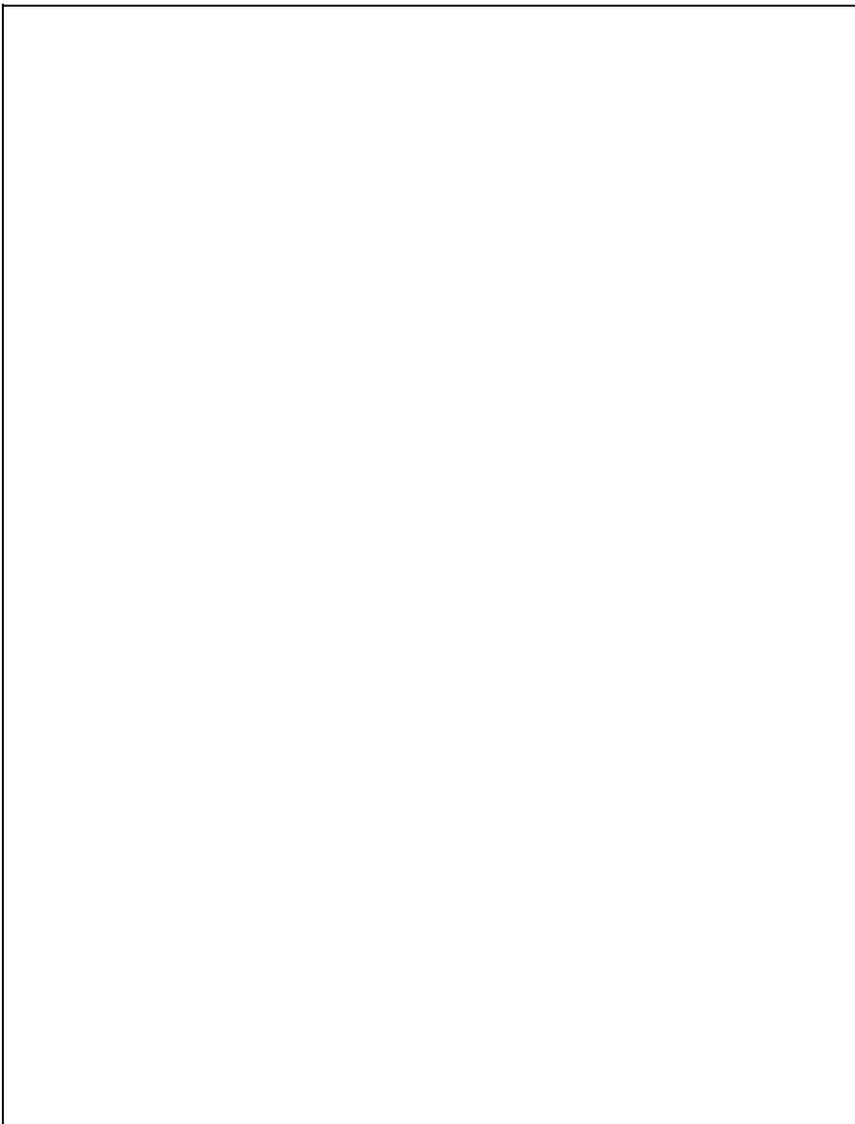
Note: Usually, the default port 5099 should be used. But if multiple instances of CIUMon will be running on the same computer, they must use different UDP ports. CIUMon will warn you if you try to set a port that matches what is already in another configuration.

3. Make a note of the UDP port that you've chosen. You will need this number in order to configure Hyperview to get live data.
4. Click **OK** to close the **Options** window.

5.3.2 Configuring CIUMon for your CIU type

1. On the **Options** menu, click **CIUMon Settings**. Click the **Configuration** tab.
2. In the **CIU Type** box, click your CIU type.
3. If any of the following special situations apply, change the settings as needed:
 - ◆ If you are connecting to a serial port module (SPM01 or SPM02), type the loop number in the **Loop** box and the PCU number in the **PCU** box. These parameters tell CIUMon which Loop and PCU the serial port module is connected to.
 - ◇ CIUMon will send an environment fetch to get the loop and PCU from an SPM02, but the fetched loop and PCU will be meaningless. CIUMon must use the specified loop and PCU to know what requests to send to the serial port module.
4. Click **OK** to close the **Options** window.

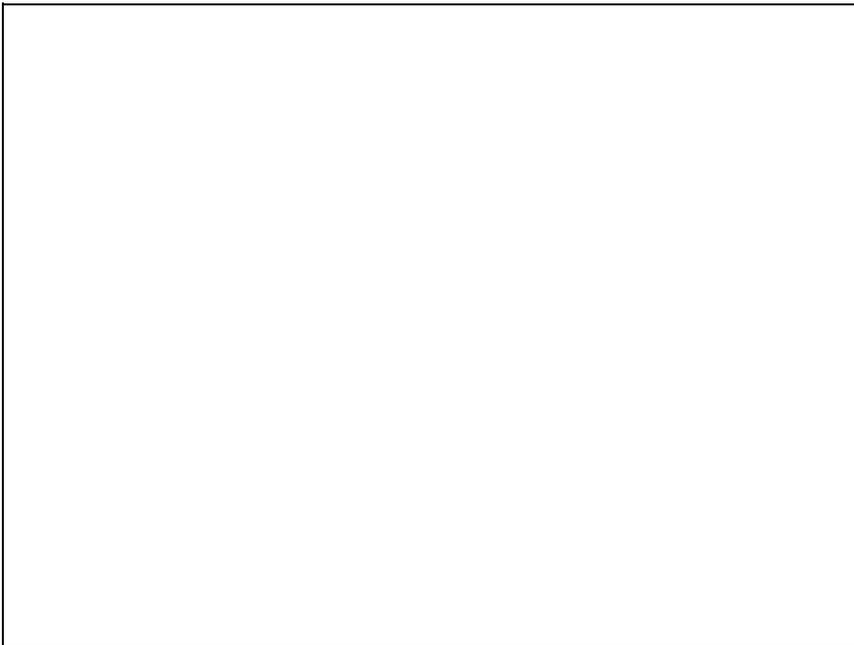
5.3.3 Connecting to a TRAX emulator



1. On the **Options** menu, click **Settings**.
2. Click the **Communication Modes** tab.
3. Click **Connect to a TRAX CIU emulator**.
4. Type the IP address of the computer running the Trax emulator in the **TRAX computer name or IP address** box.

Note: When you disconnect CIUMon or another program from the Trax emulator, you will need to wait for approximately four minutes before you can connect to the emulator again. This is because of how Trax handles its network sockets.

5.3.4 Making sure you can get data from CIUMon



Once you have configured CIUMon to connect to your CIU, you should test it to make sure your configuration works. The best end-to-end test is to connect Hyperview and make sure you can get data.

1. If the **Options** window is still open, click **OK** to close it.
2. On the **File** menu, click **Connect**. CIUMon should display some information, including your CIU's ring and node. The word "OK" should appear in green, on a line by itself.
3. Using Hyperview, check to make sure you can access live data. See Live Data Configuration in Hyperview and Displaying Slow Mode live data in Hyperview.
4. Once CIUMon is connected to your CIU, you may wish to:
 - ◆ Run CIUMon automatically (as a service)
 - ◆ Build the connection settings into your project

5.3.5 Running CIUMon manually

Most of the time, you should run CIUMon automatically. However, there may be occasions where you need to run it manually.

To run CIUMon manually:

1. Start CIUMon. On the **Start** menu, point to **All Programs**, then point to **GMCL DBDOC**, and then click **Start CIUMon Online Data Server**.
2. If you have more than one configuration, CIUMon will ask you to choose which configuration you want to load. Select the appropriate configuration, and click **OK**.
3. CIUMon may connect automatically, if it is configured to do so. (To configure CIUMon to connect automatically: On the **Options** menu, click **Settings**; and then select "Auto Connect".) If it does not connect automatically, then on the **File** menu, click **Connect**.

Note: If you are running CIUMon as a service, you will not be able to run the same configuration manually unless you stop the service first.

When CIUMon is run manually or interacting with the desktop, you can use the CIUMon menus to interact with CIUMon.

You can use the menus to connect or disconnect from the CIU, clear the displayed messages, check the connection status, or change CIUMon's settings.

To connect to the CIU:

- On the **File** menu, click **Connect**.

To disconnect from the CIU:

- On the **File** menu, click **Disconnect**.

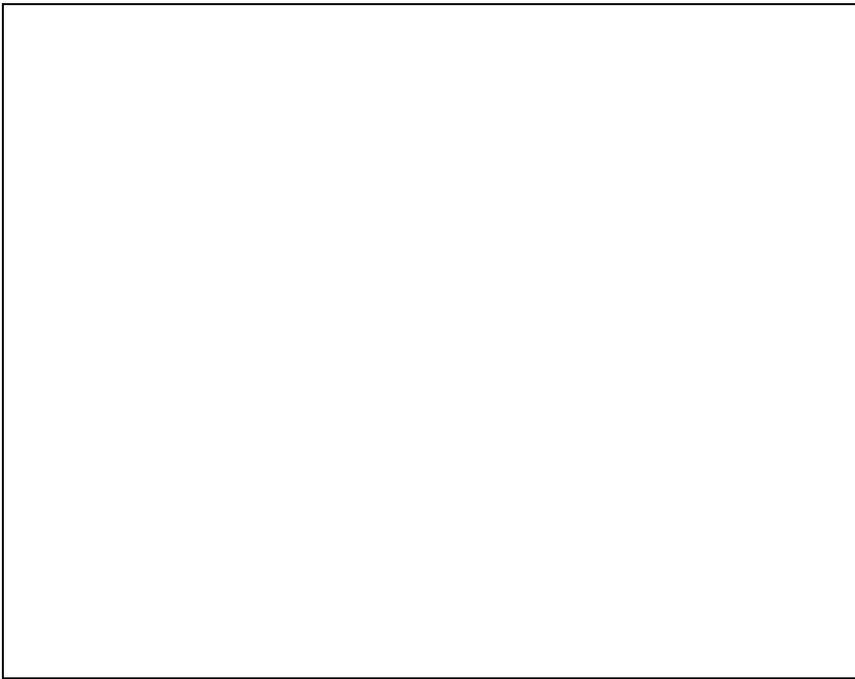
To clear the displayed messages:

- On the **File** menu, click **Clear Window**.

To change CIUMon's settings:

1. On the **Options** menu, click **CIUMon Settings**.
2. Change the settings as needed.
3. To make the changes permanent, save the configuration.

5.4 Connecting to RoviSys Turbo



Follow these steps to connect DBDOC to your CIU using RoviSys Turbo:

1. Install RoviSys Turbo on a computer that connects to a CIU.
2. Connect RoviSys Turbo to your CIU.
3. Connect CIUMon to RoviSys Turbo.
4. Set up RoviSys Turbo to accept connections from any programs CIUMon is sharing the CIU with (such as WinCAD or Composer).

Note: These instructions are for RoviSys Turbo version 7.2 or later. If you are using an older version of RoviSys Turbo, we recommend you upgrade. If you must connect CIUMon to an older version of RoviSys Turbo, contact us for instructions.

5.4.1 Installing RoviSys Turbo

A trial version of RoviSys Turbo is included with DBDOC. The installer for the trial version is located in either `C:\Program Files\DBDOC\PROGRAMS\RoviSys\` or `C:\DBDOC\CIUMON\Programs\RoviSys\`, depending on how CIUMon was installed.

Note: The RoviSys Turbo demo will run for up to one hour at a time, and it can be configured to allow for a one-time 45-day trial. The demo on the DBDOC install DVD might not be the most recent version. To get the newest and best version of the demo, download it from <http://www.rovisys.com>. There is a link to the RoviSys Turbo demo download page on <https://www.gmcl.com/downloads.php>.

To continue using RoviSys Turbo, you will need to purchase a license from RoviSys. Please contact us for information. You may also contact RoviSys directly through their web site at <http://www.rovisys.com>.

Note: You may need to uninstall the demo version of RoviSys Turbo when upgrading to the full version.

Upgrading to a newer version of RoviSys Turbo may also require you to uninstall the old version. Before uninstalling RoviSys Turbo, we recommend backing up your configuration: in RoviSys Turbo, on the **File** menu, click **Save As** and save a copy of the configuration file to a backup folder.

5.4.2 Connecting RoviSys Turbo to a CIU

1. On RoviSys Turbo's **File** menu, click **Open**. Open DBDOC.cfg, which is in **C:\Program Files\OPC90 Server\CFG** if RoviSys Turbo is installed in the default location.
2. Right-click **CIU** in the left-hand pane, then click **Properties**.
3. In the **Primary Port** list, click the COM, SCSI, or TCP port that is connected to the CIU.

Note: If you are using RoviSys Turbo on a computer with EWS software, you might want to move the CIU to a different port to avoid changing the EWS software configuration.

4. Make any other changes you need, in order to configure how RoviSys Turbo communicates with your CIU. Click **OK**.
5. If the CIU is connected to a COM or TCP port, set up that port. On the **Edit** menu, click **COM Ports** or **TCP Ports**. In the **Port** list, click the port that is connected to the CIU. Set the baud rate (for COM ports), IP address (for TCP ports), or other settings needed to connect to the CIU. Click **OK**.

Note: RoviSys Turbo COM ports default to 19200 Baud with other settings that are normally correct. For TCP ports, refer to the RoviSys OPC90 manual for details about how to configure a TCP connection to an IET800 or PNI800 CIU.

6. If you want to log data, right-click CIU and select each of the first four logging options.
7. On the **File** menu, click **Save**.

Note: If you are using RoviSys Turbo on a computer with User Account Control (UAC) enabled, you may not be able to save the configuration in the default folder. Choose another folder in which to save the configuration.

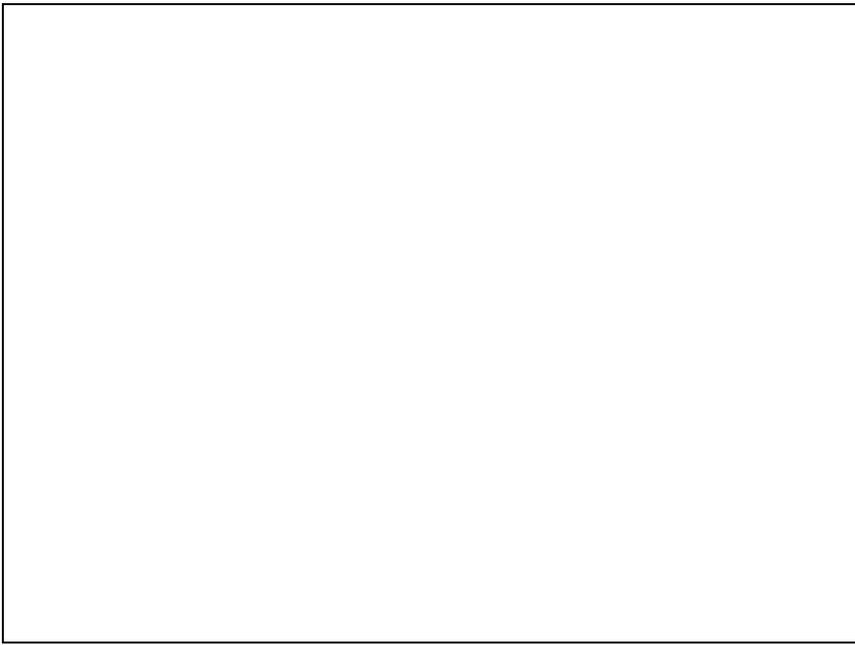
8. Start the connection to the CIU to make sure it works: on the **View** menu, click **Monitor**. After a moment, the text "CIU: ONLINE" will appear in the RoviSys Turbo status bar. You must see **MyRing** and **MyNode** having non-zero values, except for Plant Loop, in which case **MyRing** will be zero.

To configure RoviSys Turbo's communications with your CIU:

- In RoviSys Turbo, right-click **CIU** in the left-hand pane, then click **Properties**. The **Device Properties** window appears.

For most people using DBDOC, we recommend that you:

- Change the **Import exception report request rate** to **60000 ms**, unless you are using OPC tags.
- Select the **Enable Turbo Polling** check box.
- Select the **Non-keyed output polling** check box.
- Select the **Enhanced Analog Precision** check box.



Normally, programs like Composer and DBDOC must get data from your system one block at a time, but "turbo mode" allows RoviSys Turbo to fetch multiple blocks at once. Using turbo mode, you can get data more quickly, while reducing the load on your system.

If you are having problems getting data with turbo mode, please contact us for help. We will work with RoviSys to make sure that you can get data reliably while benefiting from turbo mode's higher speeds and reduced load on your system. Meanwhile, to make sure that you are always able get data, you can temporarily disable turbo mode.

To disable turbo mode:

- Clear the **Enable Turbo Polling** box.

To enable turbo mode:

- Select the **Enable Turbo Polling** box.

Depending on how much data is being asked for, RoviSys Turbo can make multiple requests for data at once by using different "keyed" requests. However, you can usually get data more quickly if RoviSys Turbo is making only one request at a time. So, for most situations, we recommend setting RoviSys Turbo to make only one request at a time by selecting the **Non-keyed output polling** box.

5.4.3 Methods of connecting CIUMon to RoviSys Turbo

There are two ways to connect CIUMon and RoviSys Turbo:

- **RoviSys Turbo virtual ports** are easy to set up and provide high data rates. If RoviSys Turbo and CIUMon are on different computers, they can still be connected with RoviSys Turbo virtual ports

through the network. RoviSys Turbo virtual ports require RoviSys Turbo 7.2 or newer and CIUMon 10.4 or newer.

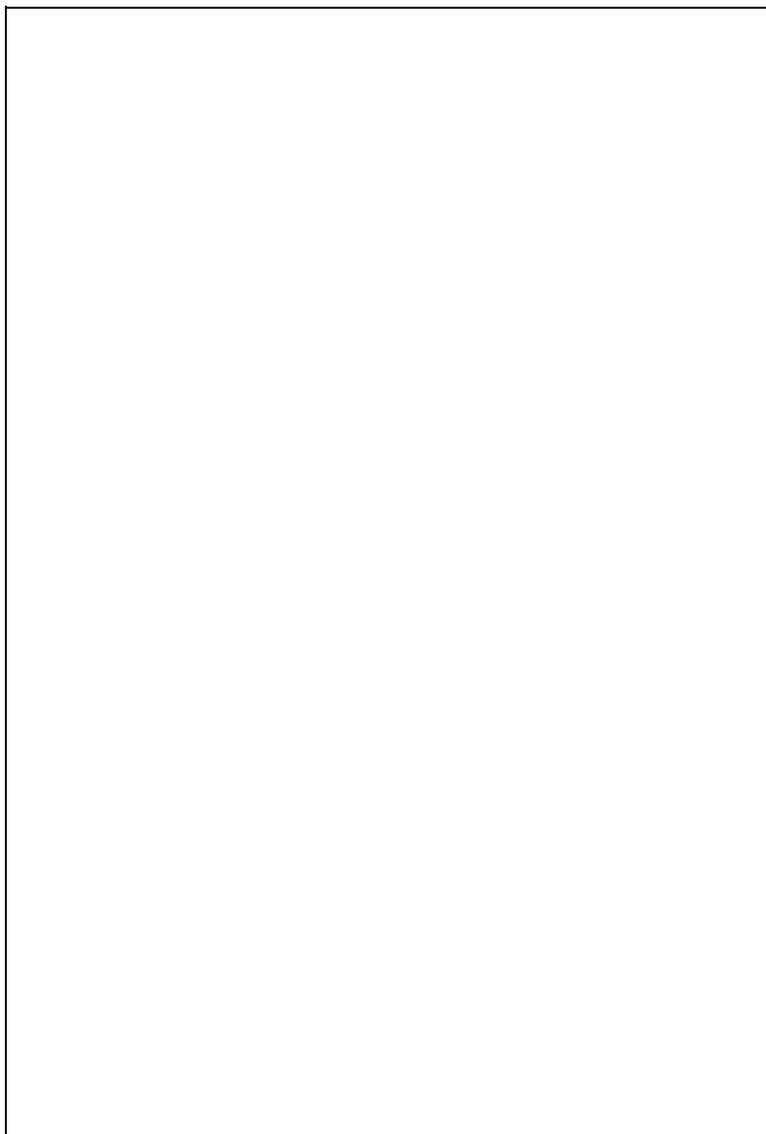
- **Serial ports** are generally harder to set up and slower. Some system managers may choose to use serial ports if RoviSys Turbo and CIUMon are on different computers, in order to keep OPC data off the network. Serial ports can also be used with older versions of RoviSys Turbo and CIUMon.

5.4.4 Connecting to RoviSys Turbo with virtual ports

To connect CIUMon to RoviSys Turbo by using RoviSys Turbo virtual ports:

1. Configure the virtual ports in RoviSys Turbo
2. Set up CIUMon to connect to the RoviSys Turbo virtual ports

To set up RoviSys Turbo:



1. In RoviSys Turbo, stop the connection to the CIU: on the **View** menu, clear the **Monitor** check mark.
2. Expand **CIU** in the left-hand pane. Right-click **DBDOC**, then click **Properties**.
3. In the **Addressing** box, select **Virtual port**.
4. In the **Port** list, select a different port from the one the CIU is using.
5. Select **Read only** and **Allow R4**. Ensure that **Prefetch** is *not* selected.
6. Click **OK**.
7. On the **File** menu, click **Save**. Then, start the connection to the CIU: on the **View** menu, click **Monitor**.

To set up CIUMon:



1. On CIUMon's **Options** menu, click **Settings**. Choose a UDP port, and select your CIU type.
2. Click the **Communication Modes** tab. Select **Connect to RoviSys Turbo** and **via virtual port connection**.

3. Type the IP address or hostname of the computer running RoviSys Turbo in the **RoviSys computer name or IP address** box.
4. If you changed the name of the **CIU** or **DBDOC** blocks in RoviSys Turbo's left hand pane, change the **Device / MUXCIU block name** box to match.
5. If you know that any of the other settings need to be changed for your setup, change them now. Usually the defaults are fine.
6. Click **OK** to close the Options window. Then, make sure you can get data from your CIU.

5.4.5 Connecting to RoviSys Turbo with serial ports

To connect CIUMon to RoviSys Turbo by using serial ports:

1. Set up a pair of serial ports on your computer
2. Configure RoviSys Turbo to provide data to one port
3. Configure CIUMon to connect to the other port

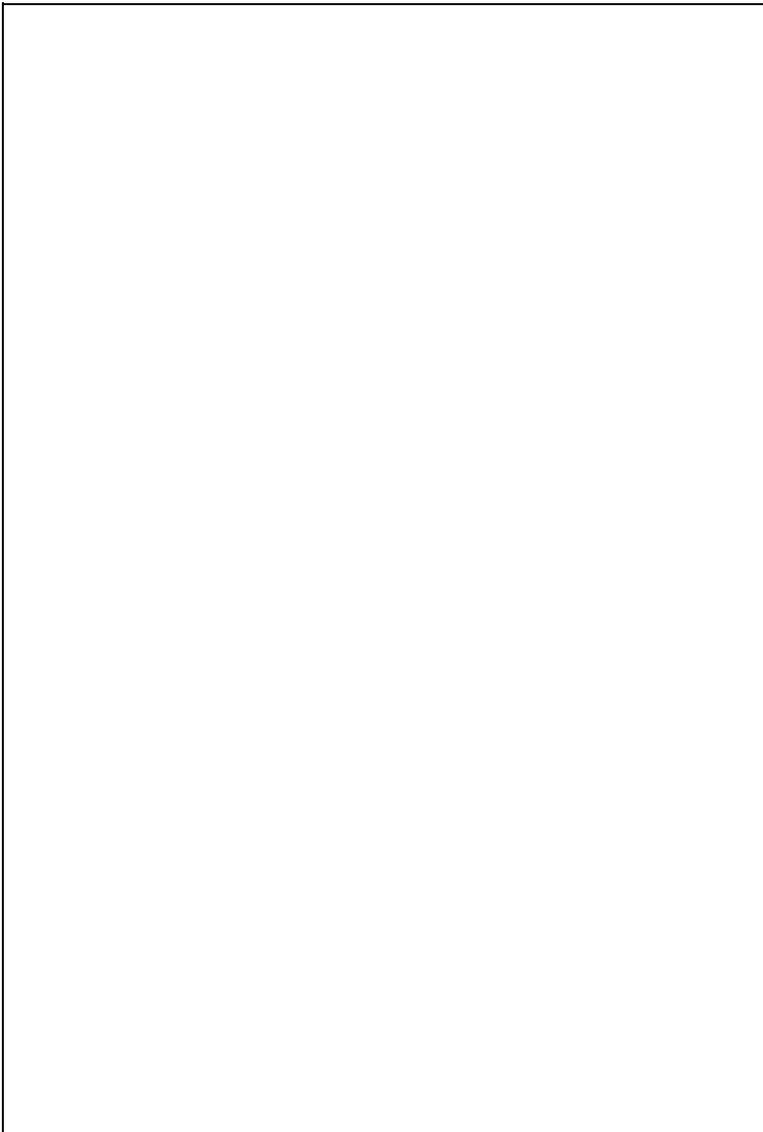
To set up the serial ports:

1. If RoviSys Turbo is running, exit RoviSys Turbo: on the **File** menu, click **Exit**.
2. Connect two serial ports with a null-modem cable for CIUMon and RoviSys Turbo to communicate with each other through. RoviSys Turbo will connect to one of these ports, and CIUMon will connect to the other.
3. Write down which ports you set up and which program you plan to connect to each of them.

Notes:

- ◆ Very old versions of RoviSys Turbo cannot connect to serial ports above COM9, but you can use low port numbers for RoviSys Turbo and higher ones for CIUMon in that case.
- ◆ If RoviSys Turbo and CIUMon are on the same computer, and it does not have a 64-bit operating system, you can use virtual serial ports.
- ◆ If RoviSys Turbo and CIUMon are on different computers or on a computer with a 64-bit operating system, and there are not enough serial ports, you can add more ports with an internal interface card or a usb-to-serial converter. Alternately, you can use a third-party virtual serial port program.

To set up RoviSys Turbo:



1. Start RoviSys Turbo.
2. Expand **CIU** in the left-hand pane. Right-click **DBDOC**, then click **Properties**.
3. In the **Port** list, click the port that you set up for RoviSys Turbo. Select **Read only** and **Allow R4**. Click **OK**.
4. On the **Edit** menu, click **COM Ports**. In the **Port** list, click the port that RoviSys Turbo was set to use in the previous step. Set the baud rate to **57600**. Click **OK**. **115200** is overly vulnerable to ambient interference and should not be used.
5. On the **File** menu, click **Save**. Then, start the connection to the CIU: on the **View** menu, click **Monitor**.

To set up CIUMon:



1. On CIUMon's **Options** menu, click **CIUMon Settings**. Choose a UDP port, and select your CIU type.
2. Click the **Communication Modes** tab. Select **Connect to RoviSys Turbo** and **via serial port**.
(**Connect directly through serial port** can be used instead, but there is some loss of performance using that option.)
3. Click the **Comm and Flow Control Settings** tab. Select the serial port you set up earlier for CIUMon in the **Serial Port** list. In the **Baud Rate** list, click the baud rate of the serial port connected to the CIU.

Note: If you are using virtual serial ports, this setting doesn't matter.

4. If you know that any of the other settings need to be changed for your setup, change them now.
Usually the defaults are fine.
5. Click **OK** to close the Options window. Then, make sure you can get data from your CIU.

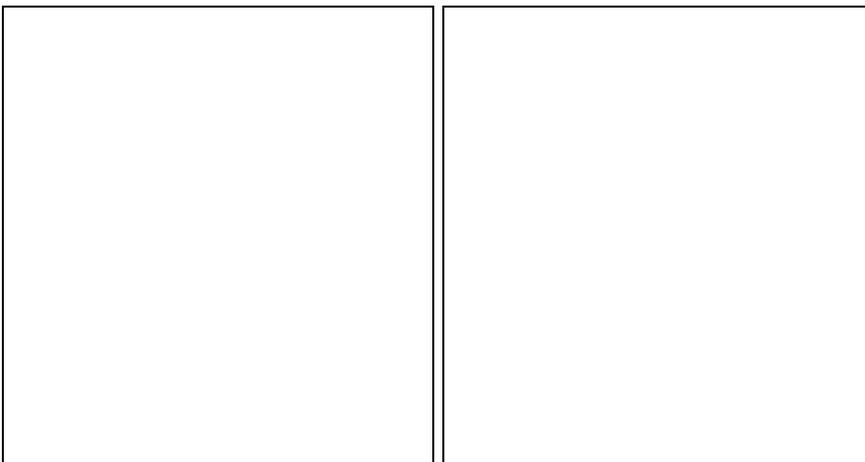
5.4.6 Adding serial ports to your computer



RoviSys Turbo may connect to CIUMon or your EWS software through a serial line. To do so, RoviSys Turbo and the other program must each have a COM port, and these ports must be connected. In many cases, if RoviSys Turbo and the other program are running on the same computer, they can connect through virtual serial ports rather than real serial ports.

- Instead of using serial ports, CIUMon can connect to RoviSys Turbo using RoviSys Turbo virtual ports.
- On 32-bit platforms, you can connect your EWS software to RoviSys Turbo using virtual serial ports included with DBDOC.
- The included virtual serial ports do not work on 64-bit Windows platforms (including Windows 7). To add serial ports to one of these platforms, use a usb-to-serial converter, or better, use internal serial ports on the motherboard or install a PCI serial card. Alternately, you can use a third-party virtual serial port program.

To set up virtual serial ports:



1. On the CIUMon **Options** menu, click **Virtual Serial Port Setup**. The Virtual Port Setup program will open.

2. On the Virtual Port Setup **Options** menu, click **VSP Configuration**.
3. In the **Virtual Serial Port Creation** list, click any port. Hold down CTRL and click a second port. Click **Create Now**. The two ports that were selected will disappear from the **Virtual Serial Port Creation** list and appear in the **Virtual Serial Port Deletion** list.
4. Click the **Virtual-To-Virtual Connections** tab.
5. In the **VSPs Available** list, click the first port, then click the second port. Click **Add To List**. The two ports will disappear from the **VSPs Available** list, and the new connection will appear in the **Virtual-To-Virtual Connections** list.

Note:

- The **Virtual Serial Port Setup** option is disabled on 64-bit Windows platforms (including Windows 7). Also, they do not work on Windows 8. To add serial ports to 64-bit Windows, use a usb-to-serial converter. Alternately, you can use a third-party virtual serial port program.
- If you have not run Virtual Port Setup before, it will be installed when you try to run it. You will then be prompted to reboot the computer.
- The first time you set up and try to use virtual serial ports after installing them and rebooting the computer, you may see an **Error: COMx.Read (), code 87** message in the Virtual Port Setup window. If this occurs, reboot your computer again -- the virtual serial ports should work fine after this second reboot.
- If you get a **Could not find OEM file** error after your computer reboots, copy the file "vsp.oem" from the folder where CIUMon is installed to the system32 subfolder of your computer's Windows folder.
- If you later install more physical serial ports, you should delete all of the virtual serial ports and connections first, then re-create them after the new physical ports are installed and working.

To remove virtual serial ports:

1. In **Windows Explorer**, browse to the `Programs` folder in your installation directory (this is usually in either `C:\Program Files\GMCL\DBDOC` or `C:\DBDOC\CIUMON`).
2. Double-click `VSPCleanup.exe` and follow the instructions.
3. If the computer was not rebooted in the previous step, reboot it now.

To set up internal serial ports:

1. If you are installing a new interface card to add serial ports, you will have to power down the computer and open it up. Identify an unused slot of the type required by the card and remove the corresponding cover from the back of the case. Depending on the number of ports supported by the card, you may also have to connect a *serial port header* to the card and install it in another unused opening in the back of the case.
2. Identify the serial ports you've installed:
 1. Right-click the **My Computer** icon on the desktop, and then click **Properties**.
 2. Click the **Hardware** tab, then click the **Device Manager** button.
 3. Expand the **Ports (COM & LPT)** entry. The serial ports should each be listed as **Communications Port (COMn)** or **mumble Serial Port (COMn)**, where *n* is the port number and *mumble* is some descriptive text.
3. Connect two ports with a null modem cable, to provide a connection between them.

To set up a usb-to-serial converter:

1. Plug the converter into an available USB port on your computer.

After a short delay, Windows will recognize and install the converter. When it has finished installing the converter, it will display a message in the system tray.

2. Check that it is installed correctly:
 1. Right-click the **My Computer** icon on the desktop, and then click **Properties**.
 2. Click the **Hardware** tab, then click the **Device Manager** button.
 3. Expand the **Ports (COM & LPT)** entry. The new serial ports should each be listed as **USB Serial Port (COM n)**, where n is the port number.
3. Connect two ports with a null modem cable, to provide a connection between them.

Note: GMCL has USB-to-serial converters available for sale. Please contact us for details.

A note about USB-to-serial converters and Windows 7:

On Windows 7 and Windows 2008 Server, USB-to-serial converters do not work as well as internal serial ports. They are slower and have higher error rates. In some cases, they may not work at all. You should use internal serial ports if possible. Alternately, you can use a third-party virtual serial port program.

If you do use USB-to-serial converters on Windows 7 and 2008 Server, do not use speeds above 115200 bps.

If Windows 7 or Windows 2008 Server does not automatically install your USB-to-serial converter, you will need to download drivers from the converter manufacturer's website. The drivers for the USB-to-serial converters sold by GMCL are available at http://www.vscom.de/616_download.htm.

Third-party virtual port alternatives:

As an alternative to RoviSys Turbo virtual ports, the virtual serial port software included with DBDOC, or physical serial ports, you can use a third-party virtual serial port program to connect CIUMon or your EWS software to RoviSys Turbo.

These software packages can create virtual serial ports for CIUMon or your EWS software to use for connecting to RoviSys Turbo:

- The com0com null-model emulator is an open source virtual serial port driver that supports both 32-bit and 64-bit versions of Windows. Note that to use this program with 64-bit Windows, you will need a signed version of the com0com driver. Contact us for information about obtaining a signed version of the driver.
- The Eltima Virtual Serial Port Driver is a commercial software package that can create virtual serial ports. It supports both 32-bit and 64-bit versions of Windows.

5.4.7 Connecting other programs to RoviSys Turbo

If you are using RoviSys Turbo to share a serial connection to a CIU between CIUMon and EWS software (such as Composer or WinCAD), you can avoid changing the EWS software settings with the following strategy:

1. Move the CIU connection to a different serial port, and connect RoviSys Turbo to that serial port.
2. When you set up RoviSys Turbo to accept connections from the EWS software, connect a null cable between the serial port that the CIU was connected to and a third serial port.
3. Then, have RoviSys Turbo use the third serial port to communicate with the EWS software. The EWS software will connect to RoviSys Turbo through the null cable, using the same port as before.

A side benefit of this approach is that you can reconnect the CIU directly to the EWS simply by moving the cable back. If the computer does not have the extra serial ports you need for this approach, you can add serial ports with an internal interface card or a usb-to-serial converter.

To connect other programs to RoviSys Turbo:

1. If RoviSys Turbo is running, exit RoviSys Turbo: on the **File** menu, click **Exit**.
2. Set up two connected (real or virtual) serial ports for RoviSys Turbo and the other program to communicate with each other through. Write down the two port numbers. RoviSys Turbo will connect to one of these ports, and the other program will connect to the other one.

Note: If you moved the CIU connection so you wouldn't have to change settings in the EWS software, then use a connection between the serial port it was originally using and another real serial port.

3. Start RoviSys Turbo and open the saved configuration.
4. Right-click **CIU** in the left-hand pane, then click **New Block**.
5. Type a name for the block in the **Name** box (e.g. WinCAD).
6. In the **Block Type** list, click **MUXCIU**.
7. In the **Port** list, click the port that you set up for RoviSys Turbo. Click **OK**.
8. On the **Edit** menu, click **Ports**. In the **Port** list, click the port that you set up for RoviSys Turbo again. Set a baud rate that your EWS software supports (typically **19200**). Click **OK**.
9. Start the connection to the CIU: on the **View** menu, click **Monitor**.
10. Set up the other program to connect to the port you set up for it.

Note: WinCAD and Composer have been tested to work when connected to a RoviSys Turbo MUXCIU block as described here. Conversely, while S+ Engineering tools may have basic functionality, some functions such as firmware and program downloads will not work, and so RoviSys does not officially support connecting to S+ Engineering tools in this way. Therefore, if you have the option it would be best practice to have a separate, dedicated CIU to use with S+ Engineering tools instead of attempting to multiplex it with DBDOC using RoviSys Turbo.

5.5 Connecting to Harmony OPC Server

DBDOC can connect to HAOPC 6.0 or later. This can allow you to get data from an IET800 ethernet CIU. To do this:

1. Configure the Harmony OPC Server by adding three DBDOC tags. To do this, you will:
 - ◆ Generate Harmony OPC backup configuration files
 - ◆ Modify those files by adding three DBDOC tags
 - ◆ Upload your changes to the configuration
2. Set up CIUMon to connect to the Harmony OPC Server.

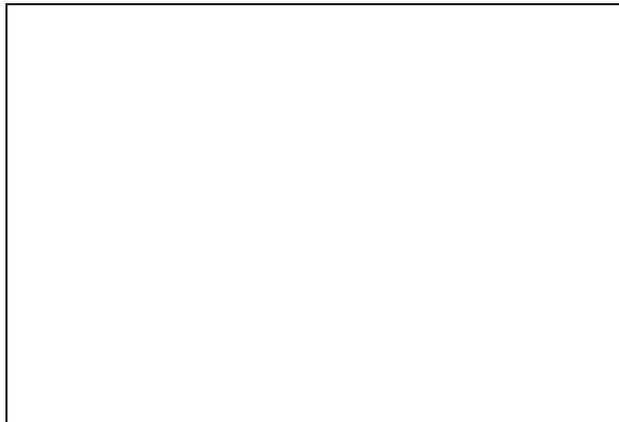
Generate backup configuration files:

You will need to edit your Harmony OPC backup configuration files in order to add DBDOC tags. If you do not already have these configuration files, use these steps to generate them.

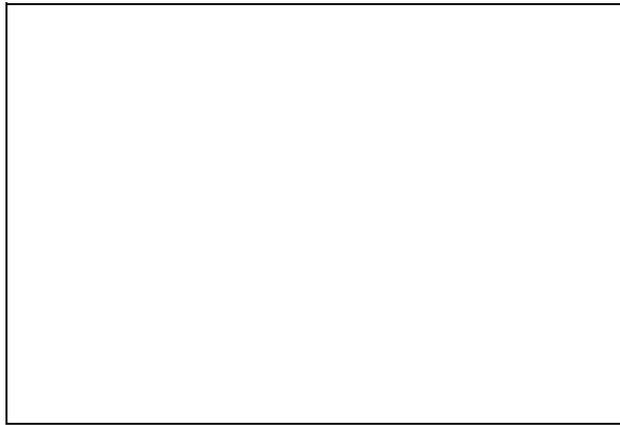
1. Start the **HOPC Online Utility**: On the **Start** menu, under **Programs**, click **ABB Symphony Plus**, then click **Harmony OPC**, and then click **HOPC Online Utility**.
2. In the **HOPC Online Utility** window, on the **Configuration** menu, click **Backup Configuration**. Choose a folder where you would like to save the backup configuration files.
3. When the backup completes, click **OK**. Then, close the **HOPC Online Utility** program.

Add DBDOC tags to your configuration files:

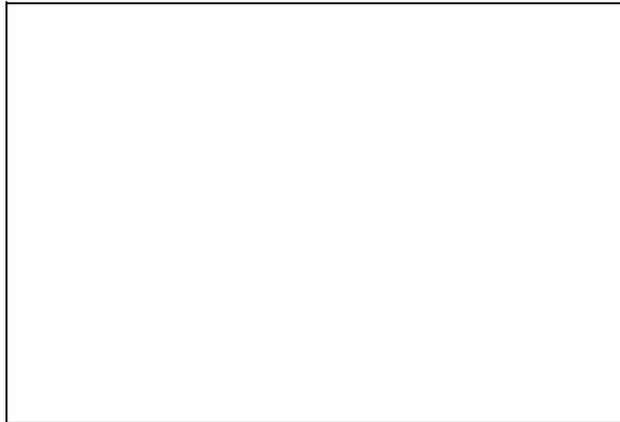
1. Start the **HOPC Offline Utility**: On the **Start** menu, under **Programs**, click **ABB Symphony Plus**, then click **Harmony OPC**, and then click **HOPC Offline Utility**. Then, on the **File** menu, click **Load** and choose to load the backup configuration files that you previously saved.
2. Add tags to your configuration: On the **Edit** menu, click **Add**. You will need to add three tags:
 - ◆ A tag with the **Tag Type** "BLOCKOUTPUT" and **Tag Name** "DBDOC_OUTPUT", for loop 0, node 0, module 0, block 1.



- ◆ A tag with the **Tag Type** "BLOCKTUNE" and **Tag Name** "DBDOC_TUNE", for loop 0, node 0, module 0, block 1.



- ◆ A tag with the **Tag Type** "BLOCKMODSTATUS" and **Tag Name** "DBDOC_MODSTAT", for loop 0, node 0, module 0, block 0.



3. After adding the three tags, click the **Verify** button to make sure the changes are valid.
4. Click the **Save** button to save these tags. Then, click **Exit**.

Upload your changes to the configuration:

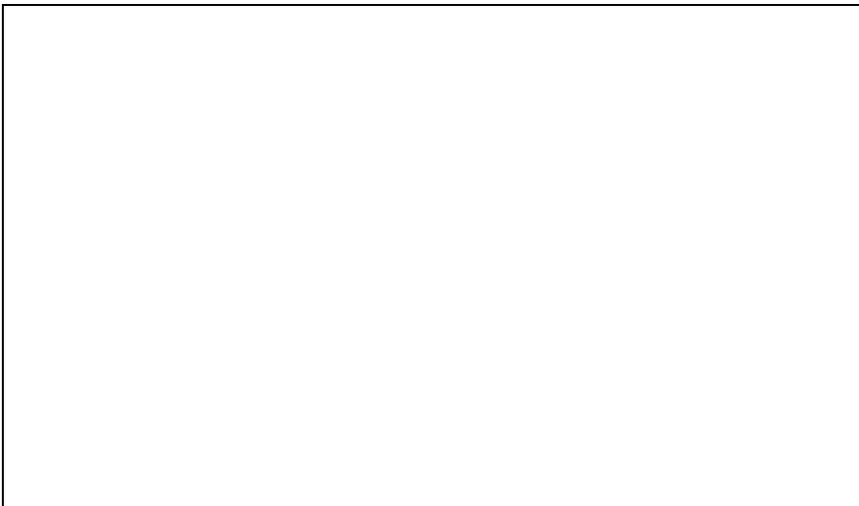
1. Start the **HOPC Online Utility**: On the **Start** menu, under **Programs**, click **ABB Symphony Plus**, then click **Harmony OPC**, and then click **HOPC Online Utility**.
2. On the **Tools** menu, click **Apply Tag Configuration**. Load the backup configuration file that you modified by adding the three DBDOC tags.
3. Click the **Verify** button. The HOPC Online Utility should show that three changes are detected. Then, click **Apply** to load and merge your changes onto the HOPC server.
4. Close the **HOPC Online Utility** program.

Set up CIUMon to connect to Harmony OPC Server:

1. In CIUMon, on the **Options** menu, click **Settings**.
2. Click the **Communication Modes** tab. Select **Connect to Harmony OPC Server**.
3. Select the appropriate server instance for CIUMon to connect to.
4. Type the IP address or hostname of the computer running Harmony OPC Server in the **Harmony OPC computer name** box.
5. Click **OK** to close the Options window. Then, make sure you can get data from your CIU.



5.6 Managing CIUMon configurations



CIUMon can store different configurations. This is especially useful if the computer running CIUMon is connected to multiple CIUs, since it allows a separate configuration to be stored for each CIU. These configurations can be used to run multiple copies of CIUMon connecting to different CIUs, or to change which CIU a single copy of CIUMon is connected to.

Current versions of CIUMon store configurations in the registry. Older versions of CIUMon stored settings in a file in the Windows directory. The first time a newer version of CIUMon is run, it will convert old settings to a configuration called "Configuration 1", which will be loaded automatically.

On startup, CIUMon will attempt to load the configuration that was opened last. If that configuration is already in use, it will open the CIUMon Configurations window so that you can select another configuration.

To keep your configurations organized, use CIUMon's configuration management tool to identify unused configurations that can be hidden or deleted.

5.6.1 Loading, saving, and deleting configurations

To save a configuration:

1. On the **File** menu, click **Save Configuration**.
2. In the "Save Current Configuration As" box, type a name for the configuration.
3. Click **OK**.

To load a configuration:

1. On the **File** menu, click **Load Configuration**.
2. In the **Select the configuration that you wish to load** box, click the configuration to load. Or, click **(New DCS Configuration)** to load CIUMon's default settings for connecting to a CIU. Use **(New Relay Configuration)** only if you are setting up CIUMon in relay mode or collector mode.
3. To see what settings are in the configuration before loading it, click **Preview Settings**.
4. Click **OK**.

To delete a configuration:

1. On the **File** menu, click **Delete Configuration**.
2. In the **Select the configuration that you wish to delete** box, click the configuration you want to delete. You can select several configurations to delete at the same time.
3. To see what settings are in the configuration before deleting it, click **Preview Settings**.
4. Click **Delete**.

Note: If you delete the currently loaded configuration, you will be prompted to confirm first. After closing the dialog, you will be prompted to choose another configuration to load.

Note: You can also use CIUMon's configuration management tool to delete configurations.

5.6.2 Keeping your configurations organized

Use CIUMon's configuration management tool to keep your configurations organized, by identifying unused configurations and hiding or deleting them.

To use this tool: On the **File** menu, click **Manage Configurations**.



To identify unused configurations:

Each UDP port and serial COM port on your computer can only connect to one program at a time. So, if you have two CIUMon configurations that are both using the same UDP port or COM port, you cannot use them both at once. Usually, in this situation, one of the configurations will be the one you are currently using, while the other is unused: for example, the unused configuration might be an old configuration that is no longer relevant, or it might simply have been created by accident.

To identify configurations that share a UDP or COM port: In the **Sort configurations by** box, select one of the following.

- **UDP Ports.** This will show you which configurations are using each port to receive requests for data. You can easily see if there are multiple configurations that all use the same port. Since every CIUMon configuration must use a UDP port to receive requests, all of your configurations will appear somewhere in this list.
- **Serial COM ports.** This will show you which configurations use each COM port to communicate with the CIU. Some configurations may not use COM ports ? for example, you might have configurations that connect to a TRAX emulator, to a Harmony OPC server, or to RoviSys Turbo by using a virtual connection. These configurations will not appear in the list when you choose to sort by **Serial COM ports**.
- **Collector Ports.** This will show you which collector receiver configurations are using each port to listen for data from the upstream CIUMon. Only relay configurations that are set up as collector receivers will appear in this list.
- **Serial Port Module Loop/PCU.** If you have configurations configured to connect to a serial port module (SPM01 or SPM02), this will show you which configurations are set up for each loop and PCU.
- **Configurations with/without conflicts.** This will show you all of the configurations that share a UDP port, COM port, collector port, or serial port module loop/PCU with some other configuration.

Click **Preview Settings** or double-click on any configuration to see exactly what is being used and by which configurations.

- **Configurations with/without services.** Configurations that are actively in use may have services set up to run them automatically. So, the configurations without services displayed in this list are more likely to be unused.

If you have two or more configurations that share a UDP or COM port, only one of those configurations will be able to connect at a time. So, the others might be unused. Use these techniques to determine which configurations are unused.

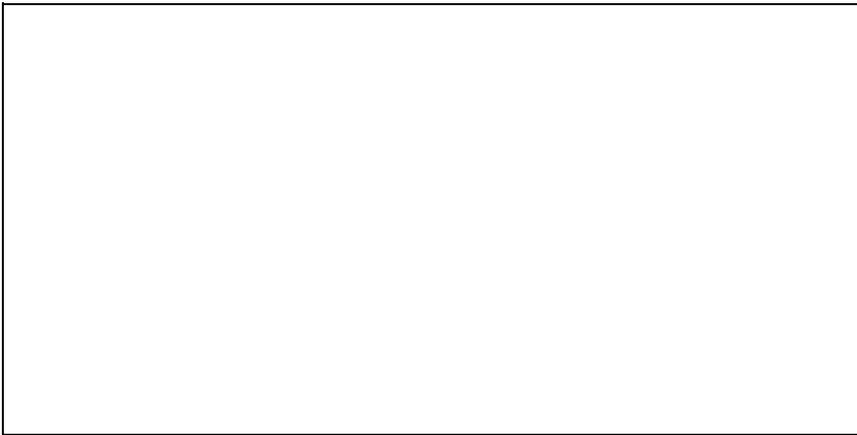
- To view the settings for any configuration: Click **Preview Settings**; or double-click on the configuration. The preview window displays all of the options for the configuration, as well as a list of other configurations that share a COM or UDP port.
- Use the "Service" column in the list to see which configurations have services set up to run them automatically. Configurations that have services associated with them are also displayed in green. Configurations without services might be more likely to be unused.

To hide or delete unused configurations:

- **To hide unused configurations.** Click on the configuration you want to hide, and click **Make Inactive**. You can hold down shift or ctrl to select multiple configurations and hide them all at the same time. CIUMon will not load inactive (hidden) configurations and you will not see them in the Save Configuration or Load Configuration windows, but you can restore them if you would like to use them again.
- **To view inactive configurations.** Select the **Show Inactive Configurations** box. You can identify which configurations are inactive by using the **Inactive** column in the list. Inactive configurations will also be displayed in grey. Alternatively, in the **Sort configurations by** box, select **Active/Inactive Configurations** to get a list of all inactive configurations.
- **To restore inactive configurations.** Select the configuration you want to restore and click **Make Active**.
- **To permanently delete unused configurations.** Click on the configuration you want to delete, and click **Delete**. You can hold down shift or ctrl to select multiple configurations and delete them all at the same time.

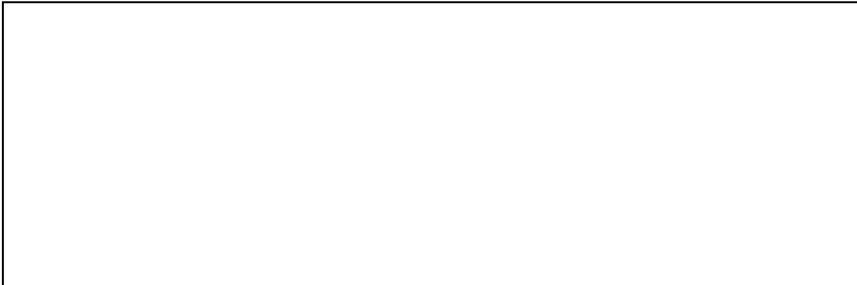
5.7 Getting live data across a network

DBDOC can provide read-only live data across most network setups.



5.7.1 Getting live data through a firewall

By default, CIUMon listens on UDP port 5099. In order to connect to CIUMon from across a firewall, this port must be unblocked to allow UDP communication.



To avoid having user connections that must be allowed through the firewall, CIUMon can be run in relay mode or collector mode on a computer outside the firewall. This CIUMon connects to the main CIUMon through the firewall, and live data users connect to the relay or collector mode CIUMon.



If your firewall rules require the upstream CIUMon to initiate all communications, you may use collector mode instead of relay mode. As in relay mode, CIUMon will accept requests for data to pass along to an upstream CIUMon. In collector mode, however, CIUMon does not initiate any communication with the upstream CIUMon; instead, it waits for that CIUMon to ask for requests.

By default, in relay mode the firewall would need to be configured to allow the relay CIUMon to communicate with UDP port 5099 of the upstream CIUMon. In collector mode, the firewall would need to be configured to allow the upstream CIUMon to communicate with UDP port 6099 of the relay CIUMon. Note that the direction of traffic, and which computer is the one listening, changes depending on the mode. Also, the default port is different in collector mode -- this is because the relay CIUMon also listens for Hyperview requests on port 5099 by default, so a different port needs to be chosen for communication with the upstream CIUMon.

However, you can easily configure CIUMon to use other port numbers instead of these defaults.

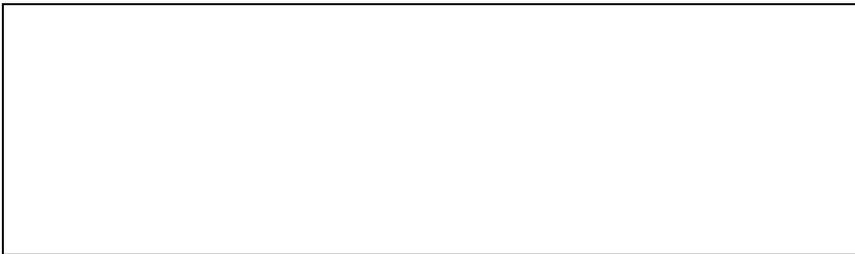
5.7.2 Getting live data across an intermediate LAN with relay mode

Some networks have an intermediate LAN so that computers on the business LAN can access the historian computer, but not the DCS LAN. Relay mode or collector mode can be used in these setups to allow users on the business LAN to access live data in DBDOC.



5.7.3 Getting live data directly from a business LAN

In some setups, it may be acceptable to run CIUMon on a server on your business LAN and connect it to a read-only RoviSys Turbo port on the DCS LAN through a serial line. RoviSys Turbo connects to the CIU and sends data back to the CIUMon on the business LAN. Because this setup uses a RoviSys Turbo read-only port, no changes to the DCS LAN or the plant can be made from the business LAN.



5.7.4 Running CIUMon in relay mode

In relay mode, CIUMon will send requests for data along to an upstream CIUMon, and pass back the responses. If there is a firewall between your users and the upstream CIUMon, you need only configure the firewall to allow the relay mode CIUMon to pass requests through it on a single UDP port, and this will allow all of your users to get data by communicating with the relay mode CIUMon. If your firewall rules require the upstream CIUMon to initiate all communications, you may use collector mode instead.

To run CIUMon in relay mode:

1. Load a new relay configuration: on the **File** menu, click **Load Configuration**. Then, select "New Relay Configuration" and click **OK**.
2. On the **Options** menu, click **Settings**.
3. Click **Relay mode**.
4. Type the IP address of the CIUMon that you want to connect to in the **Computer name or IP address of upstream CIUMon** box.

If you have changed the UDP port for the CIUMon you want to connect to, specify the port in the **UDP port of upstream CIUMon** box.

Once you have configured CIUMon to run in relay mode, save your configuration and make sure you can get data from your CIU.

Note: In the display of the upstream CIUMon that is connected to the CIU, requests that come from the relay mode CIUMon will have the IP address of the relay mode CIUMon. However, the logon name will be the user who is using the data.

5.7.5 Running CIUMon in collector mode

In collector mode, CIUMon will wait for an upstream CIUMon to ask if there are any requests for data. Then, it will reply to the upstream CIUMon to send along any requests. If there is a firewall between your users and the upstream CIUMon, you need only configure the firewall to allow the upstream mode CIUMon to communicate through it on a single UDP port, and this will allow all of your users to get data by communicating with the collector receiver CIUMon. The collector receiver CIUMon does not initiate any communication with the upstream CIUMon; it waits for that CIUMon to ask for requests. If your firewall rules do not require the upstream CIUMon to initiate communications, you may use relay mode instead.

To set up the upstream CIUMon:

1. Start CIUMon on the upstream computer that is connected to your CIU.
2. If CIUMon is connected, disconnect it: on the **File** menu, click **Disconnect**.
3. On the **Options** menu, click **Settings**. Click the **Collector Source** tab. Select **Act as collector source**, and type the IP address of the computer where you want to set up CIUMon in collector mode. You may also select which UDP port will be used to communicate with the collector receiver CIUMon.
4. Click **OK**, and save your configuration. Then, connect CIUMon: on the **File** menu, click **Connect**.

To set up the collector mode CIUMon:

1. On the computer where you want to run CIUMon in collector receiver mode, Start CIUMon and load a new relay configuration: on the **File** menu, click **Load Configuration**. Then, select **(New Relay Configuration)** and click **OK**.
2. On the **Options** menu, click **Settings**.
3. Click **Collector Receiver**. Make sure that the port in the **Collector receiver UDP port** box matches the one on the **Collector Source** tab of the upstream CIUMon.
4. Click **OK** and save your configuration. Then, make sure you can get data from your CIU. Note that it may take about a minute before you can start getting data, as the collector CIUMon waits for the upstream CIUMon to establish a connection with it.

5.7.6 Is it safe to let everyone on the network access live data?

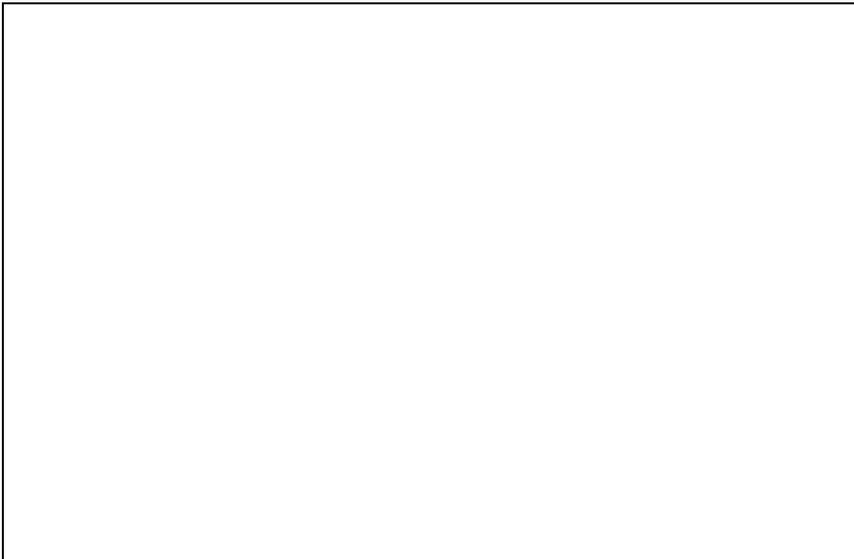
DBDOC's live data features are read-only. Remote users cannot change anything using DBDOC.

CIUMon does have the ability to restart your CIU, which may be useful in troubleshooting. This ability cannot be accessed remotely. It can only be used from the CIUMon computer that has a physical connection to the CIU.

5.8 Running CIUMon automatically (as a service)

Running CIUMon as a service ensures that CIUMon will run if the computer is restarted for any reason. It also restarts CIUMon after some errors in some circumstances, and allows CIUMon to run without any user logged in.

5.8.1 Setting up a CIUMon service



1. If CIUMon connects to your CIU through RoviSys Turbo, set up RoviSys Turbo to run automatically
2. On the **Options** menu, click **Services**.
3. Click **Add A Service**.
4. In the **Configuration To Load** list, click the configuration that the service will use.
5. If you like, you can make changes to the name for the service in the **Name of Service** box, or create a description in the **Description of Service** box.
6. Make sure that the **Allow service to interact with desktop** check box is cleared. Clearing this box causes CIUMon to run silently.
7. Click **OK**.
8. Click **Close**.
9. Close CIUMon and reboot the computer.

Note: You can also create a service using CIUMonController: in the CIUMonController window, click **Add a Service**.

5.8.2 Changing a CIUMon service

There are two types of settings that affect a CIUMon service: the settings in the configuration it is using, and the settings for the service itself. The different types of settings are changed in different ways.

The service settings can be changed through the Services window.

The logging and statistics options in the configuration can be changed using CIUMon Service Controller.

The other options in the configuration must be changed through CIUMon options.

To use CIUMon Service Controller,

- On the **Start** menu, under **Programs**, click **GMCL DBDOC** and then click **CIUMon Service Controller**.
- Or, in the folder where DBDOC was installed, open the **Programs** folder and run **CIUMonController.exe**.

To change service-only options:

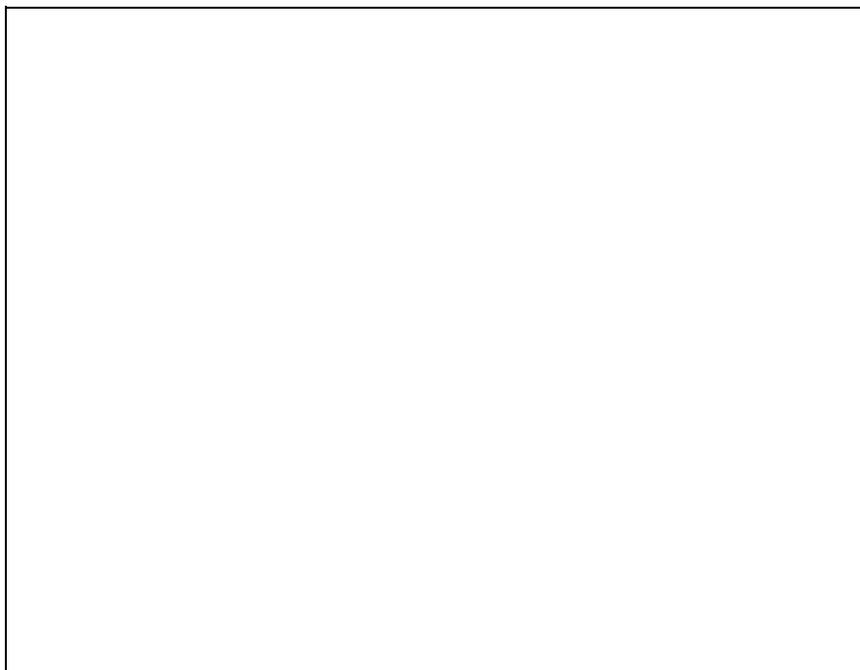


Use the **Services** window to start or stop a CIUMon service, prevent a service from starting when you reboot, make a service run in a window, or allow non-administrators to start and stop the service by renaming a file.

1. Open CIUMon.
2. On the **Options** menu, click **Services**.
3. In the **Current CIUMon Services** list, click the CIUMon service to be modified.
4. Change the service's options.
5. Click **Apply Changes**.

Note: You can also modify a service using CIUMonController: in the CIUMonController window, click the service you want to change and then click **Modify Service**.

To log CIUMon messages for troubleshooting:



1. Start CIUMon Service Controller.
2. In the list of services, click the service that you want to create logs for.
3. Click **Modify Configuration Options**. Choose the options you want. Then, click **OK**.

Note: Some options may not be available in CIUMon Service Controller. To access these options, stop the service and run CIUMon manually. Then, load the appropriate configuration. On the **Options** menu, click **Settings** and then select the **Logging Options** tab.

To view statistics about who is using live data:

1. Start CIUMon Service Controller.
2. In the list of services, click the service that you want to view statistics for.
3. In the **Statistics Options** box, click **View Communication Statistics** or **View Usage Statistics**.

To change other options:

1. Start CIUMon Service Controller.
2. Select the service you are modifying and click **Launch in Manual Mode**. If the service was running, it will stop. Then, a CIUMon window will open.
3. In CIUMon, change the options as required.
4. In CIUMon, save the configuration.
5. Quit CIUMon.
6. In CIUMon Service Controller, click **Start Service** to get your service running again.

Note: If you are using an account without administrator privileges, you may still launch CIUMon Service Controller and make changes to services. However, when CIUMon Service Controller starts, you will be

asked for the user name and password of an administrator account.

Note: Logging options can be changed directly in CIUMon Service Controller. Service options can be changed without stopping the service.

5.8.3 Error recovery when running as a service

If CIUMon is running as a service and it stops without exiting cleanly (ie it crashes), it will be restarted. This should prevent live data from being unavailable, and avoids requiring the administrator.

If CIUMon crashes again within 30 seconds, it will not be restarted a second time, as this generally indicates a problem that requires attention. If this happens, please contact GMCL for further assistance.

5.8.4 How to tell if CIUMon is running as a service

If you are running CIUMon as a service, the CIUMon window will not normally be displayed on your computer. However, there are still several techniques that you can use to make sure that CIUMon is running properly.

- The best way to make sure that a CIUMon service is running properly is by checking if you can access live data in Hyperview. See Live Data Configuration in Hyperview and Displaying Slow Mode live data in Hyperview.
- You can also use CIUMon Service Controller to check if your CIUMon service is running. Start CIUMon Service Controller and check the **Current service state** in the list of services.
- If you have a single CIUMon service, you can use Windows Task Manager to check if it is running. Press CTRL+SHIFT+ESC to open Windows Task Manager. Click the **Processes** tab and look for **CIUMON.exe** in the list of processes. If **CIUMON.exe** is not present, then the service is not running.
- You can temporarily configure a CIUMon service to interact with the desktop so that a window will appear when the service is running. However, in this windowed mode, CIUMon will not start until you log on, and it will stop when you log off. After you are finished troubleshooting, you should turn off this mode so that CIUMon will run even when you are not logged on.

5.8.5 Running RoviSys Turbo automatically

If DBDOC connects to your CIU using RoviSys Turbo, then in order to run CIUMon automatically, you will need to run RoviSys Turbo automatically.

Recent versions of RoviSys Turbo can be run as a service. Running RoviSys Turbo as a service ensures that it is always available, both for CIUMon and for other programs that use it, such as a Composer or WinCAD EWS computer.

Older versions of RoviSys Turbo can be started automatically by CIUMon instead.

To run RoviSys Turbo as a service:

1. Make sure CIUMon will not try to run RoviSys Turbo. In CIUMon, on the **Options** menu, click **CIUMon Settings**.
2. Click the **Communication Modes** tab.
3. Clear the **Launch RoviSys Turbo on start-up** check box.
4. In RoviSys Turbo, on the **View** menu, click **Run as Service**.

To run RoviSys Turbo from CIUMon:

1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **Communication Modes** tab.
3. Select **Launch RoviSys Turbo on start-up**.

Note:

- RoviSys Turbo will usually perform best if it is running at high priority. To make sure that RoviSys Turbo runs at high priority, click **Priority** on RoviSys Turbo's **Edit** menu, and then click **High**.
- Some older versions of RoviSys Turbo do not have a feature to set the priority. For these older versions, use the **Set priority level of RoviSys Turbo** box in CIUMon to set RoviSys Turbo's priority. This feature will only work for these older versions of RoviSys Turbo.

5.8.6 Starting and stopping a CIUMon service with CIUWatcher

CIUWatcher can allow non-administrator users to stop and start CIUMon.

To allow non-administrator users to stop and start CIUMon, CIUWatcher looks for the existence of some file. When this file is deleted or moved, CIUMon is stopped. When this file is created or replaced in its original location, CIUMon is started again. By setting CIUWatcher to look for a file that any user can create or delete, you can allow any user to stop and start CIUMon.

To allow non-administrator users to stop and start CIUMon:

1. Decide where you will put the file that CIUWatcher will look for. Make sure that it can be moved or deleted and created by the users you would like to be able to stop and start CIUMon, and not by other users. Create the file.
2. Start CIUMon.
3. On the **Options** menu, click **Services**.
4. In the **Current CIUMon Services** list, click the service you would like other users to be able to stop and start.
5. Select **Run the service only when this file exists**.
6. Enter the location of the file for CIUWatcher to look for in the box below.
7. Click **Apply Changes**. Click **Close**.

5.8.7 Removing a CIUMon service

1. On the **Options** menu, click **Services**.
2. In the **Current CIUMon Services** list, select the service to remove.
3. Click **Delete Service**

Note: You can also remove a service using CIUMonController: in the CIUMonController window, click the service you want to remove and then click **Delete Service**.

5.9 Improving CIUMon's performance

There are various methods to improve the performance of CIUMon to make sure you are getting as much live data as possible and not slowing down the computer running CIUMon too much.

5.9.1 Limitations on how much live data you can get

Each CIU that CIUMon is connected to can provide around thirty values per second. Using RoviSys Turbo can greatly increase this amount. In many cases, a single CIU can provide enough live data for your users. This is because Hyperview is set up to use live data as efficiently as possible, and because there normally are not many users accessing live data all at once. If you need to control access to live data so that the users who need it most can get it, please look at Managing access to live data. If you need to get more live data, please look at Getting more live data.

Check your communication statistics to see how much live data has been requested from your system. Comparing the amount of data requested to the amount of data sent can indicate whether CIUMon is providing data quickly enough to meet the needs of your users.

If you find that one or two individual users are using much more data than others, it may be useful to review best practices for using DBDOC live data.

5.9.2 Best practices for using DBDOC live data

Because there is a limit to how much live data you can get in DBDOC, a user who consumes a large amount of live data resources can greatly reduce the amount of live data available to everyone else. Everyone should use as much live data as they need in order to make the best use of DBDOC when they are analyzing their system, but we recommend the following best practices to ensure that users do not use too much live data when they do not need it.

Check your communication and usage statistics to see how much live data has been requested from your system, and to check if there might be any one or two users who are using far more live data than everyone else. If so, it can be helpful to review how these users are using DBDOC and see if they are following these best practices.

Ensure that you are not getting data for a block that is offline or that doesn't exist in your system.

For example, if you are using an out of date .dbdoc file and you try to get data for a sheet that shows an old block that has since been deleted from your system, that will greatly slow down live data for all of your users and greatly reduce the amount of live data that DBDOC can produce. This is because the request for data will stay on the loop until it times out, after about five seconds, so for those five seconds live data might hang for everyone. If this sort of thing happens repeatedly (for example, if a user is trying to look at an entire sheet of old blocks that no longer exist in your system), it could cause live data to become entirely unavailable.

Use the camera to get Hyperview live data instead of the clock.

Using the clock button to get live data uses up roughly 10 times more live data than using the camera so we recommend that all users should use the camera for most work -- you should only turn on the clock if you really need it, and you should be diligent about making sure that you turn it off when you are done with the work that required it. Otherwise, you could be using up as much data as 10 more responsible users.

Close or minimize the Hyperview window when you are finished using it.

If Hyperview is left running on a computer after you are finished with it, it can continue asking for data. Closing Hyperview when you are done with it, or even just minimizing it, can help a lot to make sure that Hyperview isn't getting a lot of data that users don't actually need. (Hyperview will not get on-document data while it is minimized, but it will still get data for the watch window.)

Pause data collection in watch window groups when you are finished with them.

The watch window is mainly intended to be used as a short-term analysis tool -- for most watch window work, we expect that you will trend data for a few minutes and then analyze what that data is doing and then turn off the trending in the watch window. (To turn off data collection: on the **Group** menu in the watch window you can click **Pause All Blocks** to stop getting data for that watch window group; or as a quicker shortcut you can click the clock icon so that it turns white.) If you leave a lot of watch window groups running, that can consume a lot of live data. You can use the watch window to trend blocks for a longer period of time, but the most important thing is to make sure that you turn it off when you are finished getting the data you need -- otherwise, it will keep getting data when Hyperview is open, even if the watch window itself isn't being shown. If you need to keep track of what a block is doing over a very long period of time, or if you want to constantly monitor something all the time, that is the sort of situation where it is better to use a dedicated historian tool instead of the watch window.

Be careful when opening multiple Hyperview windows at once.

If you keep opening new Hyperview windows without closing old ones, each Hyperview window could keep getting data -- once you have a lot of them open, it is easy for Hyperview to be requesting a lot of data at once.

Turn off Show Live Specs when you are done looking at live specs.

If you are using Show Live Specs and you turn on Live Specs for a lot of blocks, it can take more data to display the live specs because the live specs don't benefit from OPC90's turbo mode. We recommend that when you are done looking at live specs, it is best to turn off Show Live Specs to make sure that you are not getting additional live specs when you don't need them.

5.9.3 Getting more live data

If you are not getting enough live data, there are several actions you can take. To make sure that you are getting as much data as possible from your current hardware, you should make sure CIUMon is running at

high priority. You may also wish to adjust CIUMon's timing. As well, make sure that CIUMon is connected at a baud rate of at least 600.

In addition using RoviSys Turbo can greatly increase the amount of live data you can get.

Making sure CIUMon is running at high priority: CIUMon runs at high priority by default. The only reason it would run at lower priority is if someone changed the priority deliberately, perhaps to reduce CIUMon's impact on computer performance. Since lowering the priority can slow down live data, you may wish to raise the priority again.

To check whether CIUMon is running at high priority:

1. Press CTRL+SHIFT+ESC. Windows Task Manager will open.
2. Click the **Processes** tab.
3. Locate **CIUMON.exe** in the list of processes. Look at the **Base Priority** column for **CIUMON.exe**. It should be "High".

Note: If there is no **Base Priority** column, then on the **View** menu, click **Select Columns**. Select the **Base Priority** check box and click **OK**.

Adjusting CIUMon's timing: When CIUMon requests data, it first sends a request for the CIU to fetch the desired value. Usually the data is not ready immediately, so CIUMon has to send a second request for the CIU to respond with the fetched value. The delay between these two requests is fixed at 50ms and is not adjustable.

If the data was not returned, then CIUMon retries up to the number of times specified by *Number of Retries*. The interval between consecutive requests is set by the *Response Retry Delay*.

The default *Response Retry Delay* is 60 milliseconds. In systems where the CIU is used with RoviSys Turbo, this may be too short. Lengthening the delay can reduce the number of times CIUMon requests each value, which can improve performance. In other systems, the CIU may have the data ready in less than 110 ms. In this case, shortening the delay can improve throughput slightly.

To change the response retry delay:

1. On the **Options** menu, click **Settings...**
2. In the **Response Retry Delay** box, type the new response retry delay (in milliseconds).

5.9.4 Reducing the impact of CIUMon on computer performance

CIUMon should have less impact on your computer's performance when it is run automatically. When it is run manually, it will have less impact on performance when it is minimized.

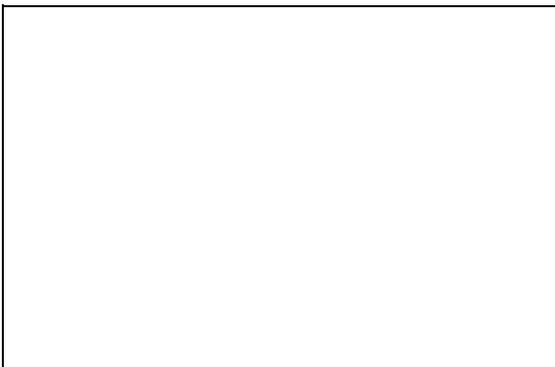
You can also reduce the amount of resources CIUMon uses by lowering it's priority. Lowering the priority will improve computer performance at the expense of CIUMon performance.

Changing CIUMon's priority: When CIUMon is run interactively, that is, when it either has been run manually or has been made to interact with the desktop, you can change the priority with which it runs.

The priority you set will affect all CIUMons that start (or restart) after you change the priority. However, it will not affect CIUMons that are already running, except the one in which you change the priority.

To change CIUMon's priority:

1. On the **Options** menu, click **Set Priority....** The **Set CIUMon Priority** window will appear.
2. In the list of priorities, click the priority at which CIUMon should run.
 - ◆ CIUMon performs best when it is running at **High** priority.
3. Click **OK**.



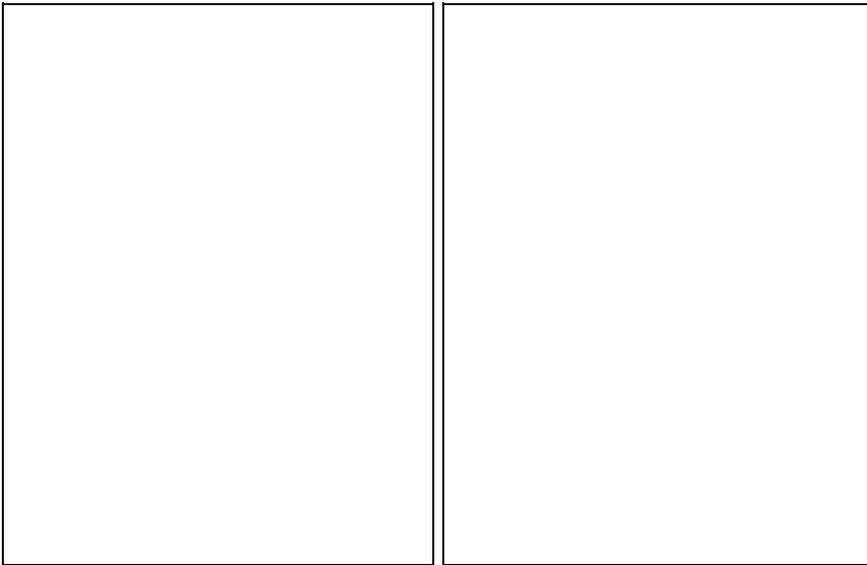
5.10 Managing access to live data

You may want to manage access to live data because the amount of live data available is limited, or for security or regulatory reasons.

With DBDOC, you can:

- Restrict access to live data;
- Track who is using live data;
- Allow some users to get more live data than others.

5.10.1 Restricting access to live data



Because the amount of live data available is limited, it may be useful to restrict access to it. DBDOC provides two ways to do this: by Windows logon name or by IP address. Do not use both methods at the same time.

Live data access restrictions are not configuration specific. They are shared by every CIUMon and every configuration running on a given machine.

Blocked users who try to access live data will receive an error message letting them know that they have been blocked.

Restricting access to live data by Windows logon names: A logon name can be either blocked or approved. If any logon names are approved, only approved logon names will be able to access live data. There is no reason to both approve and block logon names.

Note: If a username has ever been changed, you must block or approve the original username. For example, if user "Smith" was originally user "John Smith," but the username was changed, you must block "John Smith." Blocking "Smith" will not work. This is due to how Windows stores usernames.

To approve or block access to live data by a Windows logon name:

1. On the **Options** menu, click **Settings...**
2. Click the **Approved Users** tab.
3. Do one of the following:
 - ◆ To allow the newly added user to get live data, select the **Name approved for live loop monitoring** checkbox.
 - ◆ To prevent the newly added user from getting live data, clear the **Name approved for live loop monitoring** checkbox.
4. Type the logon name to be added into the **Enter the user's logon name** box and click **Add Entry**.

Note: If any logon names are approved, only approved logon names will be able to access live data. There is no reason to both approve and block logon names.

To remove a Windows logon name from the list of approved or blocked logon names:

1. On the **Options** menu, click **Settings....**
2. Click the **Approved Users** tab
3. In the **Approved/blocked Users** list, click the logon name.
4. Click **Delete**.

Restricting access to live data by IP address: IP addresses (or ranges of IP addresses) can be either blocked or approved. If any IP addresses are approved, only IP addresses that are both approved **and** not blocked will be able to access live data. Using this feature, it is possible to approve a range of IP addresses but block some addresses in that range.

Note: There is an exception to the rule that only IP addresses that are both approved and not blocked can access live data. If you block a range of IP addresses and then approve a **single** IP address in that range, you will be able to get live data from that single address. This makes it possible to block a range of IP addresses but allow access from specific computers in that range.

To approve or block live data access from a single IP address:

1. On the **Options** menu, click **Settings....**
2. Click the **Approved IP addresses** tab
3. In the **From** boxes, type the IP address. It will appear in the **To** boxes as you type.
4. Do one of the following:
 - ◆ To allow the IP address to access live data, select the **Range approved for live loop monitoring** checkbox.
 - ◆ To block the IP address from accessing live data, clear the **Range approved for live loop monitoring** checkbox.
5. Click **Add Entry**.

To approve or block live data access from a range of IP addresses:

1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **Approved IP addresses** tab
3. In the **From** boxes, type the first IP address in the range.
4. In the **To** boxes, type the last IP address in the range.
5. Do one of the following:
 - ◆ To allow IP addresses in the range to access live data, select the **Range approved for live loop monitoring** checkbox.
 - ◆ To block IP addresses in the range from accessing live data, clear the **Range approved for live loop monitoring** checkbox.
6. Click **Add Entry**.

To change an IP address or range of IP addresses in the list of approved or blocked IP ranges:

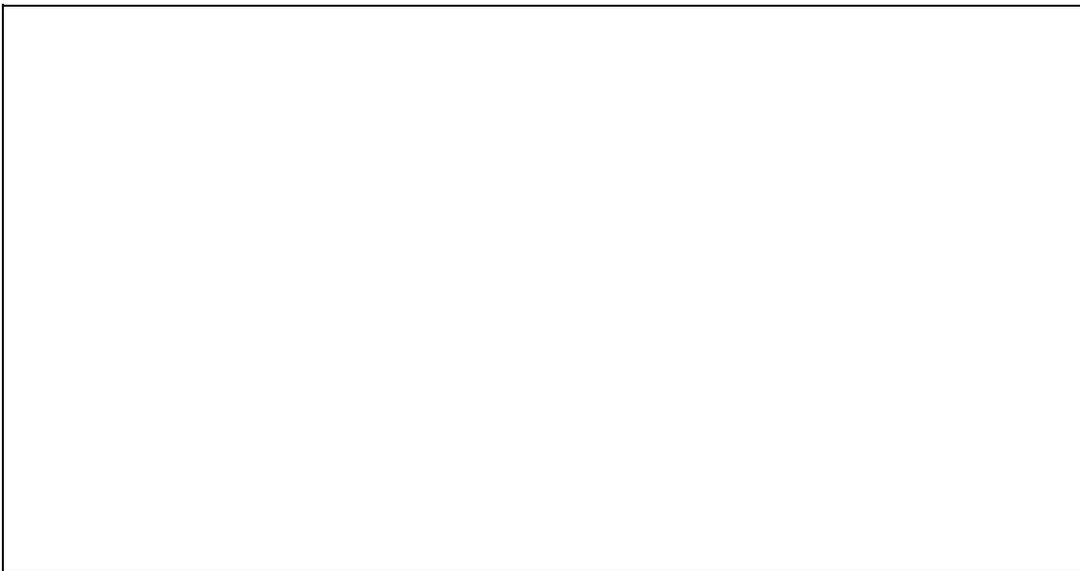
1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **Approved IP addresses** tab

3. In the **Range of approved IP addresses** box, click the IP address or range that you would like to change.
4. Click **Edit**.
5. Make any changes you want to this IP address or range. For example, you can change whether the IP address is approved or blocked. Then, click **OK**.

To remove an IP address or range of IP addresses from the list of approved or blocked IP ranges:

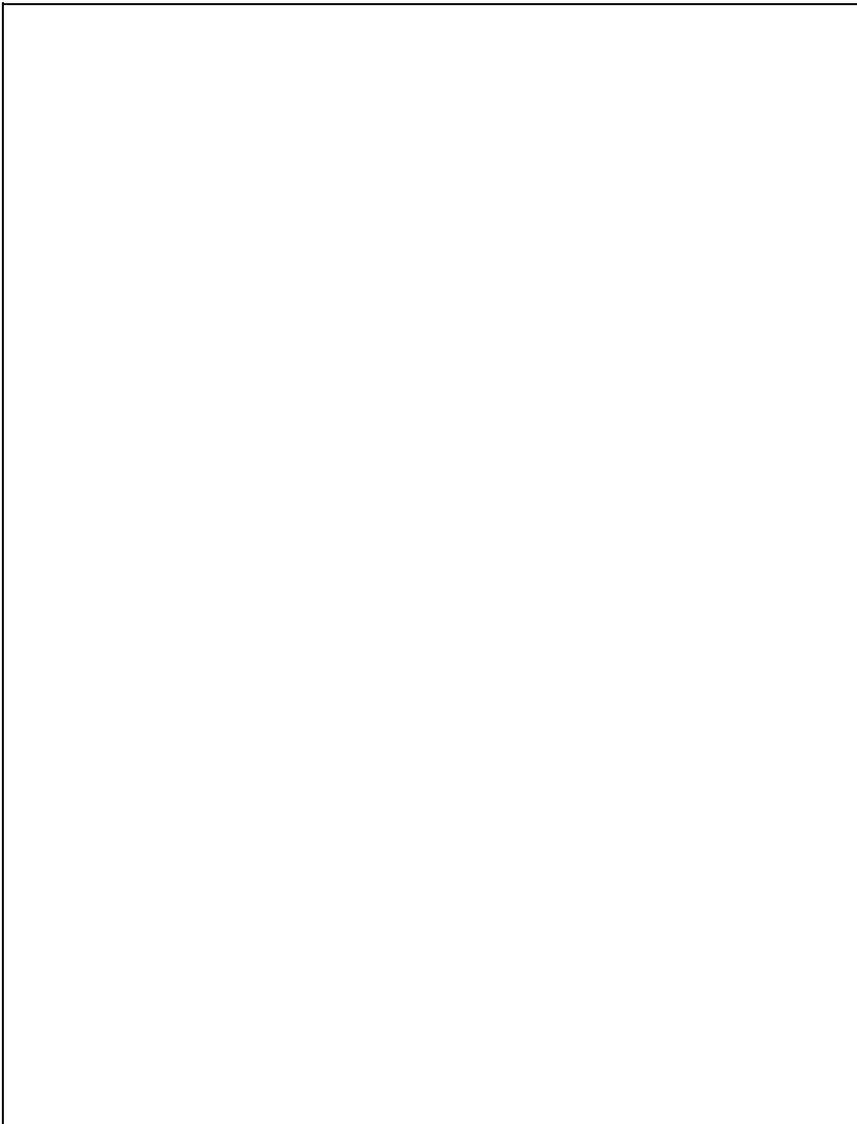
1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **Approved IP addresses** tab
3. In the **Range of approved IP addresses** box, click the IP address or range.
4. Click **Delete**.

5.10.2 Tracking who is using live data



CIUMon has two options for tracking usage. Communication statistics and usage statistics track slightly different information. They each provide a summary report and a detailed report, which display different information. These reports can be accessed through the Options menu. Statistics from both types of usage tracking can be saved to a file.

5.10.3 Allowing some users to get more live data than others



One way to deal with the limitations on how much live data you can get is to give some users more live data than others, by assigning them a higher user weight. For instance, a user with a user weight of two will get twice as much data as a user with a user weight of one. By default, all users have a user weight of one.

To assign a user weight:

1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **User Weights** tab.
3. Type the user's logon name in the **Enter the user's logon name** box.
4. In the **User Weight** list, click the user weight to be assigned.
5. Click **Add Entry**.

To delete an assigned user weight:

1. On the **Options** menu, click **CIUMon Settings**.
2. Click the **User Weights** tab.

3. In the **User Weights** list, click the user weight to be deleted.
4. Click **Delete**.

6 Build Reports and Analysis

6.1 Overview of the DBDOC build reports

When a project is successfully built, errors are identified by DBDOC.

Most of these errors can be viewed in Hyperview with the Error Browser, and more information for each error is available in the Individual errors documentation.

Some less common errors are documented in other .err files, and other helpful files are also generated.

Build results can be compared with previous builds using Winmerge and Management Of Change (MOC).

6.2 Build reports

Many errors can now be viewed and dealt with more easily in Hyperview's error browser. The most significant errors may be identified by sorting by severity. See Individual errors for more information about these.

Each time you build with DBDOC, there are additional error files in the project folder. To find the most recent **.ERR** files, open the **ERROR_FILES** folder, then the **MOST RECENT** folder.. (A file in our build system called ERRFILTER.CFG controls the filtering into these files.)

The error files are grouped as follows:

- Error summary reports, which as of DBDOC 11.2 include a **Hotspots Report** with a count of each type of hotspot:
 - ◆ DBDOC.ERR;
 - ◆ DBDOC_RESIDUAL.ERR;
 - ◆ DBDOC_SUMMARY.ERR
- Error messages not included in the error browser

Other information about your build is contained in:

- General build reports - files other than .err
- Management Of Change (MOC)

For more information and assistance from GMCL, send an error.7z file for analysis.

6.3 General build reports

BuildPlus generates a number of reports from each build of your system. Most are generated as files in the project folder, and some are found in Hyperview, in the table of contents.

AFW# folders: for extracted 800xA afw backup data, and includes OPCTags where relevant, for exported generic OPC Tags

- **afw_extractor.err** Log file containing all output and subprocess messages from afw_extractor as well as command line statements.

DECOMPOSER# folders (one for each .ebp file in the build. To see which DECOMPOSER folder refers to each Composer .ebp database, refer to SystemFolders.txt):

- **CFGFiles.txt** lists all .cfg files for the Composer .ebp database.
- Decomposer.err and Decomposer.err.UNI contain Decomposer errors and warnings as listed in the sub-process window during the build.
- **DISDirs.txt** lists <ConductorVMS graphics folder name>, "<Description from the folder's console.cfg file>" or is empty if there are no ConductorVMS graphics.
- SPEVersion.txt lists the SPE version information.

DOC folder: for AC 800M, including doc_file_processing.log

ERRORS folder: contains a MOST RECENT folder, and dated folders for each build (as long as the option to Archive Project Error Files is on). Each folder contains files for errors not appearing in error browser, as well as ZIP.TXT and ERROR.7Z and the dated folders contain archived dmpcfg files.

EXPORTS folder:

- the CFG and EWS CFG folders contain files for comparing modules' specification values. For more information, see Using Config Parser or contact GMCL.
- BLOCKS.DBF contains block, function code and reference information
- **FCnnn.TSV** and FCnnn.DBF - Extracted specifications from CAD and CLD files files include the sheet name and function code specifications for each function code
- PG2_Graphics.TSV generated for BBA graphics for analysis of file names
- PCU Map: PCU.csv, also described in Hyperview Help
- SPEC.MDB - Used to link the files in an Access database to link the FCnnn.DBF files
- Tags_in_graphics.dbf - A database with a record for each tag in each graphic
- AC800M Tags.tsv, when AC800M files are included

FC folder: contains copies of the function code documentation files, if they were included in the build or added manually

LOGS folder contains a log of each build of the project and a log of all the subprocess messages for the build (when BuildRoot Options checkboxes to log these files are checked).

MOC folder contains management of change files. See Management Of Change (MOC)

PDF folder contains doc.xml files for debugging PDF processing

SPO folder contains deobfuscated forms of SPO .mimic and .bin files for internal use and client support.

TAG folder:

- XLS# folders contain file xlsx2csv_.err which logs information related to processing each xls or xlsx file, and also .csv files extracted from each sheet of the .xls or .xlsx file
- GPI, HGS, HPG folders when these mdb files are built, containing one subfolder for each mdb
- Files in the MDBn and TDTn folders are explained under Extracted and synthesized "flattened" databases
- Other files may include attribute (.att) files if AutoCAD drawings were included in the build.

THUMBNAILS folder contains cached thumbnail images and a thumbnails.log file.

6.4 Other build report files

- File_statistics.csv, contains information about the files built into the project
- Blank_CAD.txt - Shows CAD files and CLD sheets that were empty
- SystemFolders.txt (DecomposerDir.txt prior to DBDOC 11.6), which shows which extracted folders (e.g. DECOMPOSERn, AFWn, etc) refer to each source in the system
- Sys_Info.txt, which shows system information such as the graphics and configuration used.
- DBDOC_Components.txt, which shows the versions of DBDOC programs used in the build
- dbtype_processing.txt, which shows status of processing of files with dbtype.exe
- Duplicate_Assyst_Files.txt lists Assyst graphics filenames that are duplicated
- duplicate_urls.txt lists which URLs had duplicates and the amounts of them. Can be found in either MOST RECENT or the dated folders in ERROR_FILES, and also in ERROR.7z.
- Emptyfiles.txt - Shows source files that were empty
- Excluded_Files.txt - Lists all the files that were excluded in the project due to issues during the build.
- F(X)_Specs.txt - List of interesting function generator specifications
- FCList.txt - List of the number of instances of each function code
- mdbtype_processing.txt, which shows status of processing of files with mdbtype.exe
- New_tags.txt - A list of all new tags that have been added since the last build
- newfile.txt - contains the last generated .dbdoc filename
- PGP_Files.nod - contains a list of SPO/PGP files identified by BuildPlus
- TagUsage.csv - Summary of tags used in the system
- taglist_exporter.err - which lists tags where values of fields are different in configuration from

databases.

- Unused Tag Lists - List of tags that are not in any graphic
- Unused-unconfigured_tags.txt - Not on any graphic but not on any CLD or CAD sheet
- Unused-Alarm_tags.txt - Not on any graphic but should alarm
- Unused_Tags.txt - All tags that are not on any graphic and not in the specialized lists
- Global taglists - A master tag list with all the tags in the system
- Unused_symbols.txt - This file lists symbols that are unused by the graphics
- Significant Change Report, found under **System Information** in Hyperview, this report lists significant change of exception reported values in units and percent change
- Undefined tags lists tags found on graphics but not defined in a database.
- Undefined graphics lists any hotspots that are supposed to link to other documents, but don't.
- Undefined symbols lists missing symbols or sub-models.
- AutoCAD files with tag links found under **System Information** in Hyperview, described in Hyperview Help documentation
- AutoCAD files with no tag links found under **System Information** in Hyperview, described in Hyperview Help documentation
- pg2_unidentified_record_types.txt
- pg2_dependencies.txt - Lists the elements used by each graphic. The unindented lines list the graphic name, and the indented lines below each graphic list the elements used by it.
- Short Tags.txt ? List of tags that were not matched with text because they are shorter than the minimum tag length on CAD sheets, etc. since the match was explicitly blocked because the tag length was shorter than the minimum tag length (set by the min_tag_length config file command through a custom edit). Can be found in either MOST RECENT or the dated folders in ERROR_FILES, and also in ERROR.7z.
- AutoCAD files with no text found under **System Information** in Hyperview, described in Hyperview Help documentation
- Index of Non-Block Points, found under **System Information** in Hyperview, lists points found in databases that have no Loop, PCU, Module and Block information

It is possible to include text files generated by DBDOC in a build.

It may also be possible to extract a list of all IREF's and OREF's into a file. For more information contact GMCL.

6.4.1 FCnnn.DBF - Extracted specifications from CAD and CLD files

During a build, BuildPlus produces a separate FCnnn.DBF file (e.g. FC004.DBF) for each function code used in the system, found in the Exports folder of the project directory. Each includes the sheet name, the tagname, if defined, and all the specifications for that function code. As of DBDOC 11.4, block number 0 and

negative blocks are no longer included in these files. It is essentially the same as the FUNCTION CODE INDEX. This report can be linked as tables in a Microsoft Access database using SPEC.mdb.

These reports are useful for regulatory reports, analysis of exception report traffic, and countless other applications.

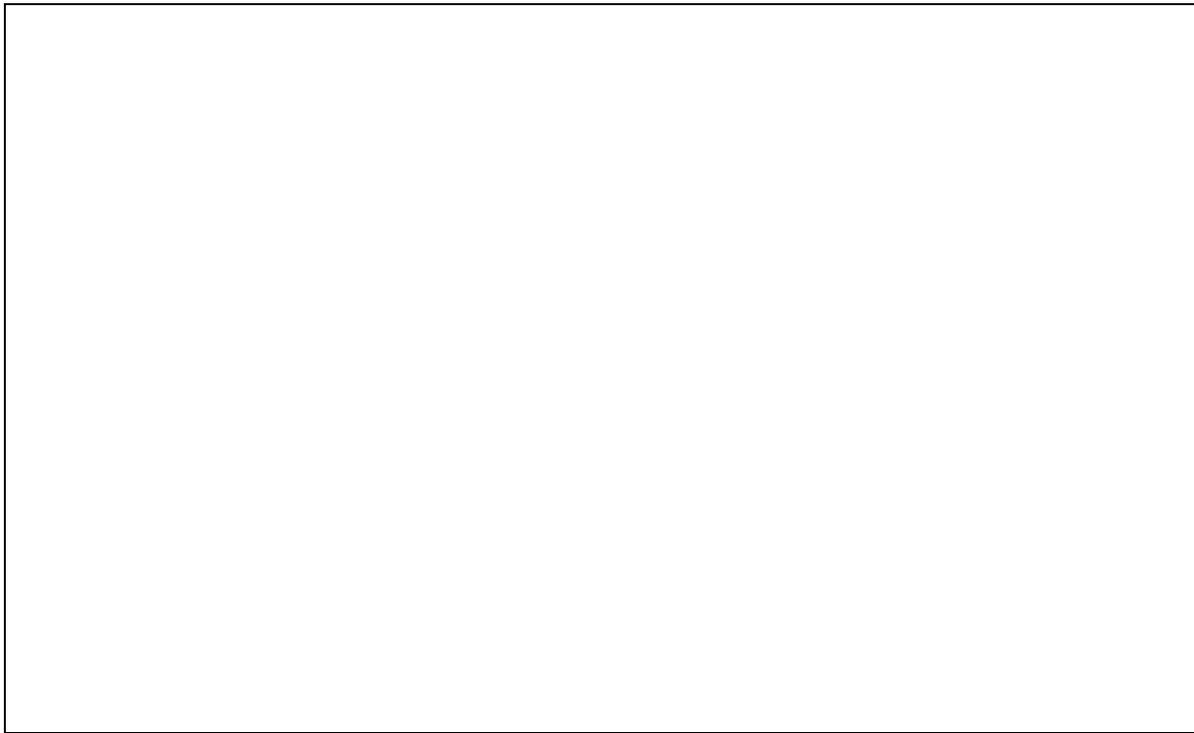
If you want to use these files, but plan to rebuild your project, move these files to another folder before starting the rebuild. (If the files are open during a build, the records will be appended to existing tables rather than getting cleared out before appending. But if this has happened, close the spec.mdb Access database and do the build again.)



6.4.2 SPEC.MDB - Used to link the files in an Access database

SPEC.MDB links the FCnnn files as tables in a Microsoft Access database. If you open spec.mdb with Microsoft Access, you should easily get an idea of the ways you can use this information. In this database you have zero, span, limits, tuning parameters, and so on - all the specs and block numbers in the whole system.

To use these FCnnn.DBF files in any other structure, you should copy them from where they are created, as the FCnnn.DBF files cannot be appended to if active.



6.4.3 F(X) Specs.txt - Diagnostic data file FC 1 spec messages

F(X) Specs.txt (formerly called FC001.txt) lists issues identified by DBDOC for each F(X): Function Generator [FC1]. The same issues are included in individual error messages, based on the type of issue. Examples include:

- X-coordinate adapted - S2 (2850.00) of Module 3,05,04 Block 1308, or Y-coordinate adapted - S9 (-82.31) of Module 3,05,04 Block 1308. See also [029] X coordinate adapted and [030] Y coordinate adapted
- X co-ordinates out of order for F(x) Module 3,05,04 Block 1087. See also [033] X coordinates out of order for F(x)
- F(x) discontinuity (step) in Module 3,06,02 Block 4204 at S10 X value of 1.500. See also [031] F(x) discontinuity
- F(x) Module 3,15,02 Block 8454 uses large spec values: output value may be unexpectedly extreme. See also [034] F(x) uses large spec values

6.4.4 FCList.txt - List of the number of instances of each function code

The FCList.txt file lists the number of times each function code appears in the system.

FC	1	225 FUNCTION GENERATOR	FC__1
FC	2	379 MANUAL SET CONSTANT	FC__2
FC	3	41 LEAD/LAG	FC__3
FC	4	48 PULSE/POSITIONER	FC__4
FC	6	79 HIGH/LOW LIMIT	FC__6
FC	7	35 SQUARE ROOT	FC__7
FC	8	15 RATE LIMIT	FC__8
FC	9	1167 TRANSFER - ANALOG	FC__9
FC	10	201 HIGH SELECT	FC__10
FC	11	72 LOW SELECT	FC__11

We use this list to find which systems have particular, rare function codes by searching it. It allows us to find the few places in our encounters that need the programming we have done or are considering doing.

This list has helped us solve obscure problems that affect very few clients. This has proven its value.

DBDOC builds all the function codes in your system into Hyperview, and creates a FUNCTION CODE INDEX. For more information see the Function Code Index article in the Hyperview Reference Guide.

6.4.5 TagUsage.csv - Summary of tags used in the system

The TagUsage.tsv file is essentially the same as the tag index pages displayed in Hyperview. It allows you to import a tagname, plus the definition of that tag in each of the databases or tag lists you have built. Tags that are listed in the Non-block points index are not included in TagUsage.tsv.



For more information, see the article Tag Index in Hyperview Help.

6.4.6 Emptyfiles.txt - Shows source files that were empty

The Emptyfiles.txt file shows source files in the system that were actually empty. You can cross-check this list for files that might have become corrupted.

```
Empty file: <D:\BAILEY SITES\PRB_DCS\L1\P40\M2\1400201A.CAD>
Empty file: <D:\BAILEY SITES\PRB_DCS\L1\P40\M2\14002.VFY>
Empty file: <D:\BAILEY SITES\PRB_DCS\L1\CONS61\HELP.DRS\HELPIB1.TXT>
```

This file can be found in the project folder, or it can be accessed from the **System Information** section of the **Table of Contents** in Hyperview.

6.4.7 Blank_CAD.txt - Shows CAD files and CLD sheets that were empty

The Blank_CAD.txt file lists CAD files and CLD sheets that have no function blocks and no significant text. Often, you will find the word "SPARE" on them. These sheets are placeholders that allow parallel names to be developed in different parts of the system.

We actually are willing to let clients have blank sheets handled for no cost, as we adjust the document count down to take them into account.

```
Blank CAD Sheets Report
Blank Sheets in Module 2,03,02: 1
\\SIRIUS\CLIENT-DATA REVISED\MCBPM#~1\PM1_PR~1\L2\P3\M2\20302E8A.CAD
Blank Sheets in Module 2,03,08: 3
\\SIRIUS\CLIENT-DATA REVISED\MCBPM#~1\PM1_PR~1\L2\P3\M8\2030828A.CAD
\\SIRIUS\CLIENT-DATA REVISED\MCBPM#~1\PM1_PR~1\L2\P3\M8\2030829A.CAD
\\SIRIUS\CLIENT-DATA REVISED\MCBPM#~1\PM1_PR~1\L2\P3\M8\2030873A.CAD
Blank Sheets in Module 2,04,04: 1
\\SIRIUS\CLIENT-DATA REVISED\MCBPM#~1\PM1_PR~1\L2\P4\M4\2040458A.CAD
```

6.4.8 New_tags.txt - A list of all new tags that have been added since the last build

The new_tags.txt file lists all the new tags that have been added since the last build. It is located in the Exports folder of your project directory.

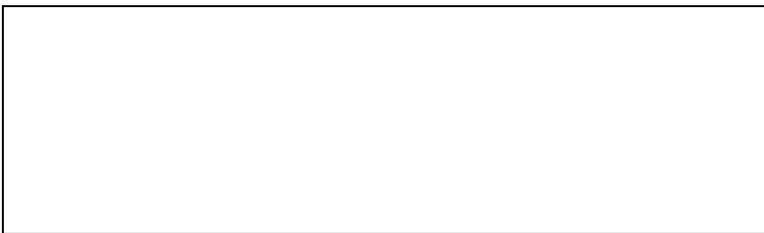
The intention is to help the user make sure that the tags get transferred to the console database, so that graphics using the tags will work. Use it as a checklist when you have to define tags by hand or export them to transfer them to the console processor.

For example, the following are new tags in a build:

```
15CEMENTLEVEL
16CEMENTELEVEL
16CEMENTLEVEL
1_2FLAPSTART
1_2TRAVELOVER
1_BESTA_FEEDER
1_COOLERHOTAIR
1_F_PIPE_TEMP
1_PRIMARY_AIR
1ABUNDCEAVAIL
1ABUNDCEOLOAD
```

6.4.9 Tags_in_graphics.dbf - A database with a record for each tag in each graphic

The tags_in_graphics.dbf is a database (found in the Exports folder) with a record for each tag in each graphic. It allows for verification of all graphics on which tags are found. This file includes the folder path for the graphic, to support systems where the same graphic name is used in multiple consoles.



6.4.10 Unused Tag Lists - List of tags that are not in any graphic

Unused Tag Lists:

- List tags that are not in any graphic
- Contain tagname, sheet, loop, PCU, module and block
- Are separated into "unconfigured", "alarm" and "all others"

You should check these for surprises.

- Unused-unconfigured_tags.txt - Not on any graphic but not on any CLD or CAD sheet
- Unused-Alarm_tags.txt - Not on any graphic but should alarm
- Unused_Tags.txt - All tags that are not on any graphic and not in the specialized lists

This file can be built into Hyperview by making a copy of the file, then adding Text files in Wizard, pointing at the project folder. For more information, see the Module: List of Unused Tags article in Hyperview Help.

6.4.10.1 Unused-unconfigured_tags.txt - Not on any graphic but not on any CLD or CAD sheet

The Unused-unconfigured tags.txt file lists the tags that are in the database, but not found on any graphic, CLD sheet or CAD sheet.

TAGNAME	LOOP	PCU	MODULE	BLOCK	TAGTYPE	SOURCE
1BESTASETPOINT	1	1	30	50	Analog	Built
1BFNOXINSTALM	1	1	30	4033	Digital	Built
1CLINKERTODATE	1	1	30	86	Analog	Built
1CM1CCOMPSELSW	1	5	2	9933	RCM	Built
2BESTASETPOINT	1	2	22	50	Analog	Not Built

This list is automatically built into the project file. For more information, see the Module:List of Unconfigured Tags article in Hyperview Help.

6.4.10.2 Unused-Alarm_tags.txt - Not on any graphic but should alarm

The Unused-Alarm tags.txt file lists tags that are not on any graphic but should alarm. DBDOC puts unused tags into this category when they are not on any graphic and the following conditions hold:

- for any tag - where the alarm group value is not zero
- for digitals - the source DO/L block does not have S2=2, which suppresses alarming on a change
- for analogs - the source AO/L block seems to have valid low and / or high limits, where valid means that, given the block range defined by its zero and span values, at least one of the limits is within span units of that range

Here are some analog examples:

- zero = 0.0, span = 100.0, if either limit is between -100.0 and +200.0, we call it an alarm tag
- zero = -50.0 span = 100.0, if either limit is between -150.0 and +150.0, we call it an alarm tag

- zero = 0.0 span = 100.0, if both limits are above +200.0, we will not call it an alarm tag, even though the working range might be +300.0 to +400.0 and the limits set appropriately.

TAGNAME	LOOP	PCU	MODULE	BLOCK	TAGTYPE	SOURCE
140SCREWLEVELA	1	1	30	2338	Analog	1013035
140SCREWLEVELB	1	2	30	2339	Analog	1023035
140SCREWLEVELC	1	4	8	2394	Analog	1040836
1ABUNDCESTART	1	5	6	2300	RMCB	1050635
1BBOTDCESTART	1	5	6	2400	RMCB	1050637
1BBUNDCESTART	1	5	6	2350	RMCB	1050636
1BCONVEYSEQSW	1	5	6	2290	RCM	1050634
1BCONVEYSTART	1	5	6	2250	RMCB	1050634
1BESTAHIGHAMP	1	1	2	8165	Digital	10102B3

This file can be built into Hyperview by making a copy of the file, then adding Text files in Wizard, pointing at the project folder. For more information, see the Module:List of Unused Alarm Tags article in Hyperview Help.

6.4.10.3 Unused_Tags.txt - All tags that are not on any graphic and not in the specialized lists

The Unused_tags.txt file lists all tags that are not on any graphic and not in the specialized lists.

The basic concept of unused was created to mean that the tag did not appear on any graphic. This has been improved to also require that the tag not have a Primary Display or Alarm Vector entry. The Primary Display and Alarm Vector entries allow a chosen display to be called up from the operator console alarm summary, so we treat the tag as being "used", even though it does not appear on a graphic.

For example:

TAGNAME	LOOP	PCU	MODULE	BLOCK	TAGTYPE	SOURCE
1BCLIBELTONOFF	1	5	2	5036	Digital	1050267
1BFAIRSERVICE	1	1	30	1800	Digital	1013025
1BFALLDRIVESRUN	1	1	17	6622	Digital	10117A6
1BFCOCORRECT	1	1	30	3599	Analog	1013060
1BFLAPPULSE	1	5	6	2812	Digital	1050645
1BFO2SERVICE	1	1	30	1820	Digital	1013025
1BFSTACKAVAIL	1	1	17	6459	Digital	10117A3
1BFSTACKRTS	1	1	17	6471	Digital	10117A3
1BFSTACKRUNFOR	1	1	17	6472	Digital	10117A3
1BFSTACKRUNREV	1	1	17	6473	Digital	10117A3

This file can be built into Hyperview by making a copy of the file, then adding Text files in Wizard, pointing at the project folder. For more information, see the Module: List of Unused Tags article in Hyperview Help.

6.4.11 Extracted and synthesized "flattened" databases

In the project folder, the \TAG\MDBn, \TAG\TDTn and \TAG\XLSn folders contain .dbf files that are flat databases containing all the tags in your consoles and configurations. They can be pretty useful.

- Every tag and trend database we process has had the relational information expanded
- This includes full taglists for Process Portal B, Operate IT and Conductor NT; plus all taglists in Composer and WinTools
- These "flattened" databases show alarm comments, logic states and engineering units as text rather than indexes
- Our synthesized databases can be imported into other database processing tools for further analysis, including Microsoft Access and Excel, that may be useful

6.4.12 Global taglists - A master tag list with all the tags in the system

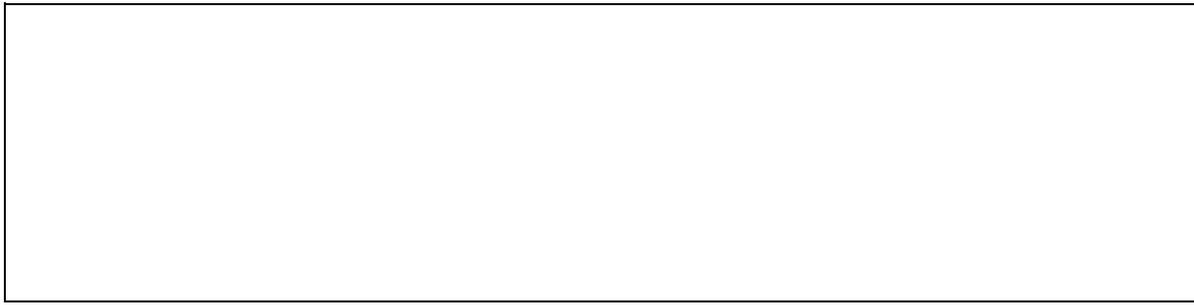
Master taglists are generated with all the tags in the graphics or configuration system. This is useful for validating server-based approaches, where the tags are distributed among various databases and lists.

Here are the taglists you might encounter:

- 800xA_taglist.dbf - all the tags built into an 800xA console system
- Composer_taglist.dbf - all the tags known to Composer
- ConductorNT_taglist.dbf - all the tags built into a Conductor NT console system
- Generic_taglist.dbf - all the tags known to or used in a system coming from .DBF or .TTG files
- OperateIT-PPB_taglist.dbf - all the tags built into a PPB console system
- WinCAD_taglist.dbf - all the tags known to WinTools

The meaning of these taglists varies with the console. For some systems, this gives you all the tags in the console. However, for others, you must reconcile individual taglists with their consoles.

These lists are useful for studying all the tags. The DBDOC processing of individual consoles is intended to find places where tags are not in the correct taglist for the console.



6.4.13 Unused_symbols.txt - This file lists symbols that are unused by the graphics

This is a list of symbols in each symbol or submodel folder that were not used in the graphics as they were built. It can be used as a list of symbols or submodels to archive. It is also intended to help clean out unused symbols and submodels, and is especially useful for upgrade and conversion projects.

Note that the organization is by folder, so a symbol might be unused in one folder but used in another.

If you are using this list, it is worthwhile combining your symbols into a single folder to be sure that you do not make the mistake of removing a symbol that is used. Alternatively, make sure you do a DBDOC build after cleaning out (archiving, not deleting) the symbols that are unused before concluding you have a good picture of the usage of the symbols in your system.

```
Unused Symbols/Submodels:  
C:\BUILDS\ESA\DECOMPOSER1\CONSOLE SYMBOL:  
ANALOG1.dy  
ANCBOL1.dy  
ANCREAL1.dy  
ANDCSRT1.dy  
ANLGCPOP.dy  
ANLGCTL1.dy  
ANLGCTL2.dy
```

6.4.14 Significant Change Report - This report lists significant change of exception reported values in units and percent change

During each build, BuildPlus creates a Significant Change Report in the project file, which lists all of the exception report blocks in your system that are used as tags or imported into other PCUs. It shows their significant change value in units and percent change. In the case of imported blocks, it links to where the block is used.

The built-in links to the import blocks are very useful. They allow you to cross-check what significant change is being transferred for the purposes of control and tabulation. A mismatch among inputs in another module can cause grief.

The links for all tags allow you to see how the tag is used to see if the significant change setting is appropriate. Both too small and too big can be problems for your operators in their understanding and operation of the process.

This report is always built into the project file and can be seen in Hyperview under the System Information chapter in the Table of Contents in Hyperview.

6.4.15 Undefined tags - lists tags found on graphics but not defined in a database

During each build, BuildPlus creates an UNDEFINED TAG INDEX, which lists tags that were not found in the database, and links to where they are used on graphics.

This report is always built into the project file, and can be seen in Hyperview under the **System Information** chapter of the **Table of Contents**.

For more information, see the Index of Undefined Tags article in Hyperview Help.

6.4.16 Undefined graphics - lists any hotspots that are supposed to link to other documents, but don't

During each build, BuildPlus creates an index of the undefined graphics in your system. It lists and links to any hotspots that are supposed to link to other documents, but don't.

This report is always built into the project file and can be seen in Hyperview under the System Information chapter of the Table of Contents.

For more information see the Index of Unconfigured Graphics article in Hyperview Help.

6.4.17 Undefined symbols - lists missing symbols or sub-models

During each build, BuildPlus creates an index of symbols that don't link anywhere in the Missing Symbols/Submodels file.

This index is always built in Hyperview and can be seen under the **System Information** chapter of the Table of Contents.

DBDOC will also create a text file listing unused symbols in your system. For more information, see the Unused symbols article in Build Reports Help.

6.5 Management Of Change (MOC)

The Management of Change (MOC) analysis allows comparison of two builds to see changes that have taken place. It is automatically executed with each build. MOC shows deleted, added or changed files. MOC now shows entire folders when appropriate. MOC also analyzes all tags in your databases and shows tags that have been modified, added or removed. From **BuildPlus | Tools | Project Options**, you can select two builds for comparison. If you keep the project files, you can print the old and new versions from a comparison of two builds.

For more information, see also **Management of Change Report** in Hyperview Help.

6.5.1 Understanding Management of Change

Each time a build is performed, if the MOC tool is enabled, a file is generated called base<date>-.txt that is used for the basis of comparison between builds. Before the hyperlink compile, the two base files for the current and previous builds are compared and a comparison file MOC<date>-.txt generated with the changes.

We take special pains with files to suppress access dates in CAD files, for example, looking only for content changes.

The base file contains information about nearly every item in the project:

- CADEWS drawings (.cad) - including Composer CLD files
- Conductor models and submodels (.m1, .m2)
- Graphics (.dis, .dr., .dt)
- Graphics symbols (.dy)
- Database records (excluding PI and XLS databases) (.dbf)
- AutoCAD / MicroStation drawings (.dwg, .dgn, .dxf)
- Applications (.txt, .b90, .ab, .rtu)
- Ladder configuration files (.lad)

Information stored in the base file:

- Name of the project file
- Database records: filename, tag name, CRC
- Other: filename, title / key, timedata, CRC

Changes are reported if:

- a database record is missing, new or its CRC has changed
- a file is missing, new or its CRC has changed (a change in the timedata stamp will not trigger a difference)

Why might you think MOC means "Mismanagement of Change"?

- If you are practicing management of change, MOC shows you what you already know.
- If you actually find an undocumented change, it means you found one that was mismanaged.

Each run of MOC, after the first initial build, will result in the creation of MOC_DIFFERENCES_<date&letter>.TXT. This file will contain the changes detected between the current build and the old build. BuildPlus processes this file to categorize all the changes into the types of differences and presents it in an easy-to-view and easy-to-use Treeview to allow the user to select specific records or files for printing. The Management of Change report is included automatically in the project file.

6.5.2 Using Management of Change with Scheduled Builds

You will notice that your project files are automatically renamed by default with the current date. Again, by default, the Management of Change report (MOC_Differences_<date>.txt) shows you changes between the current build and the previous one.

Thus, to get even more benefit from MOC, use it in conjunction with a regularly scheduled build, so you get a comparison of the previous build automatically.

For more information, see Running scheduled builds in BuildPlus help documentation.

6.6 Build reports

Many errors can now be viewed and dealt with more easily in Hyperview's error browser. The most significant errors may be identified by sorting by severity. See Individual errors for more information about these.

Each time you build with DBDOC, there are additional error files in the project folder. To find the most recent **.ERR** files, open the **ERROR_FILES** folder, then the **MOST RECENT** folder.. (A file in our build system called ERRFILTER.CFG controls the filtering into these files.)

The error files are grouped as follows:

- Error summary reports, which as of DBDOC 11.2 include a **Hotspots Report** with a count of each type of hotspot:
 - ◆ DBDOC.ERR;
 - ◆ DBDOC_RESIDUAL.ERR;
 - ◆ DBDOC_SUMMARY.ERR

- Error messages not included in the error browser

Other information about your build is contained in:

- General build reports - files other than .err
- Management Of Change (MOC)

For more information and assistance from GMCL, send an error.7z file for analysis.

6.6.1 Error summary reports

6.6.1.1 DBDOC.ERR - A log of the entire build

The DBDOC.ERR file is a record of the entire build. Showing the processing stages in this detail allows us to diagnose problems with files that have problems that we have not encountered before. The reason this "log" is so verbose is that it allows us to find problems at pretty well any phase of our processing.

6.6.1.2 DBDOC_RESIDUAL.ERR - Any messages that are not included in our documented error messages

The DBDOC RESIDUAL.ERR file contains any messages that are not included in our documented error messages. It also contains our statistics on the build, program versions and information on your build computer that helps us assist you.

```
DBDOC Kit Version: 10.30
Decomposer Converter      Version r2011-01-11; Build Date Jan 11 2011
Copyright 2000-2004, G. Michaels Consulting Ltd.
Decomposer Unicode version
DBDOC Kit Version: 10.30
hyperlink running with code page 1252
```

6.6.1.3 DBDOC_SUMMARY.ERR - Composed of every file that had errors

The DBDOC_SUMMARY.ERR file is the actual error file, composed of every file that had any errors. The verbosity of DBDOC.ERR has been filtered out. Like the other summary files, it also contains the build and system statistics:

- Errors from the database utilities
- Compression Level
- Program versions/kit mismatch info
- Hardware and software configurations
- Your build source locations and types
- Management of Change summary
- Excluded file list
- Decomposer error files
- CIUMon Settings

This is the place you look for all the errors in a single file, information that can be difficult to see in the filtered error files. Sometimes the first error in a file makes sense and is the cause of others, which would otherwise seem inexplicable.

```
DBDOC Kit Version: 10.30
>>>> FROM: C:\DBDOC_BUILDS\SITE MAIN 50019D-C\ABB Australia\wizard.err -- 2011-01-04 12:00:26 PM
Config file is: C:\Program Files\GMCL\DBDOC\param.txt.
Initialization file is: ABB Australia_WIZINIT.TXT.
Param file is: C:\Program Files\GMCL\DBDOC\filetypes.txt
>>>> FROM: C:\DBDOC_BUILDS\SITE MAIN 50019D-C\ABB Australia\decomposer1\dbebp.log -- 2011-01-31 1:17:
--mdb_extractor 10.3 work $Name: r2011-01-27b $ (1.0.1.10)
--Initializing Global Objects...
--Running as dbebp
--mdb_extractor running with code page 1252
```

6.6.2 Individual errors

6.6.2.1 800xA color not found [INTERNAL 201]

```
Unable to find 800xA color SODGCyanBlueDark in color definitions
```

This message means that a call was found for a color in an 800xA graphic that DBDOC did not resolve.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL MISSING 800xA COLOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.2 Adapt block adapts previous block [CHECK 118]

```
Adapt block Module 1,01,02 Block 1001 adapts previous Module 1,01,02 Block 125
S3 (1.0)
```

This message tells you that an adapted specification gets its value from an adapt block with a higher block number.

In the case of redundant modules, there can be a significant error if the adapted specification does not have any value until the subsequent block is calculated. This is the primary reason for this message.

If the module processes blocks in block number order, the minor problem can be that the value of the specification is used before a new value is calculated, which means there is a lag in responding to a change.

Sometimes this message can be ignored, because there are ways of making the block processing not follow block numbers.

The initial value of the adapted specification is reported. If it is not 0.0, it means that a value has been deliberately set as a starting one. "value unavailable" means that the adapted block does not exist.

```
Adapt block Module 11,01,31 Block 21321 adapts previous Module 11,01,31 Block 21309 S7 (1.11)
Adapt block Module 15,57,03 Block 25437 adapts previous Module 15,57,03 Block 21994 S3 (value unavail
```

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK ADAPT PREVIOUS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.3 Adapt block does not exist [ERROR 014]

```
Adapt block Module 1,01,02 Block 1001 does not exist
```

This message means that an adapt block was expected, but does not exist.

If a block is being adapted that does not exist, some controllers (modules) will load but not go into execute mode.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

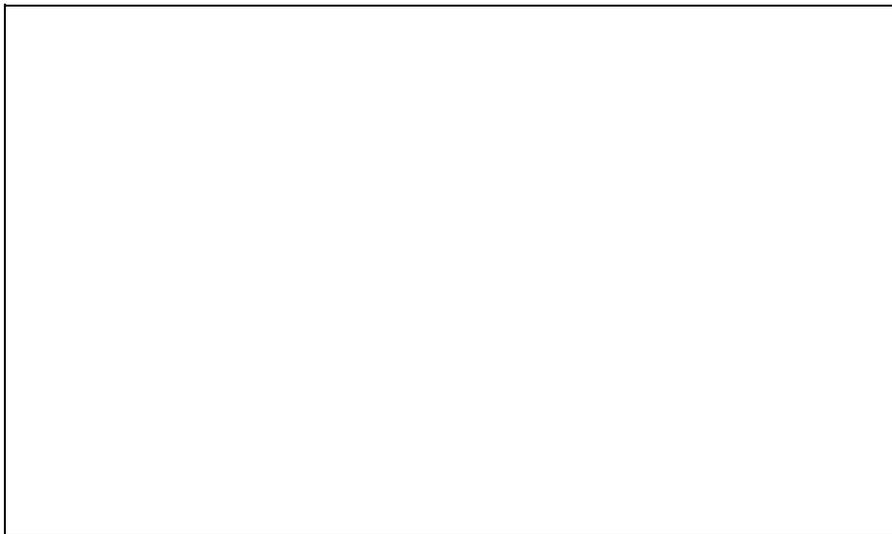
6.6.2.4 Adapt block: Adapted block does not exist [ERROR 021]

```
Adapt block Module 1,01,02 Block 1001 on drawing 1010200A.CAD: Adapted block  
1002 does not exist
```

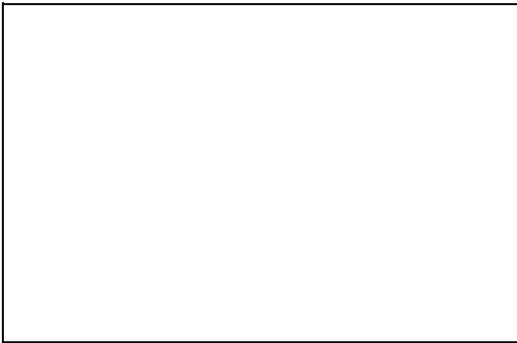
This message means that an adapt block is attempting to adapt a block that does not exist.

On-site testing has confirmed that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.

As far as we have determined, if you try a cold reload of a block with these specs, the module will not run. You can, however, load the redundant controller and fail over to it with no problem. The error should be fixed to avoid giving you problems when you do not expect them.



In the above image, the adapt block has an S2 value of 1383, however there is no block 1383 on the same module. This means that no adaptation is available. In the below image, spec 2 is a 0, so there is no block for this adapt block to adapt. Therefore this block is doing nothing.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.5 Adapt block: Not used [ERROR 019]

```
Adapt block Module 1,01,02 Block 1001 on drawing 1010200A.CAD: Not used
```

This message means that an adapt block exists, but is not used because S2 and S3 are both 0.

As far as we have determined, if you try a cold reload of a block with these specs, the module will not run. You can, however, load the redundant controller and fail over to it with no problem. The error should be fixed to avoid giving you problems when you do not expect them.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.6 Adapt block: Spec out of range [ERROR 020]

```
Adapt block Module 1,01,02 Block 1001 on drawing 1010200A.CAD: Spec 15 on Module  
1,01,02 Block 1002 out of range (FC 25)
```

This message means that an adapt block has a spec out of range.

On-site testing confirmed that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.

```
Adapt block Module 4,50,02 Block 1529 on drawing 45002P3:  
Spec 14 on block 1516 out of range, (function code 40).
```

Click on the link in the error browser to bring up the CLD/CAD sheet where we have an adapt block issue.



Double click on the hotspot hiding under Spec 2.



Zoom out and see the MSDVDR block - it is Block 1561.



This was an unusual case – DCS Dyslexia.

Many clients have made the most common adapt block error:

Logic containing an adapt block is cloned. The block numbers in the new logic are correctly modified, but spec 2 is missed. This means that it adapts the original.

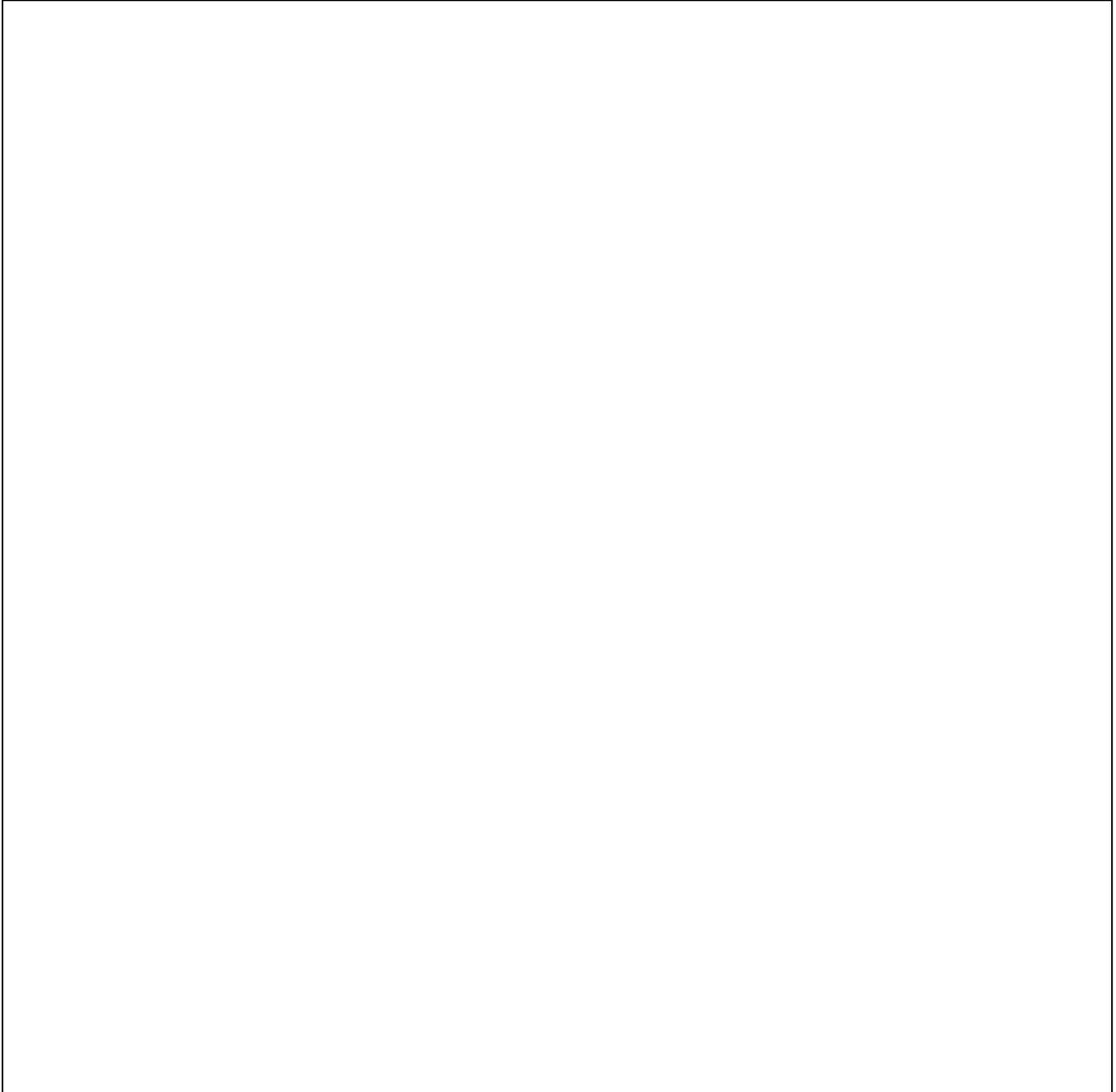
Resolving the error involves exactly the same steps:

1. Find the adapt block.

2. Find the adapted block (the target)
3. Look on the page of the adapt block for what is probably the intended target.

There are other scenarios – block number errors, changed logic, incorrect values of S3.

This drawing shows more than one, except that it is likely this is actually abandoned logic. Sheet Z9, as the last sheet, is often a boneyard sheet. Here, legal blocks have probably been used badly.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

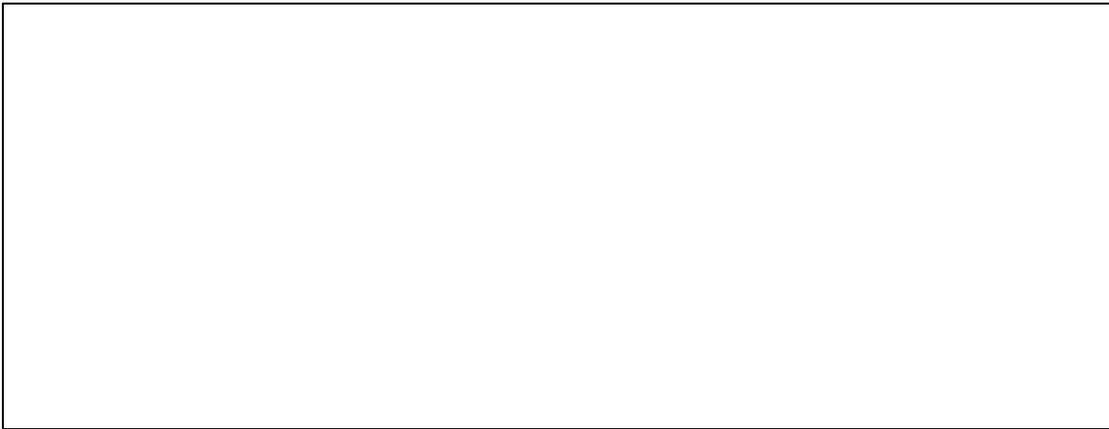
6.6.2.7 Adapt type mismatch [ERROR 169]

```
Adapt type mismatch: Module 1,01,02 Block 1001 adapts Module 1,01,02 Block 1002  
S3 (B: Binary) with Module 1,01,02 Block 1000 (Real)
```

The adapt type mismatch messages often accompany other errors, so the same block can have multiple errors. Also, adapt blocks are very flexible, so mismatches "work". However, these messages may be the only indication of an otherwise impossible to detect error.

See below. It is usually easy to figure out.

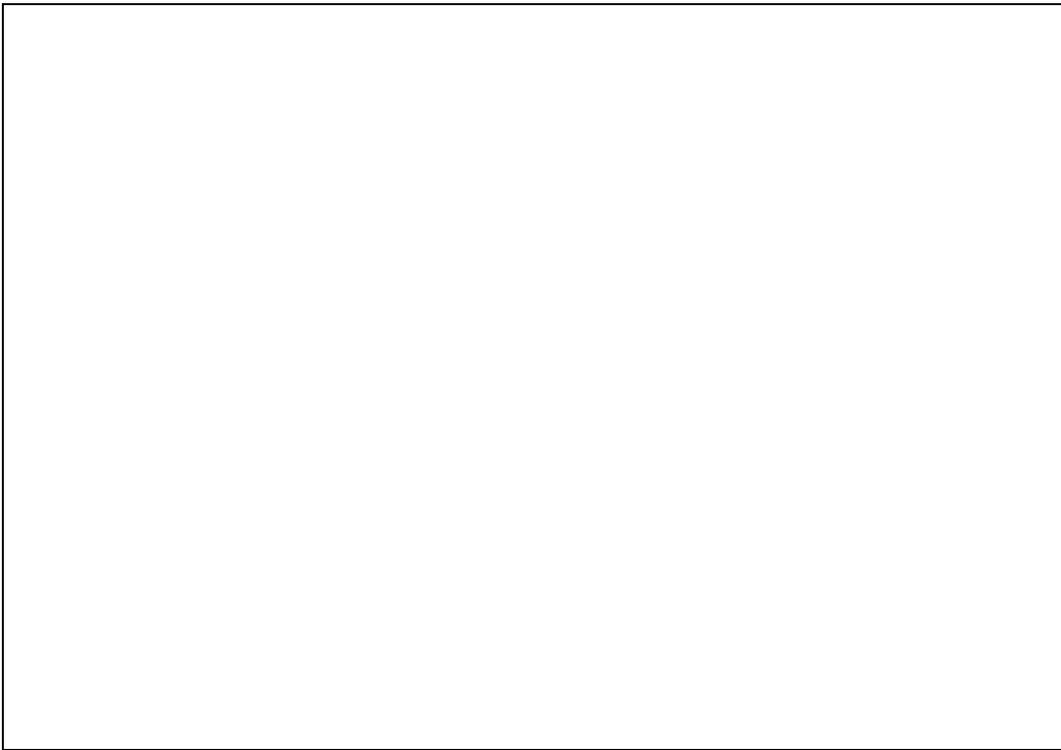
```
Module 1,11,04 block 6269 adapts block 6710 S14 - already adapted (by block 6709).
```



The normal scenario for this error is that S2 of a cloned adapt block has not been changed to match a target, even though the output block number has.



On sheets 67003B2A.CAD and 67003C2A.CAD, we will guess that there are MSDD blocks that do not have S14 adapted by an adapt block on that page. They have been orphaned.



The error here is an interesting one, not quite typical.

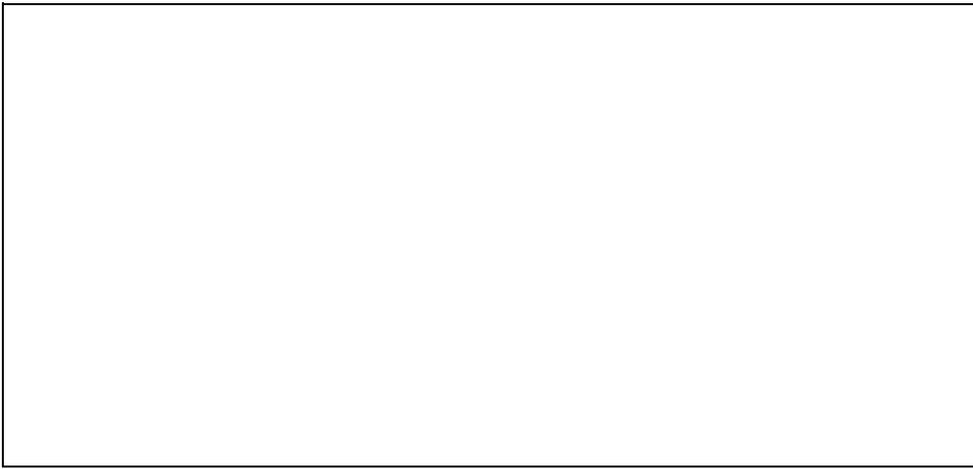
First, look at the text of the message:

```
Adapt block Module 4,50,02 Block 1529 on drawing 45002P3:  
Spec 14 on block 1516 out of range, (function code 40).
```

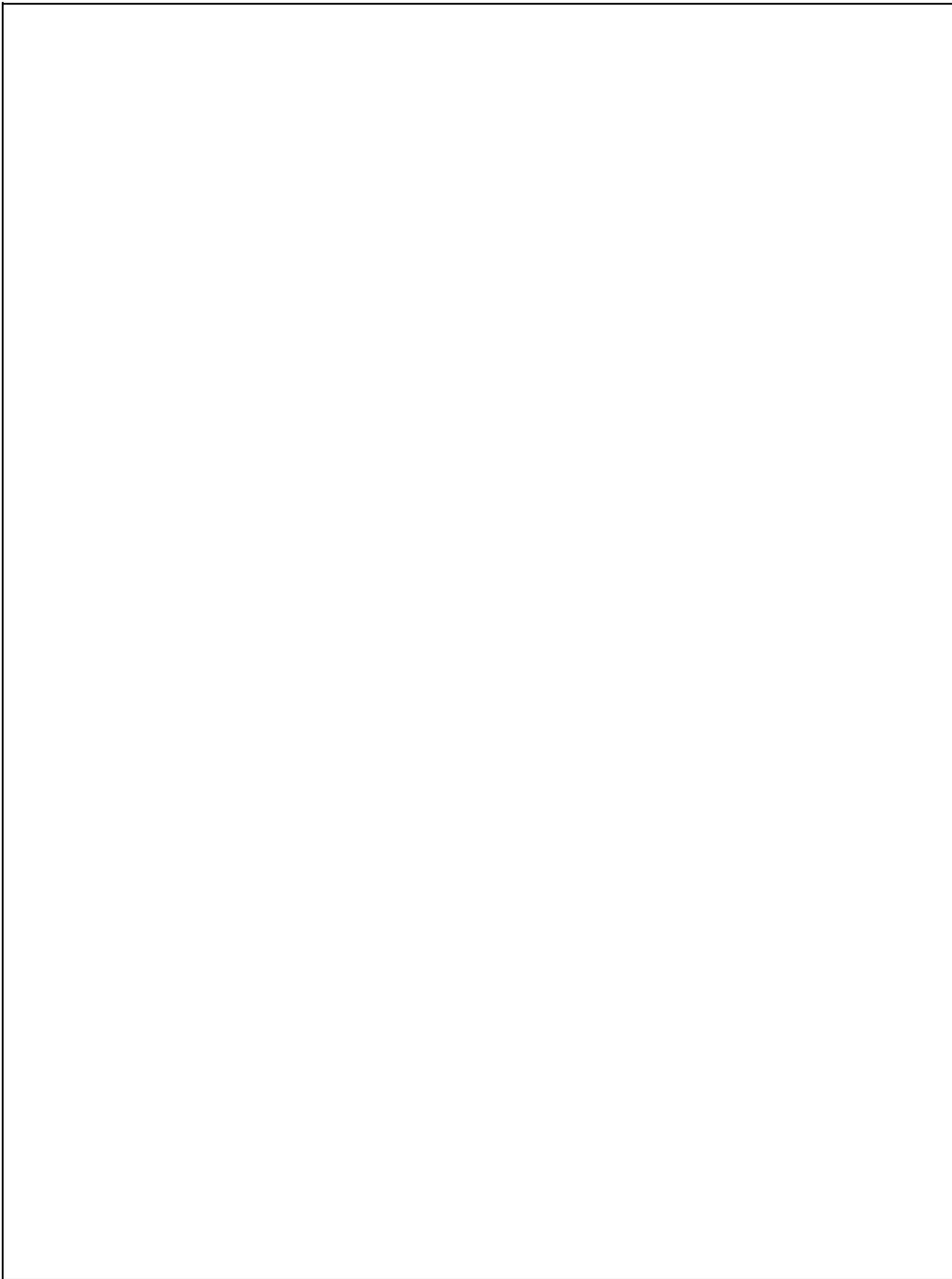
Click on the link in the error browser to bring up the CLD/CAD sheet where we have an adapt block issue.



Double click on the hotspot hiding under Spec 2.



Zoom out and see the MSDVDR block - it is Block 1561.



This was an unusual case – DCS Dyslexia.

Many clients have made the most common adapt block error:

Logic containing an adapt block is cloned. The block numbers in the new logic are correctly modified, but spec 2 is missed. This means that it adapts the original.

Resolving the error involves exactly the same steps:

1. Find the adapt block.

2. Find the adapted block (the target)
3. Look on the page of the adapt block for what is probably the intended target.

There are other scenarios – block number errors, changed logic, incorrect values of S3.

This drawing shows more than one, except that it is likely this is actually abandoned logic. Sheet Z9, as the last sheet, is often a boneyard sheet. Here, legal blocks have probably been used badly.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.8 Adapted block on other page [CHECK 044]

```
Adapted block on different page: Adapt block Module 1,01,02 Block 1001 on
1010200A.CAD modifies Module 1,01,02 Block 3001
```

This is not an error at all, but a notice requested by our clients. DBDOC eliminates any issues by linking ADAPT blocks to the specs they adapt.

We make a note of these because having the ADAPT block on the same page as the block it adapts is easier to understand.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK ADAPT PAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.9 Adapted coordinates out of order [INFO 032]

```
Adapted coordinates out of order for F(x) Module 1,01,02 Block 1001
```

You may want to check that there are base or design values for the parameters of F(X): Function Generator [FC1] blocks that are adapted. If there are, the graph will be meaningful. Also, the order of processing (adapt block before function block) will not be as important.

This is not usually a significant error. We report them for these reasons:

- They are very hard to find and can be mysterious in operation.
- By making them visible to you, you can see if the actual implementation makes sense by looking at live data
- Since DBDOC will graph the function if the X and Y coordinates make sense, this file allows you to go to the work of finding good "design" values of the specifications so that the DBDOC presentation is useful to a user
- The intent of the adapted function generator should be documented, probably right there on the CLD or CAD sheet
- There might be an error in that a block number is wrong or was reused, with unintended consequences

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO ADAPTED COORDINATE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.10 Adapted rung spec [INFO 061]

```
Adapted rung spec: Module 1,01,02 Block 1001 S5 is adapted by Module 1,01,02
Block 3001
```

This message flags places where rung block logic could be changed dynamically. The intent of this message is to flag these operations so you can verify them and keep them in view. We believe the design of the logic should be added to the CLD or CAD sheets so that people can see what is intended. (The only time we have found this message, it was a mistake caused by reusing a block number. The adapt action was not intentional, but resulted because an H//L block had become a rung block.)

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO ADAPTED RUNG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.11 AlarmVector tag has case mismatch - tag is in table [ERROR 138]

```
AlarmVector tag TagName has case mismatch - TAGNAME is in AlarmVector table
```

The AlarmVector table in Conductor NT is checked for case variants.

The case usage of the entry in the AlarmVector table in Conductor NT must be uppercase, as Unix is case-sensitive but Microsoft Access is not.

Because the uppercase tagname exists, the non-uppercase entry should be deleted (after you verify that it is calling up the same graphic as the uppercase one). It is being ignored anyway, but alarm vectoring is working because an uppercase version of the tag is in the table.

digestion.xdc

```
CT22-288A gms graphic:2500
CT22D-121A gms graphic:2700 <==
CT22E-040 gms graphic:2281
...
ZT35Q-047 gms graphic:5936
ct22d-121a gms graphic:2700 <==
di23-000 gms graphic:2282
```

To trigger block 2700, CT22D-121A, (the one in capital letters), must be present. The gms ct22d-121a, (lower case letters), is ignored. When the lower case gms is the only one present for a particular block number, the block will not trigger. If the capital and lower case instances do not trigger the same block, the upper case

one will take effect, but the lower case one will not, so the block attached to it will not trigger.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ALARMVECTOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.12 AlarmVector tag has case mismatch - tag is not in table [ERROR 139]

```
AlarmVector tag TagName has case mismatch - database tag TAGNAME is not in
AlarmVector table
```

The AlarmVector table in Conductor NT is checked for case variants.

The case usage of the entry in the AlarmVector table in Conductor NT must be uppercase, as Unix is case-sensitive but Microsoft Access is not.

Because the uppercase tagname does **not** exist in the alarm vector table, this entry should be modified to be all uppercase. Right now, it is being ignored and the desired alarm vectoring is not working.

digestion.xdc

```
CT22-288A gms graphic:2500
CT22D-121A gms graphic:2700 <==
CT22E-040 gms graphic:2281
...
ZT35Q-047 gms graphic:5936
ct22d-121a gms graphic:2700 <==
di23-000 gms graphic:2282
```

To trigger block 2700, CT22D-121A, (the one in capital letters), must be present. The gms ct22d-121a, (lower case letters), is ignored. When the lower case gms is the only one present for a particular block number, the block will not trigger. If the capital and lower case instances do not trigger the same block, the upper case one will take effect, but the lower case one will not, so the block attached to it will not trigger.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ALARMVECTOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.13 AlarmVector tag is not defined in any database associated with console [ERROR 140]

```
AlarmVector tag TagName (or any case variant) is not defined in any database
associated with console
```

The AlarmVector tag is not defined in any database associated with console.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ALARMVECTOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.14 Ambiguous wiring [ERROR 121]

```
Ambiguous wiring at 1000,1200: multiple wires at sink Module 1,01,02 Block 1001  
FC 1 input 0
```

Wiring has been found that cannot be resolved. The intent of the wiring is not clear and may be in error. DBDOC cannot make the correct connections.

When you see this, use Composer or WinCAD to look at the wiring to find what is confusing and fix it.

This message appears when DBDOC could not figure out the wiring and did not connect blocks to inputs where they are used. In the case below, we can guess that the line circled is really two separate line segments (quite possibly both digital dashed ones) and we could not figure out what to do, though it may be obvious to someone when he actually looks at it, of course.

```
Ambiguous wiring at 1120,1760: multiple wires at sink  
Ambiguous wiring at 4400,5140: multiple wires at sink Module 1,15,03 Block 23776 FC 38 S3  
Ambiguous wiring at 6110,4430: multiple wires at sink Module 7,101,04 Block 25785 FC 38 input 2 (old
```



The easiest way to find this error is by searching for its document using the Topic Title Search function. Once on the document, use the Find Coordinates function to find the ambiguous wiring.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

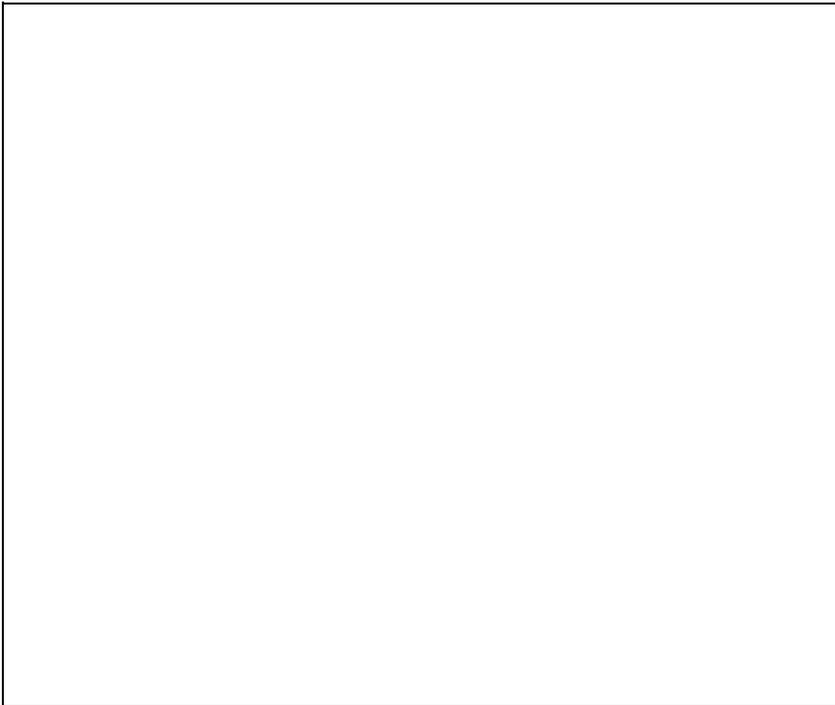
This error type was originally reported in error file **DBDOC AMBIGUOUS WIRING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.15 APID block illegal decrease flag [ERROR 086]

Advanced PID Module 1,01,02 Block 1001 using block decrease flag with S19=0

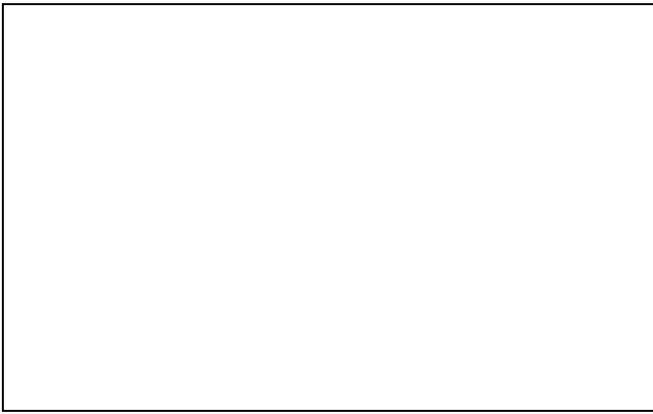
This message alerts you to a place where an APID: Advanced PID Controller [FC156] block is misused, based on ABB service bulletins. You will want to verify this block is functioning correctly.

The documentation for APID: Advanced PID Controller [FC156] says: "Note: Do not use the output block increase (or decrease) flag if using the quick saturation recovery option (S19 equals zero)."



In this example, the Block Increase/Decrease Flag Outputs are in use. The documentation indicates that you should not use the output block increase or decrease flag if the quick saturation recovery option (S19 = 0) is being used.

Advanced PID Module 4,50,02 Block 2380 using block increase flag with S19=0.



Both Block Increase BI and Block decrease BD outputs are in use, which is contrary to the documentation. S19 = 0, indicating the quick recovery form of the algorithm is being used.

It is possible to change the specifications to eliminate these messages. This should be done in close coordination with a DCS specialist. The important point about the error message is that you can look for hints that problems with outputs and reset are caused by the specs rather than by the inputs.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

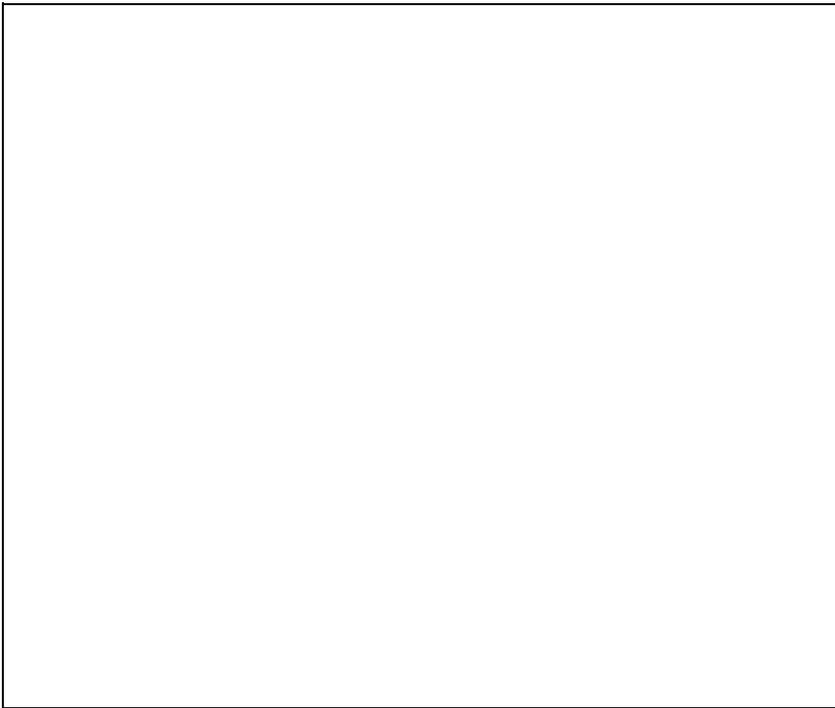
This error type was originally reported in error file **DBDOC ADVANCED PID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.16 APID block illegal increase flag [ERROR 085]

Advanced PID Module 1,01,02 Block 1001 using block increase flag with S19=0

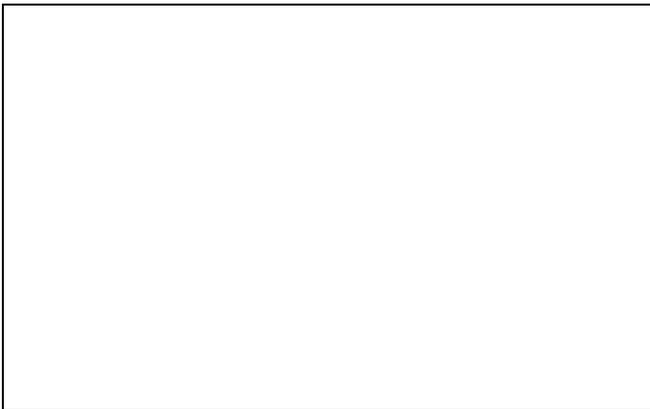
This message alerts you to a place where an APID: Advanced PID Controller [FC156] block is misused, based on ABB service bulletins. You will want to verify this block is functioning correctly.

The documentation for APID: Advanced PID Controller [FC156] says: "Note: Do not use the output block increase (or decrease) flag if using the quick saturation recovery option (S19 equals zero)."



In this example, the Block Increase/Decrease Flag Outputs are in use. The documentation indicates that you should not use the output block increase or decrease flag if the quick saturation recovery option (S19 = 0) is being used.

Advanced PID Module 4,50,02 Block 2380 using block increase flag with S19=0.



Both Block Increase BI and Block decrease BD outputs are in use, which is contrary to the documentation. S19 = 0, indicating the quick recovery form of the algorithm is being used.

It is possible to change the specifications to eliminate these messages. This should be done in close coordination with a DCS specialist. The important point about the error message is that you can look for hints that problems with outputs and reset are caused by the specs rather than by the inputs.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADVANCED PID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.17 APID old algorithm and inhibit inputs [ERROR 087]

Advanced PID Module 1,01,02 Block 1001 using old algorithm with inhibit inputs

This message alerts you to a place where an APID: Advanced PID Controller [FC156] block is misused, based on ABB service bulletins. You will want to verify this block is functioning correctly.

The listed firmware revision upgrade documents state that the original algorithms continue to exhibit bugs:

- 188390
- 121995B
- 121705Z
- 121706Z A0 - F3
- 10165

Function Code 156; New Algorithm types:

Specification S18 now accepts additional algorithm types 10, 11, 12, and 13, which are functional replacements of existing types 0, 1, 2, and 3, respectively (i.e., S18=1x executes the new replacement algorithms, S18=0x executes the original algorithms). The original algorithms were kept intact for backward compatibility with existing configurations that may rely on the current behavior. The new replacement APID algorithms correct the following problems (these problems continue to exist in the original algorithms):

1. Increase/Decrease Inhibit: When the increase or decrease inhibit is active (Specification S9 = 1, and/or S10 = 1), a rapid, unexpected change in output may occur depending on changing error signal, changing output high and/or low limits (Specification S16 and S17), and the tuning parameters.
2. RMAX PV: When transferring RMAX (9E+18) into the PV (Specification S1), the output jumps in the wrong direction. This problem is only evident at RMAX (i.e., for example, 9E+8 does not produce the problem).

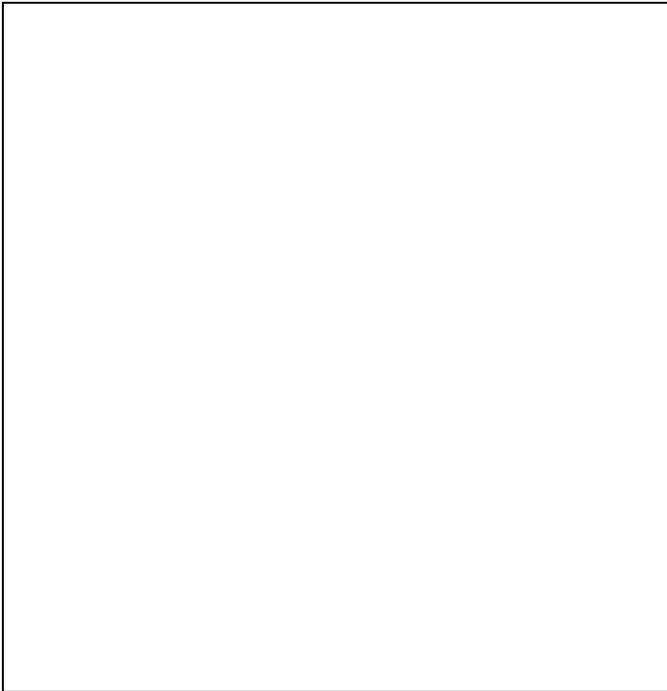
NOTE: DBDOC does NOT report a transfer of a dynamic value equal to RMAX PV. The information is included here for your reference.

NOTE: ABB provided this information to a client that will affect people with older MFP modules, MFP02s at revision F5 or earlier.

Function Code 156; Advanced PID Controller (APID): When the algorithm type specification S18 is set to run the new "C" algorithms (S18 = 1x), the block will not accept any on-line configuration changes (for any specification). Also, if S18 is changed from 1x to 0x, the block will accept the change, but the backup module will not come on-line.

Example:

Advanced PID Module 1,11,03 Block 1870 using old algorithm with inhibit inputs.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADVANCED PID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.18 Assertion failed [INTERNAL 080]

```
Assertion failed: 1 == 0, file sample.c, line 30
```

Our programs make assumptions about what they will find in data. This indicates that our compiler encountered data that violated our programming assumptions. If you get this message, please contact us so we can make our DBDOC programs handle the problem.

It would be helpful to us if you sent the file that caused the error to us, along with a screenshot, for analysis.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL ASSERTION FAILED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.19 Attempted ADAPT block redefinition [ERROR 176]

Attempted ADAPT block redefinition: Module 1,01,02 Block 1001 already adapts Module 1,01,02 Block 1002 S2, ignoring new adapt of Module 1,01,02 Block 1004 S3

Adapt block problems reported here include places where two adapt blocks adapt the same specification, and other adapt block issues. Whenever a specification is adapted by more than one block, you should verify that is the intent.

On-site testing has confirmed that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.20 Bad CAD magic number [INTERNAL 002]

Bad CAD magic number: read 0, expected 1

This message means that a CLD or CAD file was found that we cannot process due to a bad internal value we affectionately call a magic number.

Please try to open the file with your Composer or CAD system. If the file seems to be fine, please send it to us for analysis.

This error can be a symptom of a failure in the blue DBDOC dongle, or in its USB port. Simply moving the dongle to another port might make the error disappear, signalling a USB port failure. If another dongle is available, substitute it and try again. If that works, the dongle has failed and will be replaced free of charge.

You can ask GMCL for a replacement dongle at any time at no charge (except for courier cost, if you want it quickly).

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID CAD FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.21 Block adapts a block that is already adapted [ERROR 117]

```
Module 1,01,02 Block 1001 adapts Module 1,01,02 Block 1234 S3 - already adapted  
(by Module 1,01,02 Block 1002)
```

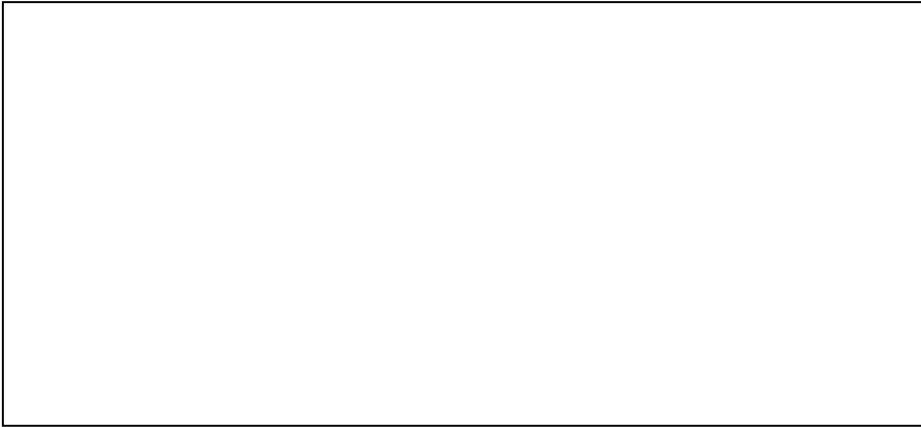
This message identifies a place where two adapt blocks adapt the same specification. Whenever a specification is adapted by more than one block, you should verify the intent.

On-site testing has confirmed that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.

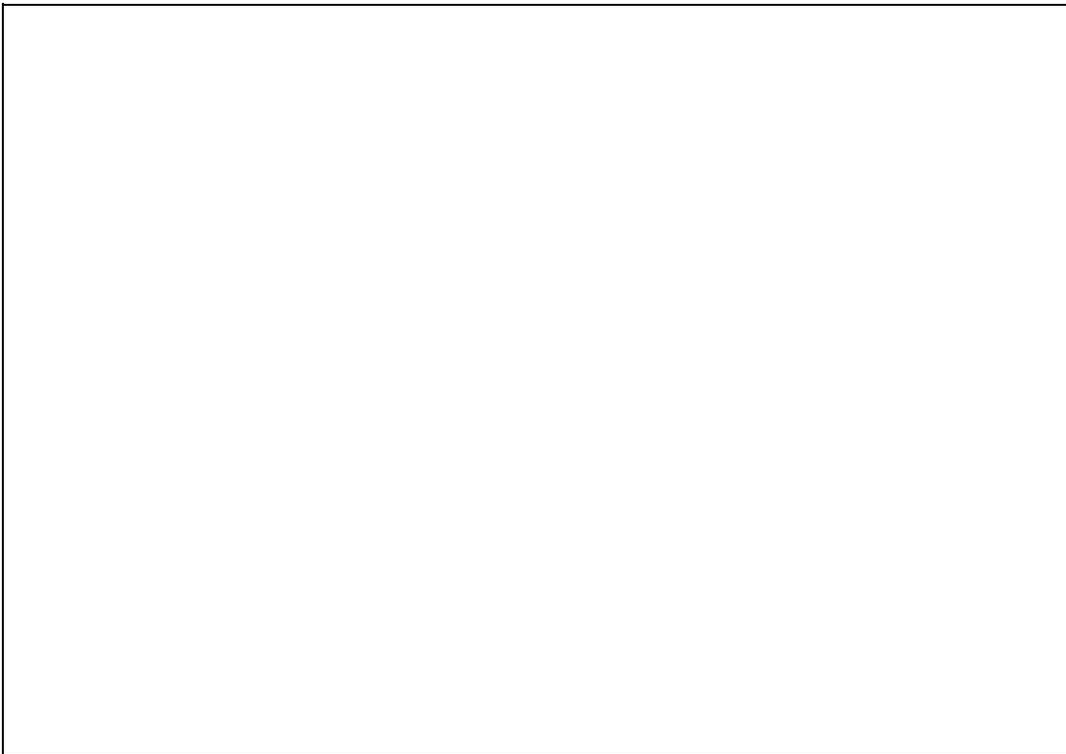
```
Module 1,11,04 block 6269 adapts block 6710 S14 - already adapted (by block 6709).
```



The normal scenario for this error is that S2 of a cloned adapt block has not been changed to match a target, even though the output block number has.



On sheets 67003B2A.CAD and 67003C2A.CAD, we will guess that there are MSDD blocks that do not have S14 adapted by an adapt block on that page. They have been orphaned.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.22 Block not wired from same function code [CHECK 186]

Block should only be wired by same function code: Module 2,41,31 Block 22100 FC 13 S25 wired from Module 2,41,31 Block 22101 FC 123

There appears to be a block connected to an input that fails to follow the documented rules for the connections.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK INPUT WIRING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.23 Block number on unwired input [COSMETIC 098]

```
No wired input: Module 1,01,02 Block 1001 FC 1 has static input from block 0  
(should be 5)
```

This message means that a block has been found that has no input attached to it. You normally can ignore these messages as a cosmetic issue.

We have seen that this condition can be caused in WinCAD and DOS CADEWS by connecting a source to a F(x) block, then deleting the line. The result is that S1 has a value 0. Composer restores the value to 5.

With some older firmware revisions, it is possible that the configuration will compile cleanly, but will not run when it is loaded into the module.



This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC NO WIRED INPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.24 Block number too high [ERROR 322]

```
Block number too high for Module 1,01,02 Block 20000. Maximum block number for  
this module is 9998.
```

This block is not in the valid block number range for its module.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised

may be in unused logic or graphics, or have other realities that make it a non-issue.

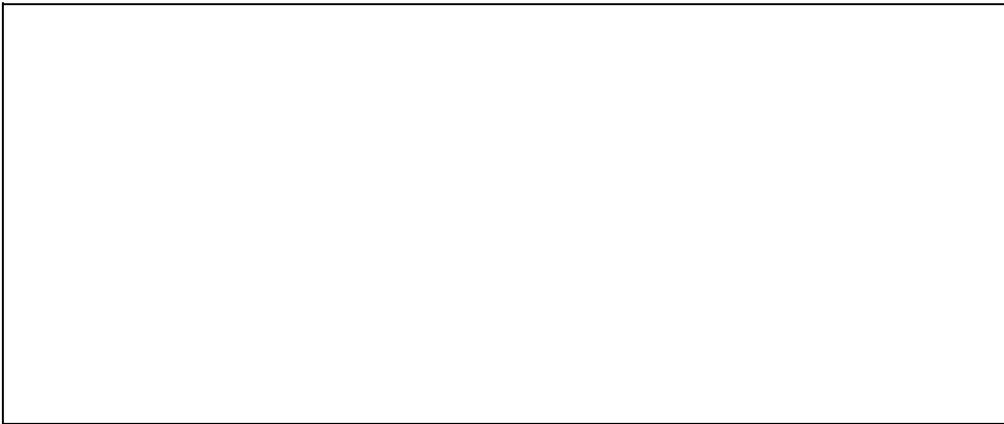
This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.25 Block out of range [ERROR 167]

```
Block 12345 out of range (0-9998) for Module 1,01,02 Block 1234 (FC 37)
```

This message means that the value of a block is smaller or larger than the allowed value range.

It is possible that a spec has been entered incorrectly.



```
Block must be between 0-9998.
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

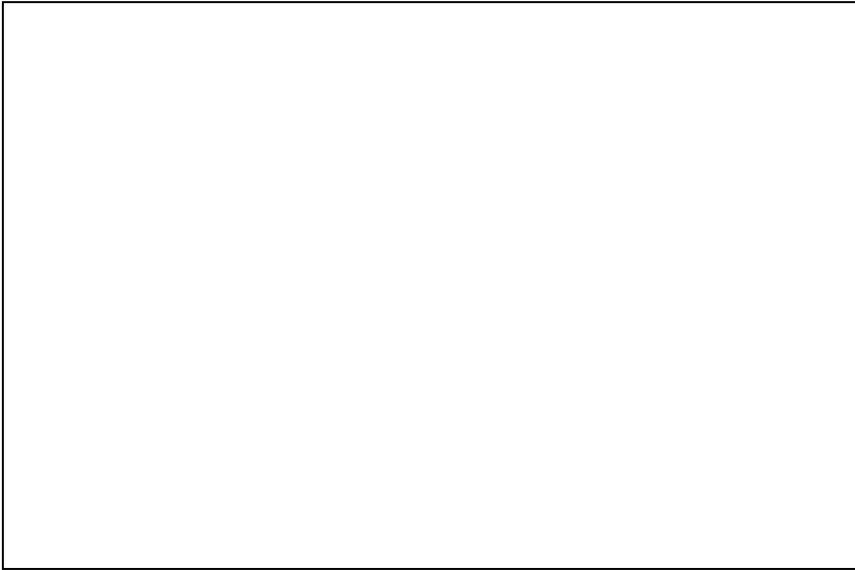
This error type was originally reported in error file **DBDOC SPEC OUT OF RANGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.26 Blocks have been overlaid [CHECK 225]

```
Blocks have been overlaid: Module 1,16,03 Block 419 (FC 62) and Module 1,16,03  
Block 423 (FC 62) overlaid at 2500,1640
```

This messages tells you that a block or reference has been placed on top of another.

There are several blocks of the same value overlapping. If one is deleted and Hyperview was refreshed, then there would be another directly underneath.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK OVERLAID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.27 Bogus source for block zero [BUILD 180]

`Bogus source for block zero on 2330500A.CAD`

This message means that the CLD/CAD sheet contains a block that puts an output block on block 0, which is illegal.

This can be caused by using function code blocks in diagrams. For example, dropping NOT blocks on a picture looked good in one system, but caused many useless messages.

```
--Generating index: Module 2,01,02 Block Index  
Bogus source for block zero on 199030A.CAD
```



This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD BOGUS BLOCK ZERO.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.28 Both 0 and 1 have same descriptor [CHECK 064]

Both 0 and 1 values have the same descriptor "MANUAL" for HTAS142 Module 1

This message identifies digital tags that have the same text descriptor for both 0 and 1 values.

This message is not generated if both text descriptors are blank.

Sometimes this message is the result of files locked in processing threads. In those cases you can ignore this message.

Example:

```
Both 0 and 1 values have the same descriptor "MANUAL" for ANLG101
Module 1,01,02 Block 6694 VAPOR TURBINE SOAK ENGAGED
```

Because of the duplicate descriptors, Hyperview will not display which is actually *MANUAL*.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK DUPLICATE DESCRIPTOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.29 Broken connection [ERROR 345]

```
Broken connection from v26M1030Ctrl_Dil to variable reference
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.30 Building multiple 800xA consoles into one project file not yet supported [INTERNAL 200]

```
GRAPHIC: Building multiple 800xA consoles into one project file not yet
supported in BCS_StationM72363AD348D045 ({7FD44C07-2CBE-47EE-8AE4-6E33889B5FC2})
```

This message means that references were found that appear to refer to multiple 800xA consoles, a feature that DBDOC does not yet support.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL MULTIPLE 800XA CONSOLES.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.31 CADNames cannot be opened [ERROR 10044]

```
CADNames cannot be opened: Failed to Load Existing CADNames
```

This error may indicate an issue with sibling projects.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **Decomposer.err**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.32 Can't find ProgID for GUID-named dependency [INTERNAL 203]

```
Can't find ProgID for GUID-named dependency  
{41FBEBF11-A4CD-44A5-8DD0-BEC7A7371838} in INFI90HarmonDA49A02A21F94  
({4995961D-59A9-40DF-AD79-CDEF5DD9544C})
```

This message means that a reference could not be resolved. This can occur in systems with 800xA graphics.

Contact us for assistance resolving this message.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNRESOLVED REFERENCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.33 Can't open DBF file RESOLVE.DBF [BUILD 10001]

```
Can't open DBF file RESOLVE.DBF: The dbf file was not found
```

This message means the file could not be opened for the reason given.

Sometimes this message may be resolved if the project is rebuilt. If a rebuild does not correct the issue, please check the file with the appropriate tools and send the file to us if it appears to be valid.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD FILE OPEN FAILURE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.34 Cannot determine target [CHECK 347]

```
Cannot determine target, too many potential matches at 0,0
```

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.35 Cannot import error database [INTERNAL 289]

```
Cannot import Error Database DECOMPOSER1\error.db
```

Hyperlink was unable to use the error information from another build system component, such as decomposer.

This may be caused by a version mismatch, or an older copy of an error.db file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL ERROR DATABASE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.36 Choosing description for XRefObjID [CHECK 10029]

```
Choosing description XRef Description for XRefObjID 45
```

This message covers situations where multiple descriptions exist for the same cross-reference element in a Composer project. This is cosmetic in the sense that Composer links source output and the destination input without comment.

For people, however, this situation is a problem. The message means that Automation Architect is capable of linking the two descriptions shown, and does so. If you look at printed sheets, you could not guess this.

Cleaning these out involves finding the two entries in the XRefObjID table in the project.EBP file and deleting one of them.

Just like Composer, DBDOC cannot tell which one to use. We have chosen one description to work with, so we are consistent. The user must decide which is the correct one.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

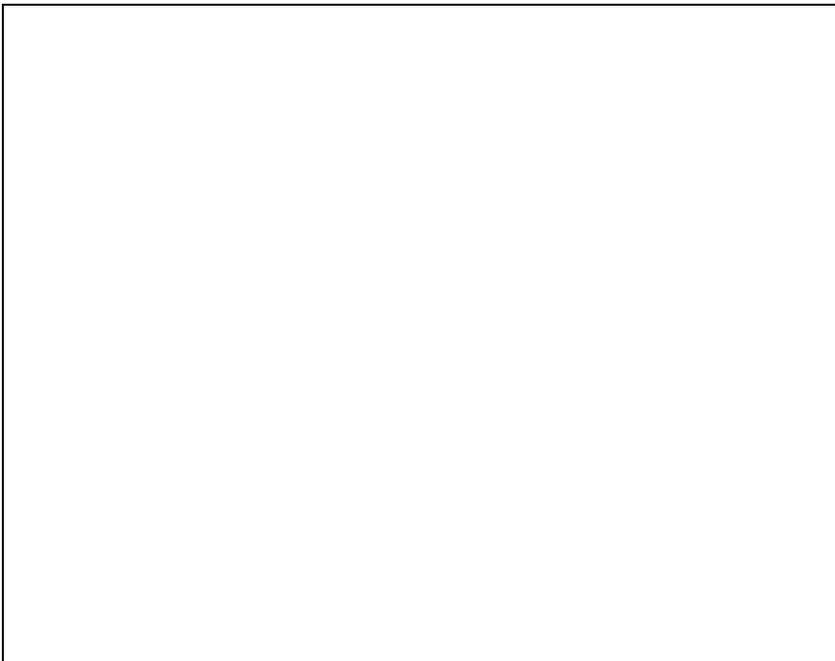
This error type was originally reported in error file **DBDOC-CHECK XREFOBJID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.37 Circular inhibit [CHECK 181]

```
Circular inhibit: tags 1FW/YS0020A and 1FW/YS004 inhibit each other
```

Circular alarm inhibits have been detected. Alarm inhibit action may be unpredictable.

In the database for a graphics system, two tags inhibit each other. That is, if one goes into alarm, it is supposed to inhibit alarms on the other. The actual effect is not necessarily predictable and probably cannot be counted on. Unreliable operation of alarm inhibits is not desirable for the operators.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK CIRCULAR INHIBIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.38 Circular reference [CHECK 184]

```
Circular reference: IREF (GEN GROSS WATTS)
```

This message means that the CLD/CAD sheet contains an IREF that feeds an OREF of the same name. This is legal, but misleading.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK CIRCULAR REFERENCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.39 Circular symbol inclusion [CHECK 182]

```
Circular symbol inclusion F
```

A symbol is including itself as a symbol. DBDOC does not understand how this works and thus interprets it as an error.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK CIRCULAR SYMBOL INCLUSION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.40 Circular symbol inclusion of tag [CHECK 183]

```
Circular symbol inclusion TagF included by G
```

A symbol includes itself. DBDOC interprets this as an error.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK CIRCULAR SYMBOL INCLUSION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.41 CLD file has not been recompiled. [INFO 10022]

Uncompiled CLD file EBY008B.CLD (11102M0) Thu Sep 30 14:37:18 2010 has not been recompiled in Module 1,11,02 file EBY4593.CFG Fri Sep 03 15:11:42 2010

The CLD file indicated was changed after the module was compiled. The Composer system has not compiled and validated the work.

For some people, common practice is to save after making a manual switch change, so these uncompiled CLD files might not indicate that actual changes were made.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO CLD FILES NOT COMPILED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.42 Completely unconnected wire [ERROR 340]

Completely unconnected wire from (4128, 1280) to (8288, 3712)

This message indicates a wire is defined with both of its ends unconnected.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.43 Connection to unconfigured variable [ERROR 342]

Connection for IW1.40.13.1 to unconfigured variable C06_Vibration.x35LSH8490_IO

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.44 Connection wired in Composer only [INTERNAL 170]

Wired in Composer only: Module 1,01,02 Block 1001 wired to Module 1,01,02 Block 1005 S5

We have found a situation in the Composer wiring tables we cannot reconcile. The message with "Wired in Composer only" can show a place where DBDOC did not make a connection.

Examples we have found so far involve lines that have a connector on the end of an input arrow, or that have an output directly connected to an input with no line connecting them.

```
Composer wiring problem: references with downstream connections never got a block source
Wired in drawing only: Module 50,155,02 Block 20050 wired to Module 50,155,02 Block 20051 S1
Wired in Composer only: Module 2,05,02 Block 20130 wired to Module 2,05,02 Block 20089 S2
Reference TRIP CONTACT?: 3
```

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL COMPOSER WIRING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.45 Connection wired in drawing only [INTERNAL 171]

Wired in drawing only: Module 1,01,02 Block 1001 wired to Module 1,01,02 Block 1005 S5

We have found a situation in the Composer wiring tables we cannot reconcile.

Example:



Note that DBDOC has drawn the missing connection in red. Examples we have found so far involve lines that have a connector on the end of an input arrow, or that have an output directly connected to an input with no line connecting them.

```
Composer wiring problem: references with downstream connections never got a block source
Wired in drawing only: Module 50,155,02 Block 20050 wired to Module 50,155,02 Block 20051 S1
Wired in Composer only: Module 2,05,02 Block 20130 wired to Module 2,05,02 Block 20089 S2
Reference TRIP CONTACT?: 3
```

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL COMPOSER WIRING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.46 Continuous exception reports generated [ERROR 145]

```
Continuous exception reports generated by Module 10,05,02 Block 3900 (FC 30):  
Significant change value is 0.0.
```

The block shown will generate an exception report on any change of value. This may be intentional, but can also be an error.

If a value comes from some external source and is rarely changed, this technique assures that the change will be immediately deployed to graphics and other import blocks. After that, the static input means that no extraneous exception reports will occur.

On the other hand, if the input is a dynamic analog value, it may well change every time it is read. This means that an exception report will typically be generated every second, which is why this is brought to your attention.

These messages are only generated when exception reports occur. Exception reports actually do not occur from all exception report blocks. They are generated only on two conditions:

- the block is enrolled in a database in a CIU serving a console or historian
- the block is imported by an exception report importing block

Example:

```
--Scanning drawing A050292: AS-SHIPPED  
Continuous exception reports generated by Module 10,05,02 Block 3900 (FC 30): Significant change val
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC CONTINUOUS EXCEPTION REPORTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.47 Control character in tag description [INFO 323]

```
Unexpected control character 0x0A (line feed) in description of tag TAGNAME in
database tags.csv
```

A control character, also known as a non-printing character, was found in a tag description.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.48 Could not locate batch include file [BUILD 252]

```
Could not locate batch include file: Data.TXT
```

The include file for your Batch file can't be found due to how the project is specified.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD BATCH NO INCLUDE FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.49 Couldn't open font file [BUILD 192]

```
Couldn't open font definition file Test Font -- using defaults
```

This message means that an AutoCAD or MicroStation file uses a font definition file that could not be opened.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

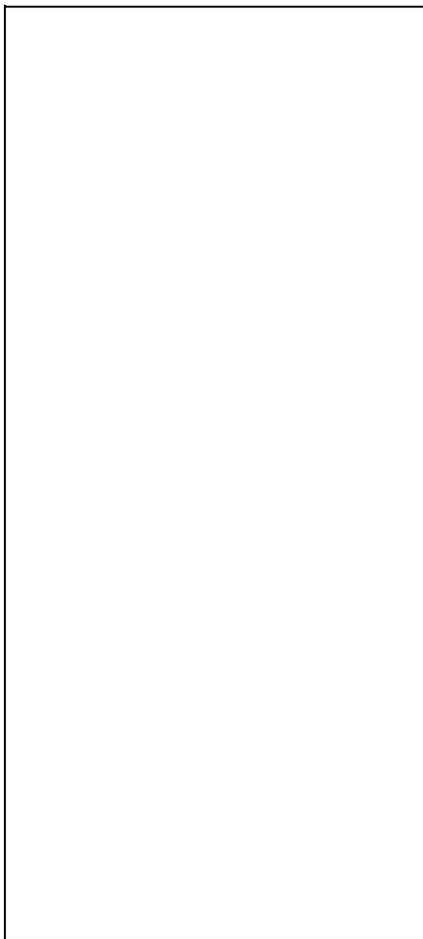
This error type was originally reported in error file **DBDOC-BUILD FONT PROBLEM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.50 DAANG alarm control not an integer [ERROR 090]

DAANG block Module 11,03,04 Block 427 S20 value is truncated to zero (check for mis-configuration)

This message means that S20 on the DAANG block has a real value where an integer is expected. Alarming may not be working correctly.

If the value of S20 is something other than one of the values associated with a bit, or a sum of several bits, the alarm will not work



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

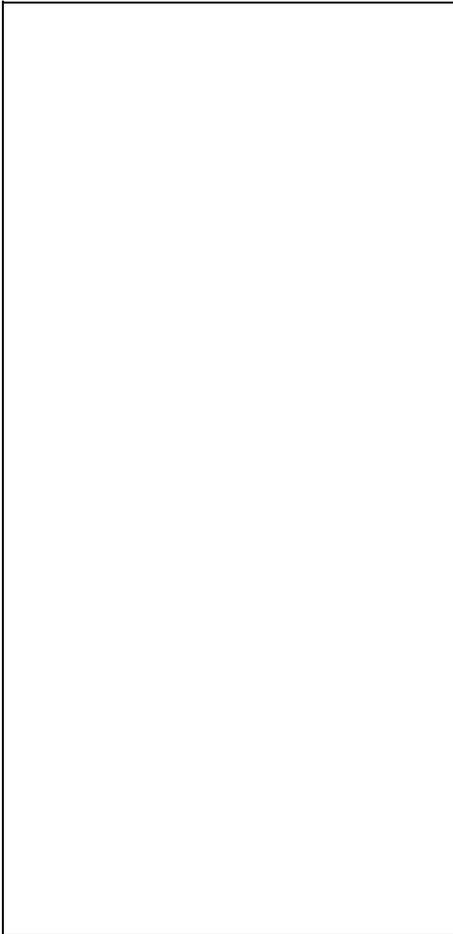
This error type was originally reported in error file **DBDOC DAANG SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.51 DAANG both de-alarm and return alarm [ERROR 089]

DAANG block Module 11,03,04 Block 427 S20 has both de-alarm and return alarm set, return alarm overrides

This message means that one aspect of alarming has been overridden. Alarming may not be configured correctly.

When the value of S20 is a sum of both 32 and 64, (among other things), then both bit 5 and bit 6 are active. Bit 5, de-alarm enabled, and bit 6, return alarm enabled, cannot co-exist. Therefore, when both 32 and 64 are in the S20 value, bit 6 overrides bit 5.



In this case, the value of S20 is 104.0, the sum of 64, 32, and 8, (bits, 6, 5, and 3).

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC DAANG SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.52 DAANG illegal alarm control [ERROR 088]

DAANG block Module 11,03,04 Block 427 S20 value out of range (0-1023): 2048

This message means that the DAANG block S20 value is smaller or larger than the allowed value range.

It is possible that a spec has been entered incorrectly.



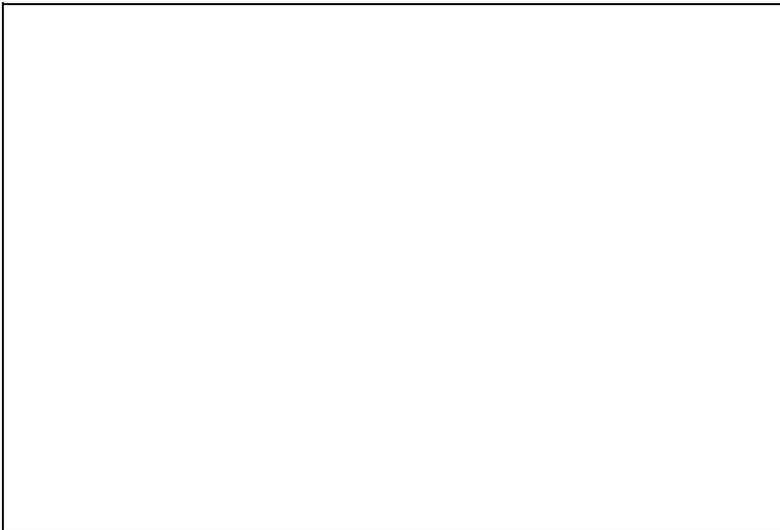
This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SPEC OUT OF RANGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.53 DAANG suspicious alarm period [ERROR 091]

DAANG block Module 11,03,04 Block 427 S31 value suspicious (please verify period for time-based alarms)

This message means that time based alarming may not be configured correctly.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC DAANG SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

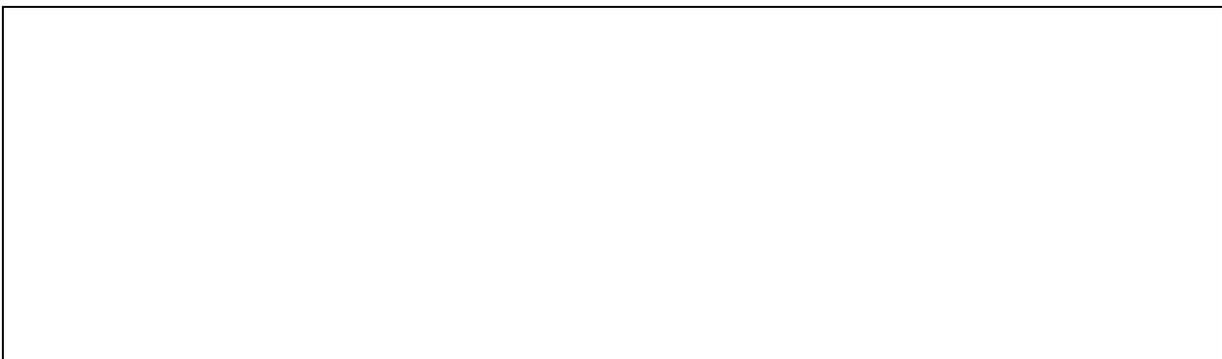
6.6.2.54 Data type mismatch [ERROR 094]

Data type mismatch: Module 1,01,02 Block 1001 (FC 25) S1 expects type Real but is wired from Module 1,01,02 Block 3456 type Digital

This message says that we found a block where the source and import block types do not match. These might not work and could identify an error.

Example:

Data type mismatch: Module 1,99,11 Block 222 (FC 30) S1 expects type Real but is wired from Module 1,



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL IMPORT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.55 Database has no records [ERROR 290]

```
Database cannot be opened: FILE1234.G
```

A database has been included in the build that has no records. Either there never were any, or a problem has occurred with the current version of a database file.

The error message is found in the following context:

```
--Scanning database file MASTERTG.csv Database cannot be opened: F:\...\CTS 01\TAG\MASTERTG.csv  
ST:Database Statistics: ST:Name : MASTERTG.csv ST:Type : Tag ST:Records : 0 ST:Fields : 0 Database is  
empty
```

If the database normally works, the extraction of this instance would appear to have failed. If the extraction is from a .XLS or .XLSX file, it means that there were no records extracted that we could process.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-BUILD FILE OPEN FAILURE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

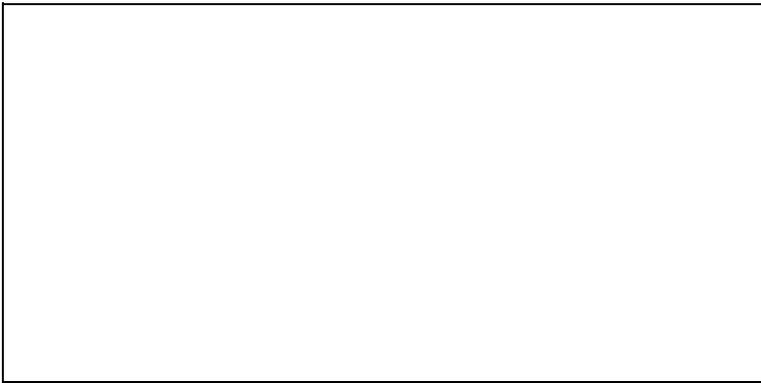
6.6.2.56 Descriptor empty [CHECK 015]

```
Output reference descriptor empty at 1000,1000
```

The input or output reference mentioned has no text in it. This suggests that the module will not compile properly and is probably not fully operational.

Examples:

```
Input reference descriptor empty at 1760,1845  
Output reference descriptor empty at 1760,2025
```



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK EMPTY REFERENCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.57 Digital in Alarm Group cannot alarm [CHECK 078]

Non-alarms digital Module 1,01,02 Block 1000 is in Alarm Group 15 in database DATABASE1

These messages show digital tags that have S2 = 2 in their DO/L block, but do not have Alarm Group set to 0. They cannot alarm, but it might be the case that they are intended to alarm.

Note that usage in some sites is to set new tags to Group 99 as a flag that they have not been completely resolved.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK NON-ALARMING DIGITAL.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.58 DIOFCSpec associated with unknown FC [INFO 10014]

DIOFCSpec associated with unknown FC 209. Please contact GMCL

DIOFCSpec associated with unknown FC. Please contact GMCL

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **Decomposer.err**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.59 Discarding description for XRefObjID [CHECK 10030]

```
Discarding description XRef Description for XRefObjID 45
```

This message covers situations where multiple descriptions exist for the same cross-reference element in a Composer project. This is cosmetic in the sense that Composer links source output and the destination input without comment.

For people, however, this situation is a problem. The message means that Automation Architect is capable of linking the two descriptions shown, and does so. If you look at printed sheets, you could not guess this.

Cleaning these out involves finding the two entries in the XRefObjID table in the project.EBP file and deleting one of them.

Just like Composer, DBDOC cannot tell which one to use. We have chosen one description to work with, so we are consistent. The user must decide which is the correct one.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK XREFOBJID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.60 Display not found [CHECK 197]

```
Display ALMSUMF2 not found (not configured)
```

The graphic calls for another graphic which was not found. This is not really an error, but is a weakness for the operators using the graphic system. In Hyperview, go to "MISSING GRAPHICS / Index of Unconfigured Graphics" in the "System Information" Chapter to see where these graphics are referenced.

The usual sources of this error are:

1. Failure to build all graphics in the DBDOC document. If the graphic is not built, the DBDOC process will not find it. This is normally the case when there are a lot of messages in this file.
2. Graphics becoming obsolete and being eliminated, without correction of other graphics that call for the obsolete graphic.
3. Graphics being planned and anticipated that have not yet been built or installed.
4. Errors in pop-ups.

When there are a lot of these errors showing up, it normally means that you did not get all your graphics built or you are dealing with graphics that are not used. It suggests that operator graphic keys and touchpoints may be "dead" because those graphics are not in the system.

You can find where the graphic is linked using either the Audit Window or the Index of Unconfigured Graphics in the System Information chapter.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.61 Display with GUID not found [CHECK 198]

```
Display {9B314B0F}_INH:Faceplate (GUID {9B314B0F}_INH:Faceplate) not found (not configured)
```

The 800xA AFW (PPA) graphic calls for another graphic which was not found. This is not really an error, but is a weakness for the operators using the graphic system. In Hyperview, go to "MISSING GRAPHICS / Index of Unconfigured Graphics" in the "System Information" Chapter to see where these graphics are referenced.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.62 Displayed tag has no source - module not built [ERROR 266]

```
No source: Module 1,01,02 Block 1001 (TAGNAME, TAGNAME2) used in display - module not built
```

This message means that the tags indicated are used on graphics, but the source modules for them were not built from the CLD/CAD files.

This message is caused by:

1. The block comes from an external source like PI or FDI that you have not defined to be external.
2. The block comes from a module that was not built.
3. Some element of the loop, PCU or module (and block), or a combination of these, defined for the tag is incorrect.

You should make sure that the blocks exist first. In the third case above, it is often the case that the error is in the definition of the tag and the block, in fact, does exist. Typing errors can be the cause of these messages. Also, you should check groups of blocks to make sure that the rules that define blocks that are external are being followed.

Note also that you can mark the blocks as being available for monitoring only in case 3. In cases 1 and 2, DBDOC cannot get a value for the block, which is the same as Composer and WinTools, because the source is not actually a module within a real PCU.

If there is no external source for these tags, the operator graphics cannot display a correct value for them.

Example:

```
No source: Module 4,01,02 Block 133 (06-TC-5304) used in display - module not built
No source: Module 4,01,02 Block 1001 (B12BREAKER) used in display - module not built
No source: Module 4,01,02 Block 1002 (890STEAM) used in display - module not built
No source: Module 4,01,02 Block 1003 (B1BREAKER) used in display - module not built
No source: Module 4,01,02 Block 1004 (BBUSLESS) used in display - module not built
No source: Module 4,01,02 Block 1005 (B11BREAKER) used in display - module not built
No source: Module 4,01,02 Block 1006 (RCMLOADSHED) used in display - module not built
```

Here we can see that the first block is well off on its own, meaning it may be an orphan, and a potential error. The big section of consecutive blocks is probably correct, but should be checked anyway.

In BuildPlus, use **Tools/Project Options/Define Blocks with External Sources** to define blocks that do not exist to eliminate the messages if they are valid references. The dialog allows you to input the Loop, PCU, Module and Block Range info for external blocks that actually exist.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG DISPLAYED MODULE NOT BUILT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.63 Document defined in another chapter [INFO 188]

```
Document x already defined in chapter y
```

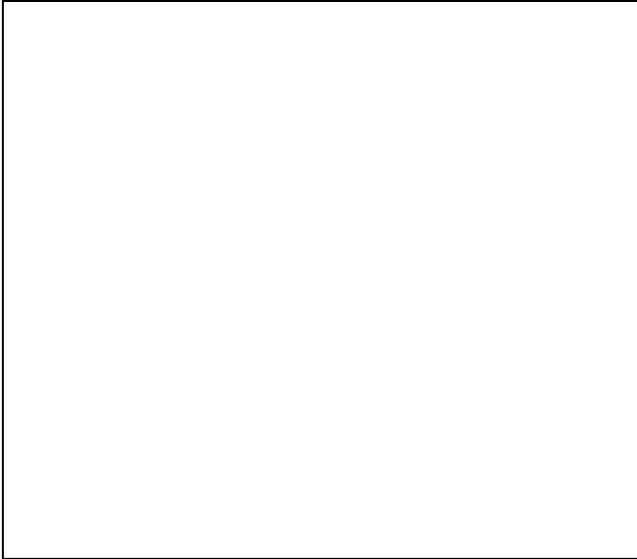
This message means that DBDOC is building the same file in twice from the same source. This can result from building files both as standard SODG Symbols and as user symbols in the case of SODG graphics, or as both models and submodels in Conductor NT, Operate IT and Process Portal B graphics.

Rebuild the project including the file only once.

Examples:

```
--Parsing configuration file C:\DBDOC_BUILDS\SITE MAIN G-P\LMS\LMS.cfg
```

```
Document CAB01.mhd already defined in chapter Module 3,14,00.  
Document CAB0120A.CAD already defined in chapter Module 3,14,00.  
Document CAB0121B.CAD already defined in chapter Module 3,14,00.  
Document CAB0122B.CAD already defined in chapter Module 3,14,00.  
Document CAB0123B.CAD already defined in chapter Module 3,14,00.
```



We can see in this example that Module 3,14,00 was added twice.

```
-Parsing configuration file C:\DBDOC_BUILDS\SITE MAIN D-F\Domtar\Domtar.cfg  
Document XREF.IDX already defined in chapter SODGSYM Symbols.  
Document ANALOG1.DY already defined in chapter SODGSYM Symbols.  
Document ANCB00L1.DY already defined in chapter SODGSYM Symbols
```

In this example, a large number of SODG symbols have been built twice.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO FILE SELECTION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.64 doVector called with non-string second argument [INTERNAL 244]

```
doVector called with non-string second argument
```

The file contains submodels that we do not understand enough. Please send us information to help us add the support needed.

The numeric second argument appears to indicate that a Conductor NT technique was used in a Process Portal B graphic without being converted to call the graphic by name instead of number.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL STRANGE ARGUMENTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.65 doVector() called with numeric 2nd argument [INTERNAL 243]

```
doVector() called with numeric second argument 0 - vectoring link may not have been converted to use name
```

The file contains submodels that we do not understand enough. Please send us information to help us add the support needed.

The numeric second argument appears to indicate that a Conductor NT technique was used in a Process Portal B graphic without being converted to call the graphic by name instead of number.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL STRANGE ARGUMENTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

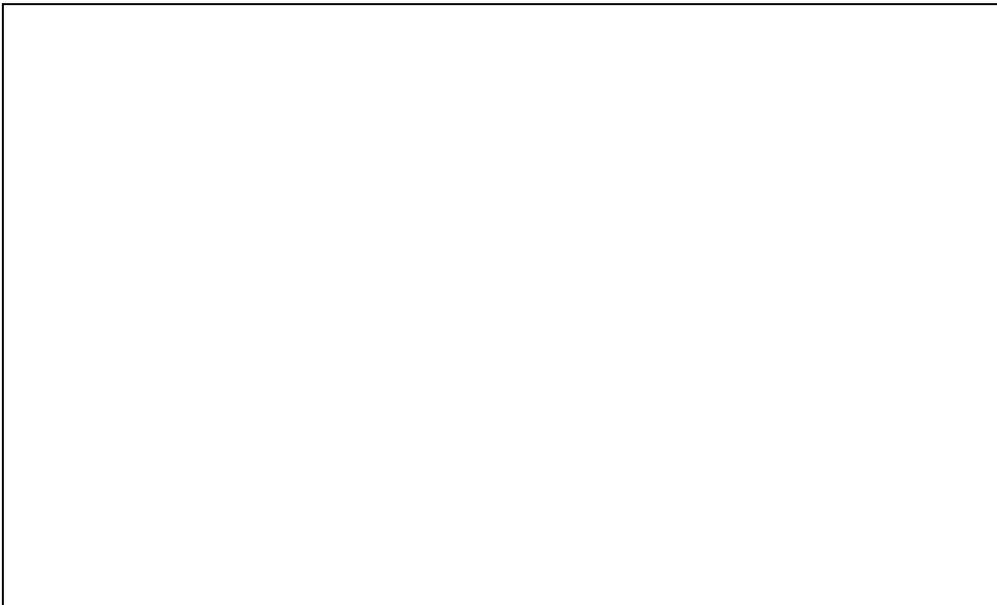
6.6.2.66 Duplicate global OREF [CHECK 190]

```
Duplicate global OREF 'BNR 5 A/R TAGOUT': Module 1,25,10 Block 3952, Module 1,26,10 Block 3952
```

The output references listed here are duplicated. Composer will rename the duplicate references to (Copy1...), (Copy2...), etc. when combining into a single Composer project. You may wish to change the reference first.

If you plan to convert this project to a single Composer project in the future, you will have to resolve each of these.

If you are building multiple Composer projects, this message will show you references that will need to be changed to allow you to merge the projects.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK DUPLICATE GLOBAL OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.67 Duplicate local OREF [CHECK 296]

```
Duplicate local OREF 'ABC-123'
```

There are two OREF definitions with the same text in the same module. This is an error.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK DUPLICATE GLOBAL OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.68 Duplicate source [ERROR 092]

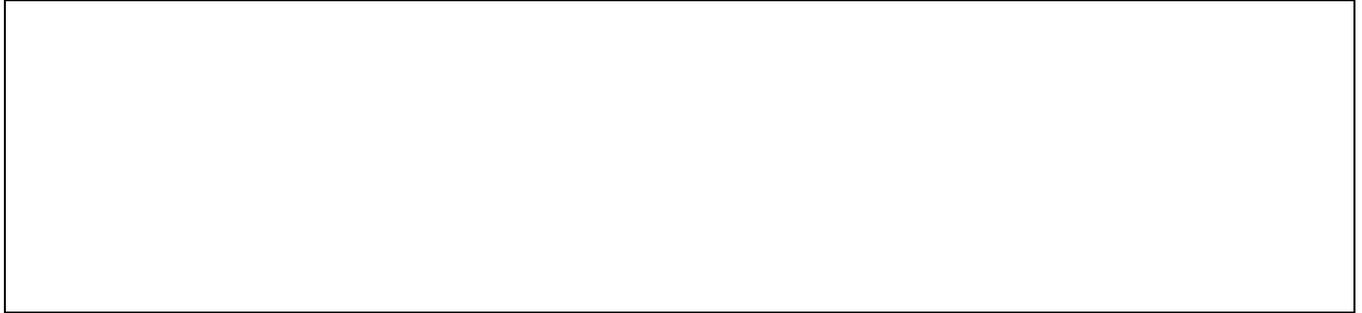
```
Duplicate source for Module 1,01,02 Block 1001 on 1010200A.CAD
```

A block appears with this message if it appears to have more than one source. This is a rare message that points to a bad build or to bad CLD/CAD sheets. The message means that more than one block exists in the module with the block number indicated.

Sometimes this creeps in when someone uses a spare block and forgets to take it off the "boneyard" page.

We do not know what happens in this situation. It is certainly possible that it is generally harmless.

```
Duplicate source for Module 1,13,13 Block 160 on 11313 CIS07 14D04 TU only
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC DUPLICATE SOURCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.69 Duplicate station tag [INFO 189]

```
Duplicate station tag abc for am91v904
```

This message means that a station tag (tagname@PV, for example) has been found in a second database where it has a different block number.

This usually suggests that inappropriate databases have been built together.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO DUPLICATE STATION TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.70 Duplicate tag in database [ERROR 103]

```
Duplicate tag TAGNAME in database: 'PMTag.csv' - duplicate ignored
```

This message means that during the scanning of the various databases a duplicate tagname was found. DBDOC depends on a unique tagname corresponding to a unique block number in a module. Any duplicate entries for that tagname are ignored.

If you get this error, you will need to use your ABB database tools to resolve it. Ask for our help if you wish. You presumably will be unpleasantly surprised when you find additional instances of the same tagname.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.71 Dynamic definition undefined [INTERNAL 277]

```
Possible dynamic definition VAR15 undefined in FileName.G at 1234
```

A possible dynamic reference was encountered but the associated dynamic is undefined.

This error can be triggered by a problem in the data or by an Unknown Access Code Error (273)

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.72 EbVector references unknown graphic [CHECK 216]

```
EbVector references unknown graphic COMMON_WATERRI_AnnunciatorPB
```

A link on the PPB graphic has been specified to a display that was not built. Either the file itself was not included in the DBDOC build or the graphic has not been updated. In Hyperview, go to "MISSING GRAPHICS / Index of Unconfigured Graphics" in the "System Information" chapter to see where these graphics are referenced.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.73 Element outside configured bounds [INTERNAL 202]

```
Drawing element Block (Type) is outside configured bounds (left=300, top=1500),  
ignoring
```

This message means that an 800xA drawing element was found positioned outside the boundaries of the graphic. Both DBDOC and the graphics processor for 800xA will not necessarily handle out-of-bounds elements properly.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL ELEMENT OUTSIDE BOUNDS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.74 Enhanced trend has illegal S2 value [ERROR 024]

Enhanced Trend: Module 1,01,02 Block 1001 has illegal S2 value 20

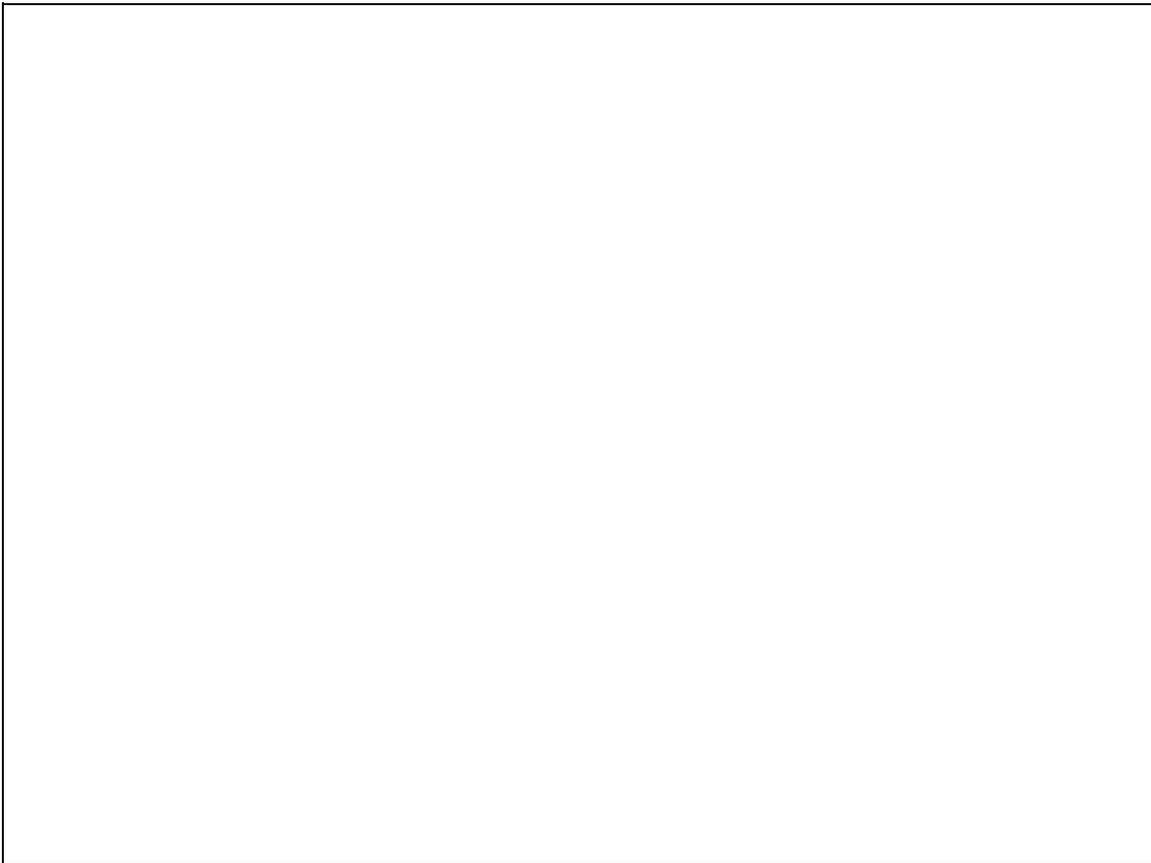
The indicated enhanced input is not a supported type. The documentation indicates that this is a configuration error.

From the documentation for ETREND: Enhanced Trend [FC179]:

S2 (Input block type) Identifies type of input to be trended.

- 0 = function code 30, analog exception report
- 1 = function code 45, digital exception report
- 2 = function code 62, remote control memory
- 3 = function code 68, remote manual set constant
- 4 = function code 80, control station
- 5 = function code 123, device driver
- 6 = function code 129, multi state device driver
- 7 = function code 136, remote motor control
- 8 = function code 177, data acquisition analog
- 9 = function code 211, data acquisition digital

Enhanced Trend: Module 1,01,02 Block 1136 has illegal S2 value 50



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ENHANCED TREND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.75 Enhanced trend: S1 wired from wrong function code [ERROR 025]

Enhanced Trend: Module 1,01,02 Block 1001 S1 is wired from FC 30 Module 1,01,02 Block 1002 but S2 1 specifies FC 45

The indicated enhanced input is not a supported type. The documentation indicates that this is a configuration error.

From the documentation for ETREND: Enhanced Trend [FC179]:

S2 (Input block type) Identifies type of input to be trended.

- 0 = function code 30, analog exception report
- 1 = function code 45, digital exception report
- 2 = function code 62, remote control memory
- 3 = function code 68, remote manual set constant
- 4 = function code 80, control station

- 5 = function code 123, device driver
- 6 = function code 129, multi state device driver
- 7 = function code 136, remote motor control
- 8 = function code 177, data acquisition analog
- 9 = function code 211, data acquisition digital

The block addressed in S1 must match the input block type in S2 or a configuration error will result.

```
Enhanced Trend: Module 1,01,02 Block 1136 S1 is wired from FC 68 Module 1,01,02 Block 5131  
but S2 0 specifies FC 30
```



In this case, S1 is coming from the wrong type of block or S2 should be 3.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ENHANCED TREND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.76 Error count exceeded [BUILD 60000]

```
Error count for Error #123 (CError) exceeded
```

The build contains a very large number (>300) of the error message specified. Because of this, not all of them are shown in the error browser.

(We believe no one would walk through the whole list, but they are listed in DBDOC_SUMMARY.ERR and DBDOC.ERR.)

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD ERRORS LIMITED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.77 External output reference is unreachable [ERROR 10015]

```
External output reference is unreachable for IREF REFERENCE on Module 1,01,02  
Block 1001: cannot import from 1,2
```

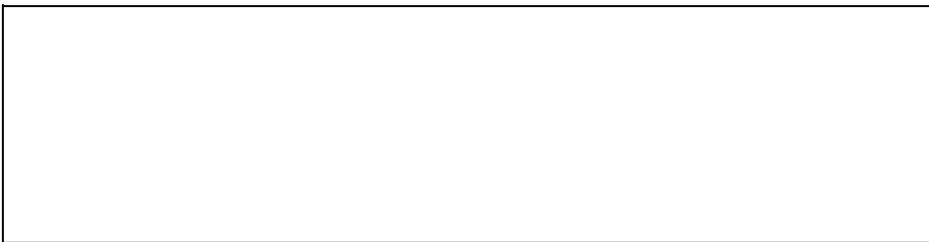
An IREF is used in the project that has an invalid output reference source.

This message will be generated if text on an import block refers to a global reference that exists, but that cannot be imported by the import function code.

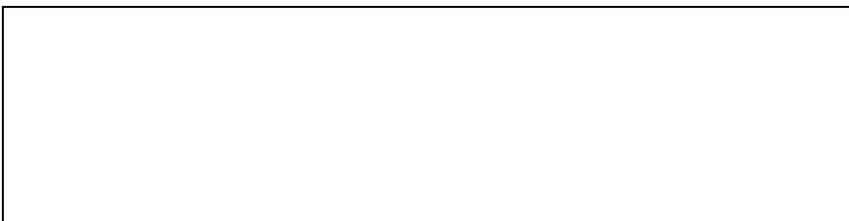
The significance is that only the specs on the block are being used to make the connection.

This message tells you that an IREF (Input Reference) exists in the Composer project. However, the function block cannot import it, because it is outside the scope of what it can import.

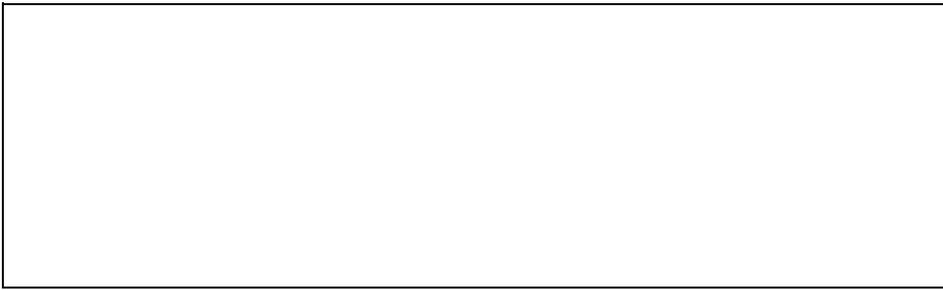
```
--Parsing file H:\...\EBY45EB.CLD... CAD Names: 2300251A.CAD  
External output reference is unreachable for IREF **1AOIK CB CLSD on Module 2,30,02 Block 1652:  
cannot import from Loop 2 PCU 20.
```



Here is the actual source, as determined by the specifications.



Here is the source that the input reference indicates.



--Parsing file H:\...\EBY3DAC.CLD... CAD Names: 2320454A.CAD
External output reference is unreachable for IREF L30D_MSG on Module

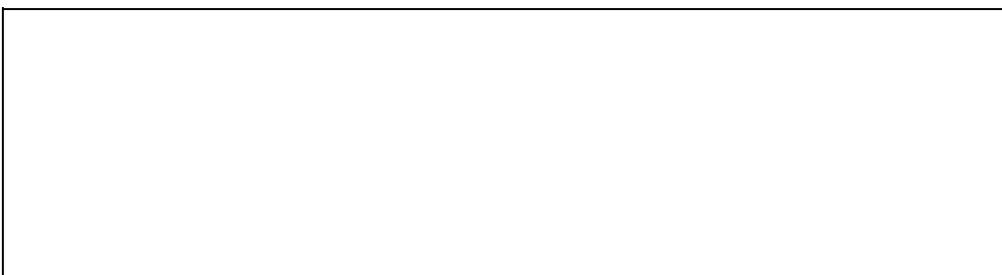
2,32,04 Block 1584: cannot import from Loop 2 PCU 22."



Here is the block that is being imported.



Here is the source indicated by the input reference L30D_MSG.



It is possible to see how Composer, in being robust enough to import files from WinCAD, also allows errors to

persist in the input references.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

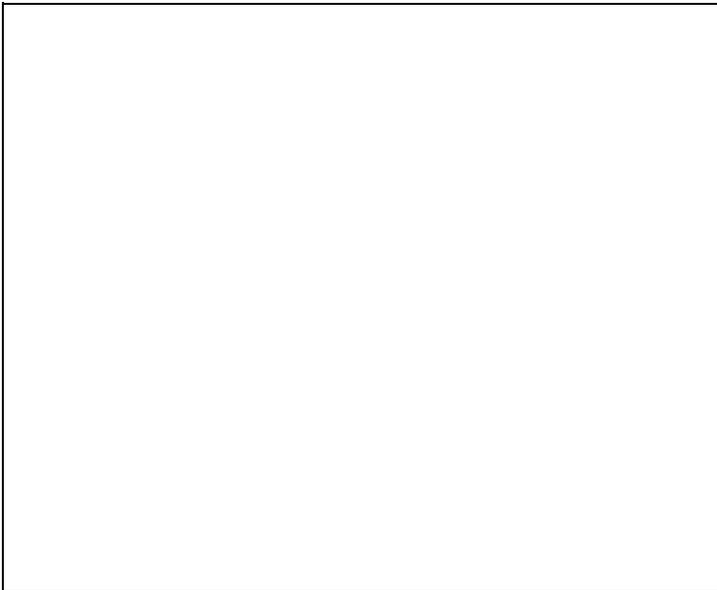
This error type was originally reported in error file **DBDOC UNREACHABLE OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.78 External output reference not found [CHECK 10008]

```
External output reference does not exist for IREF REFERENCE on Module 1,01,02  
Block 1001
```

An IREF is used in the project that had no matching External OREF. This can arise in a conversion from WinCAD, for example, when the input reference text on an AI/I reference is the same as that on OREF that distributes its value through the module.

The significance is that only the specifications on the block are being used to make the connection.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING EXTERNAL OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.79 F(x) discontinuity (step) [CHECK 031]

```
F(x) discontinuity (step) in Module 1,01,02 Block 1001 at S2 X value of 1.000
```

This message means that there is a step discontinuity in an F(x) function. This can lead to unexpected results if it is not intentional.

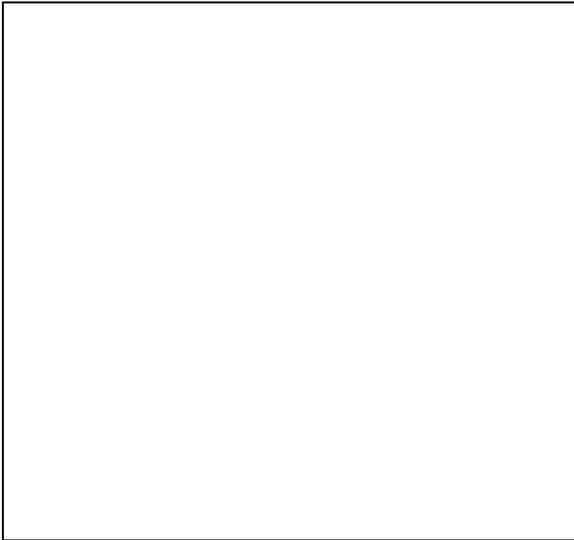
The function generator can legitimately give discontinuous results. Sometimes, as in the case of "dead bands" this is intentional. When the input is in a particular range, the output is set to an arbitrary value (often 0.0) because the sensor simply is not workable at that range.

In physical processes, discontinuities are rare. In most F(x) blocks with discontinuities, there is actually an error, often a result of typographical mistakes.

Example:

```
F(x) discontinuity (step) in Module 1,16,06 Block 4216 at S8 X value of 460.000.  
( S2, S3)          439.0          0.0  
( S4, S5)          440.0          150.0  
( S6, S7)          460.0          150.0  
( S8, S9)          460.0          143.0  
(S10,S11)          480.0          143.0  
(S12,S13)          480.0          143.0
```

The F(x) graph looks like this:



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK DISCONTINUOUS FUNCTION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

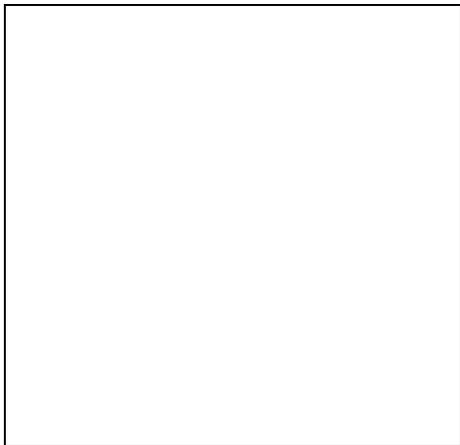
6.6.2.80 F(x) graph is flat [INFO 035]

```
F(x) Module 1,01,02 Block 1001 graph is flat - value 1.0
```

The function entered gives the same output for all inputs. This may result from adding the capability to do non-linear adjustment of inputs that ended up not needing adjustment. There may also be examples where the function has a very low non-zero slope (for example related to monitoring equipment life). Flat functions identified by DBDOC with this message have a slope of less than 1E-5.

Example:

```
--Post Processing drawing 11002B9A.CAD  
F(x) Module 1,01,02 Block 1320 graph is flat - value 1.00.
```



Notice that the Y value is 1.0 for all values of the input X value – that is, a constant.

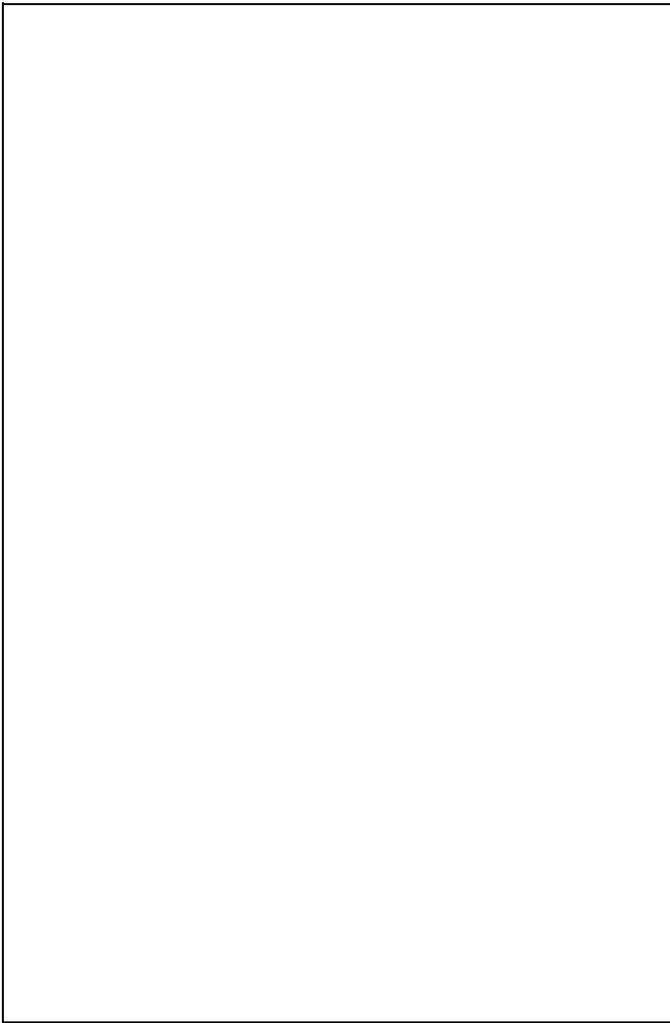
This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO FLAT FUNCTION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.81 F(x) graph is flat (unwired) [COSMETIC 036]

```
F(x) Module 1,01,02 Block 1001 graph is flat but unwired - value 7.5E-005
```

The function entered gives the same output (constant value stated) for all input values. This may not be what you want. Since the block is unwired, it has no affect on the process. Whether or not it should be wired is another question.



Since the y value for all values x is constant, (in the case of the above diagram, 8.8601), the output of this block will be, under any conditions, the constant.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC UNWIRED FLAT FUNCTION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.82 F(x) uses large values and may not clamp output [ERROR 034]

F(x) Module 1,01,02 Block 1001 uses large spec values: output value may be unexpectedly extreme

This message indicates the F(X): Function Generator [FC1] uses large specification values in some of the Y entries (i.e. S3, S5, S7, S9, S11, and/or S13). These values are used and the result is often not what people expect.

Possible problems:

- Output is not clamped (held at a constant value for increasing input values) as is often expected
- The mathematically calculated function generates an output value that can be wrong

Example:

```
--Scanning drawing ABC.CAD
F(x) Module 1,01,31 Block 22150 uses large spec values:
output value may be unexpectedly extreme
```

Check that the function is not intended to clamp the output, for example, as this message usually means that the output approaches the input value as the inputs get larger. This may not be what you want.

This message is generated when Y is not equal to 0 and the absolute value of any X or Y is $\geq 4.0E06$.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-CHECK LARGE SPECS USED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.83 Failed to locate field indices -- not a database? [BUILD 220]

```
Failed to locate field indices -- not a tag database?
```

This message indicates that the type of database processing that has been chosen is inappropriate in some way. Each type of special database processing depends on certain fields being found in the database. If they are not, the intended linking cannot work.

Please check the database processing you have chosen. If it seems correct, send us the generated error.zip and a zipped copy of the database giving these messages. We regularly add to the things we support.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD DATABASE FIELD.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.84 Failed to locate field name [BUILD 221]

```
Failed to locate field name TWOTEXT
```

This message indicates that the type of database processing that has been chosen is inappropriate in some way. Each type of special database processing depends on certain fields being found in the database. If they are not, the intended linking cannot work.

Please check the database processing you have chosen. If it seems correct, send us the generated error.zip and a zipped copy of the database giving these messages. We regularly add to the things we support.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD DATABASE FIELD.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.85 Failed to open file [BUILD 006]

```
Failed to open file sample.txt: 2
```

This message means a file could not be opened for the reason given.

Sometimes this message may be resolved by rebuilding the project. If a rebuild does not correct the issue, please check the file with the appropriate tools and send the file to us if it appears to be valid.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD FILE OPEN FAILURE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.86 Failed to read header [BUILD 001]

```
Failed to read header
```

DBDOC failed to read the header. We should have the problem file sent to us to be checked.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD FILE OPEN FAILURE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.87 Failed to read library directory [INTERNAL 013]

```
Failed to read library SAMPLE directory record
```

DBDOC failed to read a library directory record.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.88 Failed to read record [INTERNAL 004]

```
Failed to read in record
```

DBDOC failed to read a record.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.89 Failed to reseek to start [INTERNAL 005]

```
Failed to reseek to start of file
```

DBDOC failed to reseek to the start of the file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.90 FC126 S3 Exceeds 4 Linked Blocks [ERROR 318]

```
Module 1,27,02 Block 1196 FC126 if S2 = 1, S3 cannot link more than 4 Blocks
```

For function code 126 block chains, if Conversion mode spec S2 = 1, or 2, linked block spec S3 cannot chain more than 4 blocks together (including the starting block).

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. **Most errors can now be viewed in the Hyperview Error Browser.**

6.6.2.91 FC225 S5 must be 1 for FC228 [ERROR 311]

FC225 Module 1,27,02 Block 1196 is part of a chain controlled by an FC228 FDD block, so S5(=0) must be 1

This FC225 IOC/DOUT is in an FDD chain (FC228). S5 **must** be 1 for exception reporting to work with older firmware.

The Harmony Function Code Manual says:

S5

(Readback enable) The output channels can have optional readback hardware present. This specification must match the hardware configuration with respect to whether this option is enabled.

0 = disable readback

1 = enable readback

This option must always be enabled for foreign device definition (function code 228) channels.

PDP800, CI850, and CI800: Not used

From the client informing us of this problem:

Basically, exception reporting doesn't work. So, if you were historizing or using the tags in a console (graphics) you won't be able to get the actual state changes.

Further investigation indicates that the problem affects only the value displayed or fed into subsequent logic. Also, the problem is eliminated (as is the need for S5=1) if the BRC firmware is level M5 or above and the Gateway (S800 IO FC 227) is at least G5. If this is the case, there is no problem. ABB Case ABB20161007E0252 covers the details. It is not known if ABB clients were directly informed of this problem.

If the firmware revisions are not as indicated, the problem can exist.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC CHAINED FC PROBLEM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.92 Fewer fields than substitutions [BUILD 222]

Fewer fields than substitutions: 0<1

This message indicates that the type of database processing that has been chosen is inappropriate in some way. Each type of special database processing depends on certain fields being found in the database. If they are not, the intended linking cannot work.

Please check the database processing you have chosen. If it seems correct, send us the generated error.zip and a zipped copy of the database giving these messages. We regularly add to the things we support.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD DATABASE FIELD.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.93 File contains a space in its name [COSMETIC 213]

```
File D:\GRAPHICS\BLOWERS_OVERVIEW_REV3 .M1 contains a space in its name
```

This message means that the file contains a space in its name. M1 or M2 type files will not work if they have spaces in their names.

Graphx systems – Conductor NT and Process Portal B – do not accept spaces in the filenames, thus these files will not work. The usual situation is that a second file has been created with no spaces in the filename, making this file obsolete.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC SPACE IN FILENAME.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.94 File version unknown [INTERNAL 271]

```
Tokenization Failed: File Version Unknown in ABC12345
```

The version of the PGP/S+ file is not supported.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.95 Font file not found [BUILD 191]

```
Font file Test Font not found
```

This message means that an AutoCAD or MicroStation file uses a font that has not been found in the set of fonts given to the DBDOC build.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD FONT PROBLEM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.96 Frozen import block [ERROR 093]

```
Duplicate off-module reference (AI/I, AI/I) to Module 1,01,02 Block 1002:  
1020205A.CAD and 1020208A.CAD
```

This message indicates that the same exception report value is being imported into a module twice.

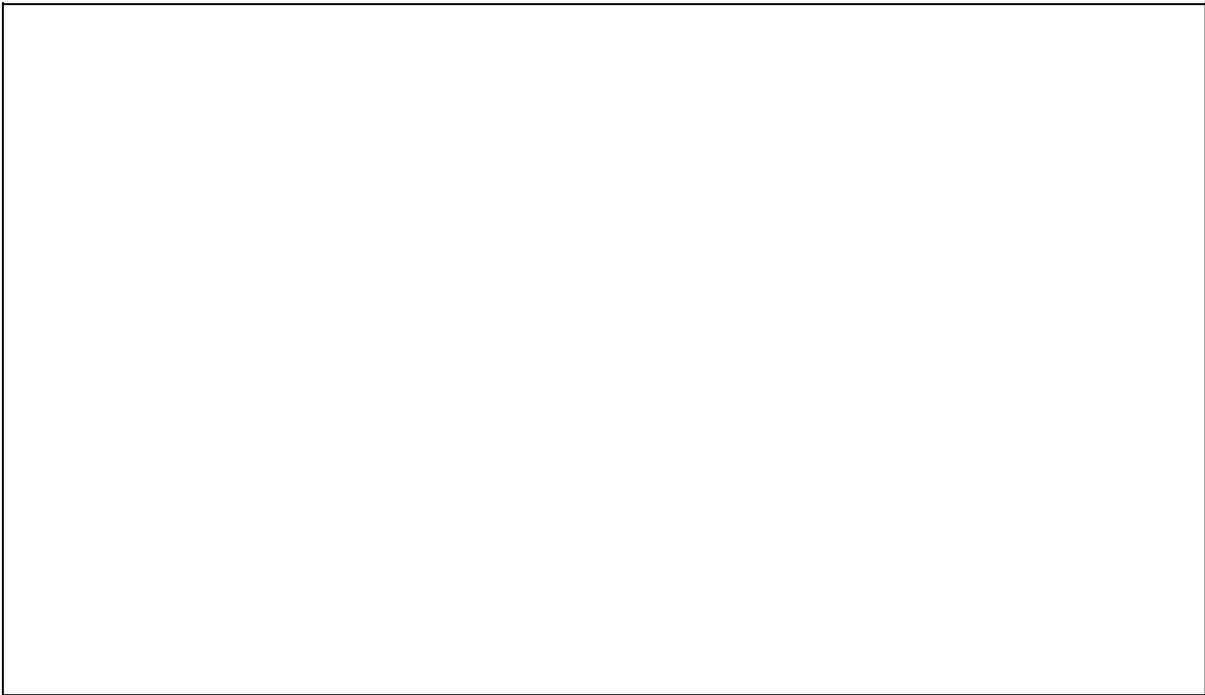
This is a serious problem if the import block is in use, because one of the values is always frozen. When you switch to the redundant module, as you do for a configuration change, the one that was working freezes and the other begins working.

In the case of analog blocks, the value will 'flat-line'. In the case of digital inputs, the imported value will simply fail to respond to a change in the source. This can be catastrophic.

You can find these in Hyperview by searching for the block then clicking on the exception report import blocks that are in the same module.

Example 1 (both import blocks used): DBDOC identifies these situations, since one of the import blocks is always frozen.

```
Duplicate off-module reference (DI/I, DI/I) to Module 2,05,02 Block 2713:  
20202Y8: Composer Initial Revision and 20202Z2: Composer Initial Revision
```

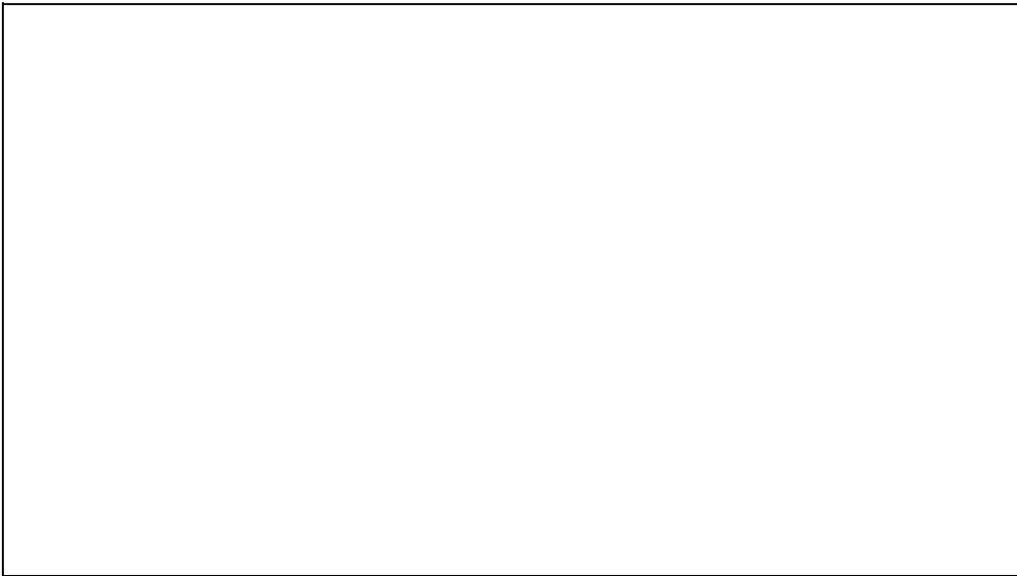


Example 2 (one import block used, one not used): DBDOC identifies these situation, since there is a 50-50 chance that the unwired block will have live data and the wired one will be frozen.

Duplicate off-module reference (DI/I, DI/I) to Module 7,27,02 Block 35: 20102Y3A.CAD and 20102ZZA.CAD



Example 3 (both import blocks not used): There is no DBDOC error message for this situation, because there is no harm possible.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC FROZEN EXCEPTION REPORT INPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.97 Function block chain loop [ERROR 325]

Chained function block link from Module 1,27,02 Block 1196 to Module 1,27,02 Block 1197 is part of a loop

The chain of function blocks which this link is part of has a loop.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.98 Function block disabled [ERROR 297]

Module 1,01,02 Block 1001 FC3 disabled by spec values

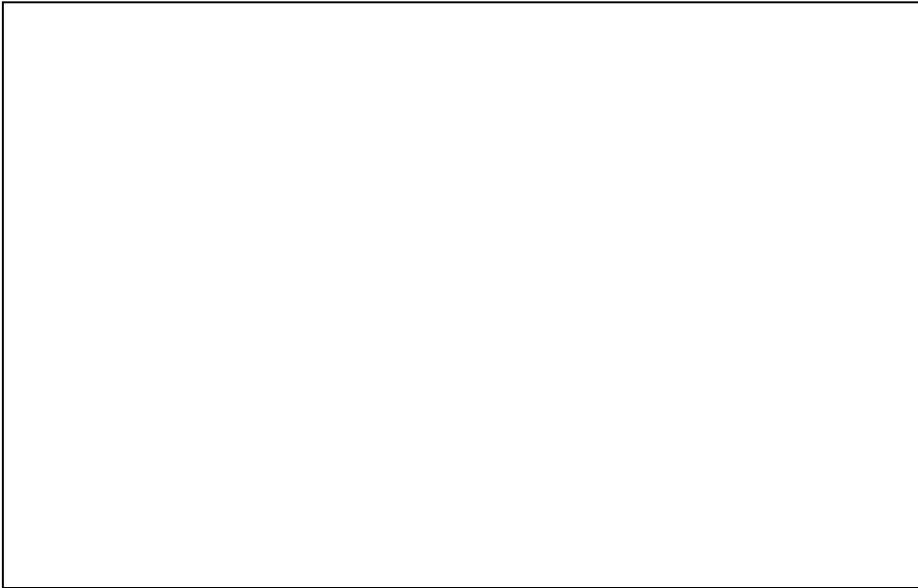
A lead/lag F(t) FC 3 or rate limiter H//L FC 8 has been disabled by the enabling specification S2 being defaulted to the disabled state 0.

This error is generated when these conditions are met:

- S1 is wired (greater than 9 is sufficient)
- S2 is 0
- S3 and S4 are NOT both 0.0

since both blocks feed the input to the output and do nothing if the above conditions are met.

If the block output can be found to be unused, the message is suppressed.



- The first example shows a lead/lag block in a control circuit that is intended to soften the dynamics of process variable changes. This fails to work because the block action is disabled by default.
- The second similarly shows that the amount of rate limiting is intended to be monitored, but this monitoring fails for the same reason.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC FUNCTION DISABLED**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.99 Gain spec is adapted [INFO 119]

```
Gain spec is adapted: S3 of Module 1,05,03 Block 6168 FC 16 adapted by Module  
1,05,03 Block 6189
```

This message indicates a gain spec is adapted.

Verify that the intent of the logic is to have the gain controlled by an Adapt block. It is well worthwhile putting a valid default on the gain, too, to make validation against the design value possible if that is appropriate.

The intent of this message is to help you find all the instances of this in your system, so you can document them for future reference.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO GAIN SPEC ADAPTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.100 Got lost parsing file [INTERNAL 10007]

```
Got lost parsing file: expected a C record and got 4660
```

This message means that a file contains information that DBDOC cannot handle.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.101 Graphic link target is null [INTERNAL 291]

```
Target is "null" for graphic link at 1000,2000 on GRAPHIC [X=2377.23,Y=100.45]
```

This link does not have a target graphic. Both the DBDOC coordinates and the 800xA X and Y coordinates are given.

The objective of this message is to show you visually, with a red coordinate ball, where a graphic link is set to be null. Both the DBDOC coordinates and the 800xA X and Y coordinates are given.

The DBDOC coordinates "at 1000,2000" show you where the red ball is. You can double-click on it to hide it, in fact. The 800xA coordinates [X=2377.23,Y=100.45] show you where 800xA had the null link.

This may be a broken link, or it may be an unused link.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL PG2.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.102 Harmony I/O block illegally wired [ERROR 095]

Harmony I/O Module 1,01,02 Block 1001 FC 228 S2 wired from IREF NoSource with no source: module will compile but not run

This message means that the function block indicated cannot have an unwired input. If it does, Composer will allow it, but it will not run.

This undocumented issue was found by Ready Engineering when they inadvertently configured some Harmony I/O with empty input references.

```
Module 1,30,05 Block 198 FC 228 S3 wired from IREF (NB-1-30-05-01.08)with no source:
module will compile but not run.
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC CHAINED FC PROBLEM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.103 History tag references tag not in database [ERROR 253]

History tag ABC123 references ABC123.<tagatom> (not found in any database)

Only Conductor NT systems can produce this message.

The history database refers to a tag that does not exist in the tag database associated with this console. The history tag has not been defined, although it is shown. Note that trend definitions are also affected. Since this history tag has no meaning, there is clearly a problem with any trend group that references it.

Example:

```
History tag CBTE01 references CBTE01.C_APV (not found in any database).
```

The Historical Tag can have any legitimate name, including CBTE01. However, the "Tag Atom" entry must refer to a legitimate database value. In this case, the problem is that there is no database tag "CBTE01", which means that Tag Atom "CBTE01.C_APV" is erroneous. If the entry in Tag Atom were changed to have the "CBTE-01" in the name, there would be no error.

Usually, this message comes because no such tag is in the database at all, often because it did exist but was deleted. DBDOC checks that the tag name and tag atom make sense.

In the example, the error refers to an invalid Historical Tag, not a missing one. The other form of the error is, of course, to name a trend tag that does not exist at all. To resolve that error, you either eliminate the trend tag entry in the Trend Configuration section, or you correct the spelling of the Trend Tag entry in the Trend group.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC HISTORY TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.104 Ignoring adapted Block 0 [ERROR 082]

Ignoring Block 0 adapted by Module 1,01,02 Block 1001

This message means that an adapt block is adapting a Block 0 block. DBDOC ignores Block 0 configuration.

On-site testing has confirmed that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT EXECUTE ERROR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.105 Illegal adapt block, spec 3 is 0 [ERROR 116]

Ignoring Module 1,01,02 Block 1001 spec 0 adapted by Module 1,01,02 Block 1234

This message means that an adapt block is adapting a spec 0.

On-site testing confirms that an adapt block continues to adapt its target block / spec, even if it has no input. Since block types cannot be changed, the unintentionally adapted spec is legal, but not likely correct.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.106 Illegal import (not exception-reported) [ERROR 129]

```
Illegal import: Module 1,01,02 Block 1001 on sheet 1010200A.CAD imports Module 1,06,06 Block 1234 (not exception-reported).
```

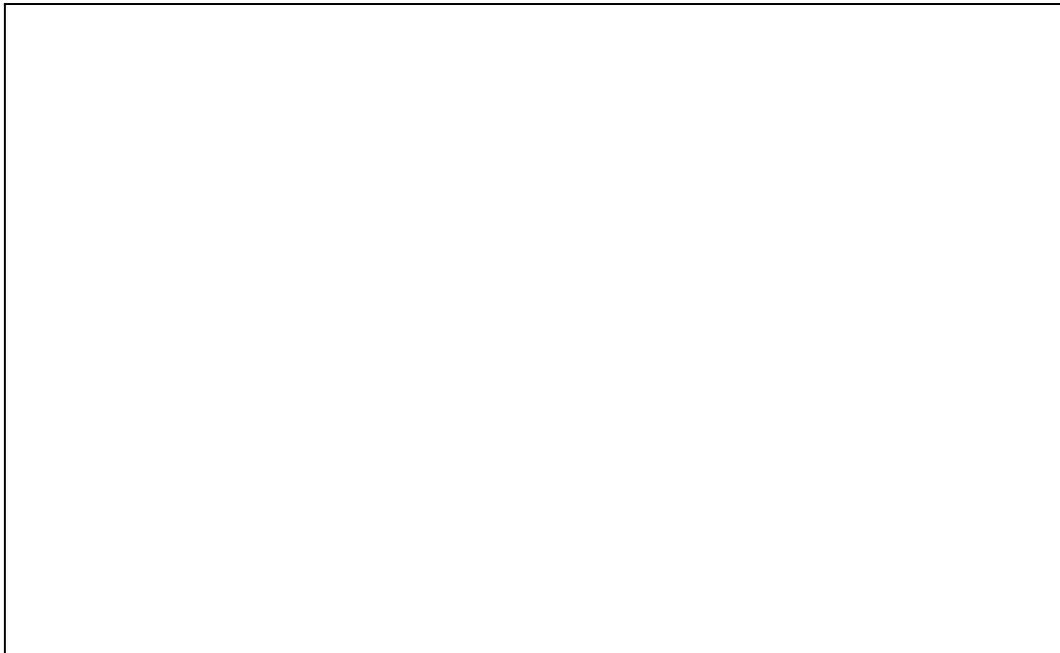
This message says that we found a block that is being imported illegally. If a block that is not an exception report is being imported from the loop, it will not work.

Example:

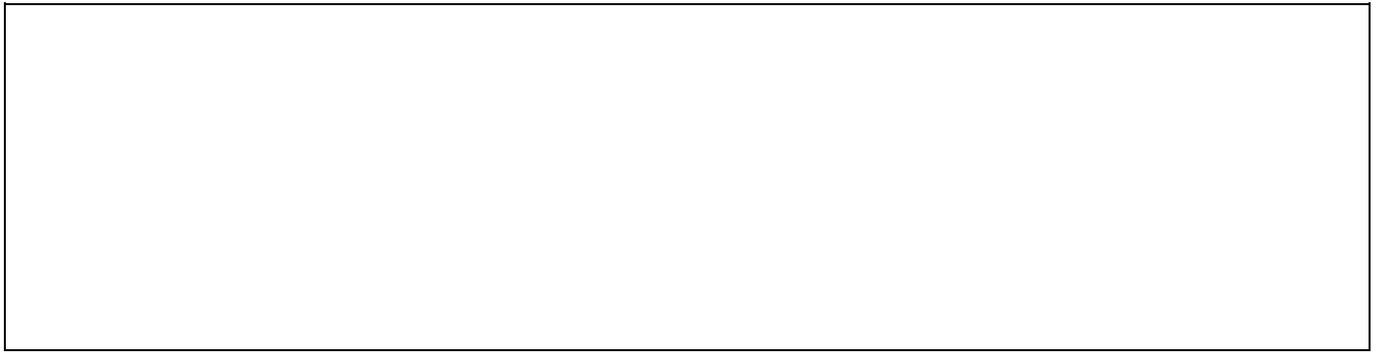
```
Illegal import: AI/L Reference on sheet 1470304A.CAD imports Module 1,10,02 Block 1010 (not exception-reported).
```

This example shows the attempt to use an AI/L block to import the output of a F(x) FC 1 block, which is not an exception report.

Often, this error leads to uncovering unused or reused logic. Blocks are sometimes used without the knowledge that their values are being imported in another module, PCU or loop. You get a check for this situation on every DBDOC build.



```
Illegal import: DI/L Reference on sheet 10404X1A.CAD imports Module 1,03,02 Block 4129 (not exception-reported).
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL IMPORT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.107 Illegal import (type mismatch) [ERROR 131]

Data type mismatch: Module 1,06,06 Block 1244 on drawing 1060621A.CAD attempts to import analog value from Module 1,01,02 Block 1001

This message says that we found a block that is being imported illegally. Source and import block types do not match. These imports might not work. They often show a configuration error.

Example:

Data type mismatch: DI/L Reference on drawing 3070341 attempts to import analog value from Module 3,05,03 Block 287



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL IMPORT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.108 Illegal import (different loop) [ERROR 130]

```
Illegal import: Module 1,01,02 Block 1001 on sheet 1010200A.CAD attempts to
import Module 1,06,06 Block 1234 (different loop)
```

This message says that we found a block that is being imported illegally. These imports will not work.

We have not found any instances of this error in any of our test builds. If you should get this error, please contact GMCL. This will allow us to test the error in greater detail, and help identify the problems you may be having. Thank you for your patience.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL IMPORT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.109 Illegal import (full status ignored) [CHECK 132]

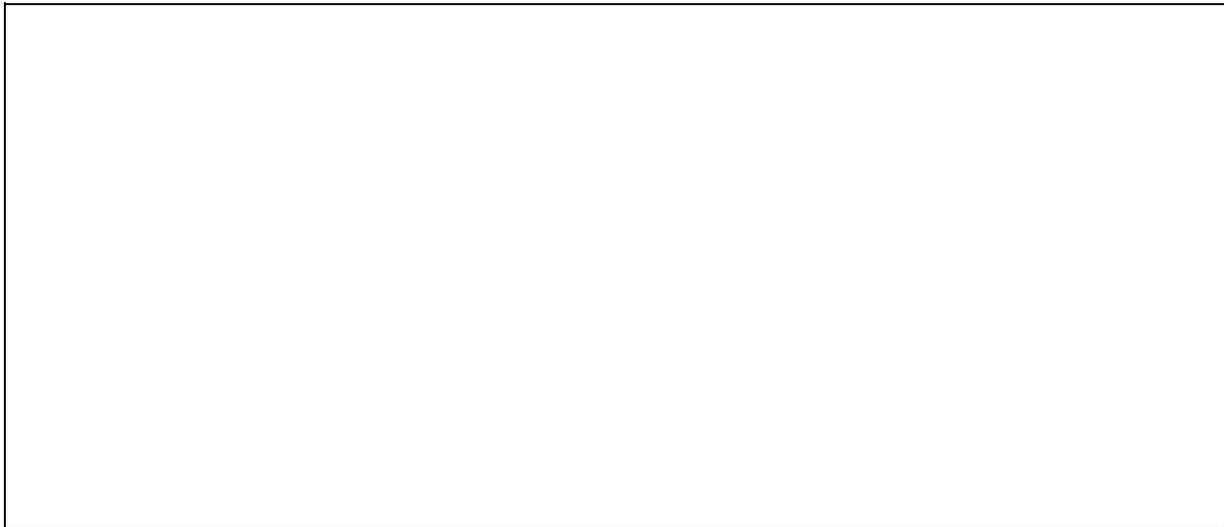
```
Extended status not imported: DI/L Reference on drawing 10104 BUS/LOOP INPUTS -
LOOP DIGITAL02 imports DADIG from Module 1,03,06 Block 1240
```

This message is a warning that the second output of a DAANG or DADIG block (full extended status information) is not being used.

Note that DAANG blocks have an extended status that cannot be imported by AI/L and AI/I functions. You must use DAANG/I to import it.

Example:

```
Extended status not imported: DI/L Reference on drawing 10104 BUS/LOOP INPUTS - LOOP DIGITAL02
imports DADIG from Module 1,04,02 Block 760
```



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK ILLEGAL IMPORT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.110 Illegal module value [INTERNAL 066]

```
Illegal value for module parameter 3569275""
```

DBDOC has encountered an illegal value for a module parameter.

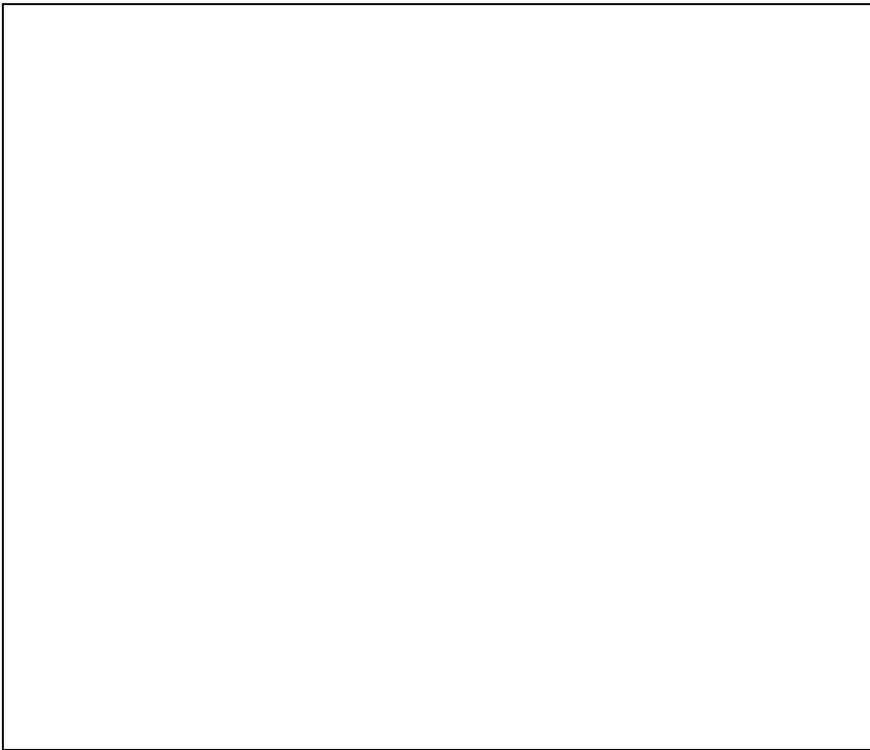
This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.111 Illegal spec value: should be 0 or 1 [ERROR 149]

```
Illegal spec value for Module 1,01,02 Block 1001 (FC 79): trip spec S4 has value  
3, should be 0 or 1
```

The specification shown is not legal, according to the documentation. We cannot tell you if the module will trip on failure or not.



The value S3 in DIGRP: Digital Input Group [FC84] should be either 0 or 1. The action on value 2 is not documented. The value of S3 should be set to the correct value for failing the module or switching to the redundant controller on a failure. What it does right now is not clear.

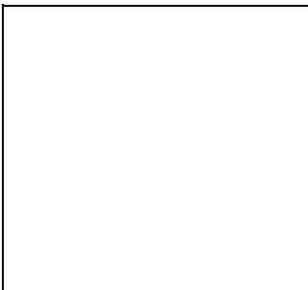
This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.112 Illegal spec value: Spec S7 from 0 - 100 [ERROR 146]

```
Illegal spec value for Module 2,31,07 Block 2446 (FC 30): Spec S7 has value  
-100000, should be 0.0 to 100.0
```

The specification shown is not a legal value, according to the documentation.



In the case of S7 of AO/L: Analog Exception Report [FC30], we can guess that the block will not properly handle exception reports.

The message always means that the specification violates the limits that are documented. In the FC 30 example, the value of S7 tells how much change causes an exception report. We have to guess that a negative value is the same as 0.0, and will cause an exception report anytime the input changes. The default value is 1 (%) and the value of S7 should be corrected.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.113 Import from same PCU [ERROR 128]

```
Import from same PCU: Module 1,01,02 Block 1001 Module 1,01,02 Block 1001
imports DADIG/I Reference from same PCU (OK in Plant loop)
```

This message means that an exception report input is being attempted from the PCU that the module is in. In an INFI Loop system, you cannot import an exception report input from the same PCU. In INFI Loop, this causes an "NPMErrror" when the module is started. Since this worked in Plant Loop, this has at times been a significant problem.

Example:

```
Import from same PCU: DI/I Reference Module 5,43,03 Block 3926 imports Module 5,43,05 Block 698
from same PCU (OK in Plant loop)
```



In this case, the error actually points to incorrect parameters that had not been detected.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC IMPORT FROM SAME PCU.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.114 Import information [ERROR 255]

Import information: AI/I Reference on drawing L8102AM-11 imports station PV from Module 21,28,06 Block 1880.

Using an AI/I or AI/L reference to import an M/A MFC/P: Control Station [FC80] block yields the PV (process value), the value on Specification S1. This means that any such import should be checked to ensure this is what is intended.

Importing the output of a Station (FC 80) gives the Process Value (PV) instead of the Output (OP). This can be misleading. We have found instances where the designer clearly intended to be importing the Output

Value. You should verify any imports like this to make sure the PV is what is intended.

The example tells a story because the error was corrected in one unit but not in its clone. Here we see the two units side by side. On the left is unit 1, where the error was corrected by adding block 9930. On the right is unit 2, which was cloned from unit 1 but the error was never corrected.



Here we see the AI/I references for units 1 and 2. Unit 1 on the left is importing from block 9930. Unit 2 on the right is still importing from S1 of block 1880.



Sometimes it is important to investigate not only why an error appeared in one unit, but why it didn't in another.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-CHECK IMPORT STATION PV NOT OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

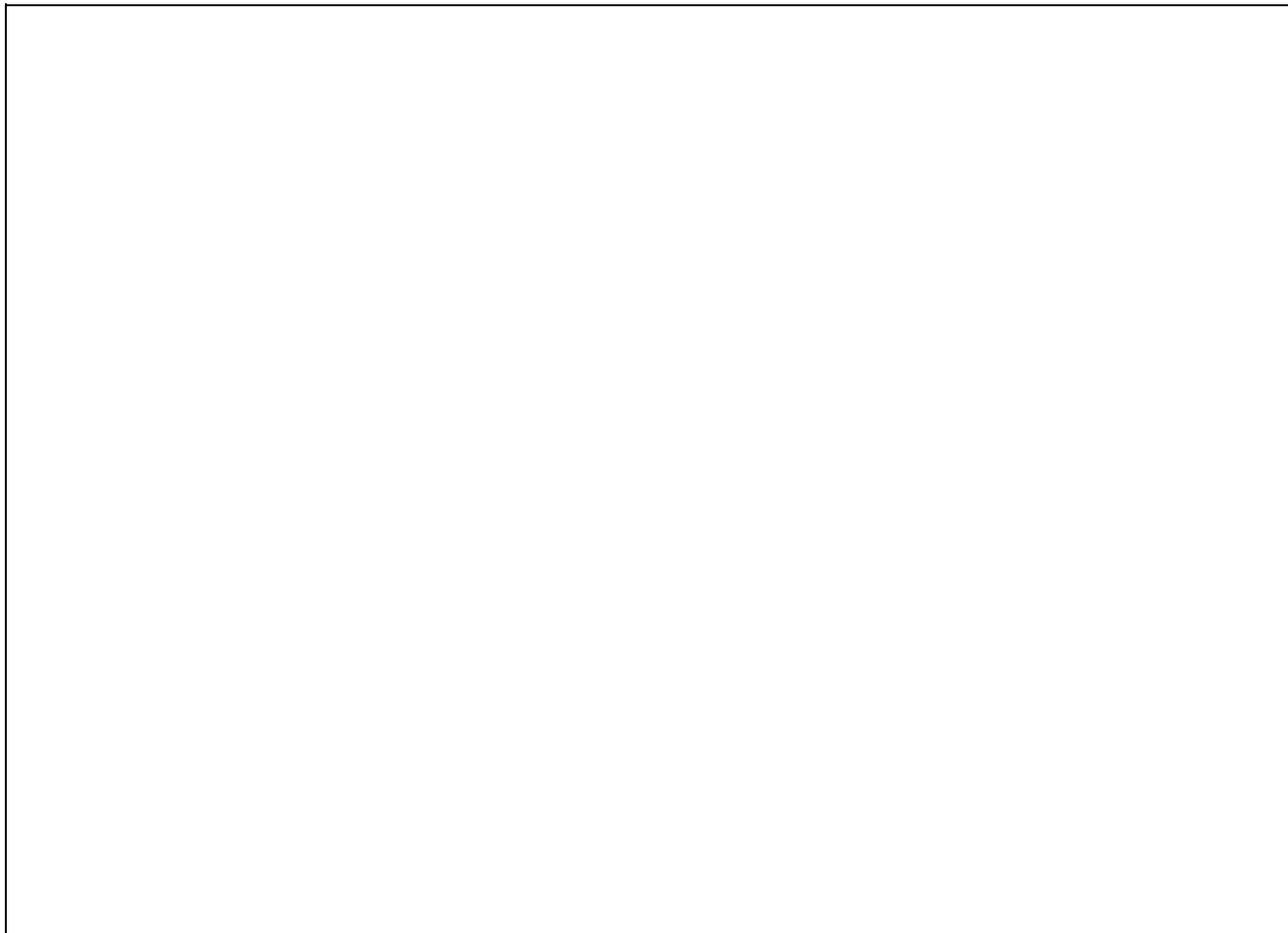
6.6.2.115 Import Module Status is wired but non-functional [CHECK 174]

Input Module Status on Module 1,01,02 Block 28600 (FC 132) is wired but non-functional

When linked, AIS/FBS: Analog Input/Slave [FC132] blocks report status only on the first block. This message warns you that testing of the second and third blocks is occurring. This means you will not get a status indication.

Applies to AIS/FBS: Analog Input/Slave [FC132] blocks.

When multiple blocks are attached together in sequence, the outputs of the earlier blocks are incorporated into the later ones. This means that the earlier blocks need not also be attached to the use of the last block in the chain. Because of this, the earlier block is ignored by the use.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK AIS NON-FUNCTIONAL STATUS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.116 Import out of range [ERROR 336]

```
DIL/B Reference Module 1,01,02 Block 1001 imports out-of-range Module 1,01,10  
Block 10000 (maximum block 9998)
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.117 Imported block has no source - module not built [ERROR 265]

```
No source: Module 1,01,02 Block 1001 imported - module not built
```

This message means that a block is being imported from a module that was not built.

Blocks may be from modules that you have not included in this build, in which case this file gives you a check list.

External systems may put values on blocks that are seen by the loop as coming from an indicated module that does not actually exist.

There are three circumstances that generate this message:

1. *The block comes from an external source like PI or FDI.*

Example:

```
--Generating index: Module 2,01,08 Block Index - module not built
No source: Module 2,01,08 Block 1350 imported - module not built
No source: Module 2,01,08 Block 3131 imported - module not built
No source: Module 2,01,08 Block 3220 imported - module not built
No source: Module 2,01,08 Block 3222 imported - module not built
No source: Module 2,01,08 Block 3227 imported - module not built
No source: Module 2,01,08 Block 3229 imported - module not built
No source: Module 2,01,08 Block 3236 imported - module not built
No source: Module 2,01,08 Block 3239 imported - module not built
No source: Module 2,01,08 Block 3275 imported - module not built
No source: Module 2,01,08 Block 5017 imported - module not built
```

Here we can see that the first and last blocks are well off on their own, meaning they may be orphans, and potential errors. The middle section of consecutive blocks are probably correct, but should be checked anyway.

2. *The block comes from a module that was not built.*

Sometimes there are multiple INFI 90 systems in a plant. When you build all your systems together, DBDOC will check them against each other and many of these messages will disappear.

3. *The error block has an incorrect specification.*

In BuildPlus, use **BuildPlus/Tools/Project Options/Define Blocks with External Sources** to define blocks that do not exist to eliminate the messages if they are valid references. Input the Loop, PCU, Module and Block Range info for external blocks that actually exist.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC BLOCK IMPORTED MODULE NOT BUILT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.118 Included image was not found [CHECK 294]

```
Included image Data.jpg was not found
```

```
    Included image a.bmp was not found
```

The DBDOC build system was unable to find an image which is included in one or more of your files.

DBDOC looks for **Symphony Plus or PGP graphics**, .jpg, .gif and .png images:

1. in the exact path specified by the Symphony Plus or PGP graphics;
2. in the path specified in the Configuration Wizard
3. in a path starting with the location of the Symphony Plus or PGP graphics then within the display folder matching the path specified in the graphic.

To resolve this error for Symphony Plus or PGP graphics, open the Configuration Wizard and add the correct file path to the missing .jpg, .gif and .png images.

DBDOC also looks for .bmp files identified in **AutoCAD files**, in the location of the corresponding AutoCAD file. To resolve this error for AutoCAD files, move or copy the .bmp files to the locations of the AutoCAD files.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK INCLUDED FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.119 Inconsistent LPM data [ERROR 10020]

```
Inconsistent LPM data for Module 1,01,02 Block 12455 FC 45, output 3: GlobalIRef  
(Module 1,03,04)
```

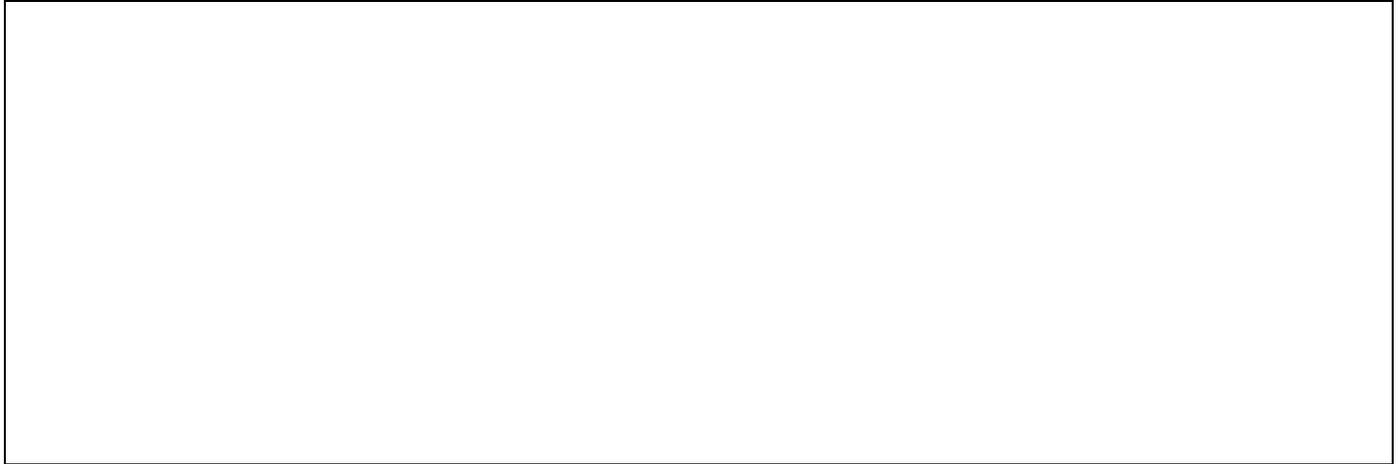
These messages cover situations where IREF text exists on an off-module input, but the reference text is already used as a local OREF reference.

This can arise in a conversion from WinCAD when the text on a local reference is the same as that needed on a global OREF required by an import block. Before Composer, the text on the import function code was not looked at.

The use of the same reference text in both a local reference and a global one is not valid. It would not be allowed in Composer, but can be imported. There may be no problem with the logic, but it is not robust. Either local or global reference should be made unique. As it is, there is no robust connection by way of Composer, because the text cannot be resolved.

Example:

```
--Parsing file E:\EBY45EB.CLD...  
No legal LPM match found for IREF **1AOIK CB CLSD on output 1.
```



In the example, it is possible that the real problem is that the DI/B import block, which is restricted to Loop 2 PCU 30, should actually be replaced with a DI/L block that can import by exception report from Module 2,20,04 Block 6839. Examination of the logic is required to find out how to resolve the problem.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC LPM REFERENCE PROBLEM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.120 Increasing drawing height [COSMETIC 023]

```
Increasing height of drawing 1010904 to 2580
```

The sheet's border was too narrow. During DBDOC processing, we extend the CLD/CAD sheet to be big enough so that we can show all the drawing elements.

This message can mean that someone converted the WinCAD project to Composer without reconciling errors or printing the sheets. Composer might throw in a default border for them.

This is not an error, but you can probably find the border that DBDOC could not find for your next rebuild.

We also find that some users of Composer make a SPARE sheet on a CLD for blocks on that CLD that they no longer are using. They do not worry about making it pretty, as they intend to consolidate the spare blocks eventually.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC NO BORDER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.121 Increasing drawing width [COSMETIC 022]

```
Increasing width of drawing 1010901 to 2580
```

The sheet's border was too narrow. During DBDOC processing, we extend the CLD/CAD sheet to be big enough so that we can show all the drawing elements.

This message can mean that someone converted the WinCAD project to Composer without reconciling errors or printing the sheets. Composer might throw in a default border for them.

This is not an error, but you can probably find the border that DBDOC could not find for your next rebuild.

We also find that some users of Composer make a SPARE sheet on a CLD for blocks on that CLD that they no longer are using. They do not worry about making it pretty, as they intend to consolidate the spare blocks eventually.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC NO BORDER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.122 Inhibit tag not found [ERROR 348]

```
Inhibit tag for ABC123 not found: {01234567-89ab-cdef-0123-456789abcdef}
```

DBDOC will attempt to find and replace the ID in the INHBTAG field of your database with the name of the corresponding tag. This error indicates it was not able to find the tag with the given ID.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.123 Input list attempts to import nonexistent block 9999 [ERROR 334]

```
Module 1,01,02 Block 1009 imports nonexistent Module 1,01,10 Block 9999 (should  
be set to 31999 to be spare)
```

The imported blocks are set by specifications:

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.124 Input list attempts to import out-of-range block 9999 [ERROR 335]

```
Module 1,01,02 Block 1009 imports out-of-range Module 1,01,10 Block 9999 (must  
be set to 31999 to be spare)
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.125 Input list imports block 9999 [CHECK 333]

```
Module 1,01,02 Block 1009 imports Module 1,01,10 Block 9999, but may have been  
meant to be spare
```

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.126 Input wiring expected - Harmony [CHECK 187]

```
Harmony I/O FC (222-225, 229) block expected: Module 1,99,10 Block 22561 FC 11  
S4 wired from Module 1,99,10 Block 22562 FC 229
```

This message means that the function code expected to find a Harmony block, but did not.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK INPUT WIRING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.127 Inputs have been overlaid [CHECK 226]

```
Blocks have been overlaid: Module 1,16,03 Block 419 FC 62 S1 and Module 1,16,03  
Block 423 FC 62 S7 overlaid at 2500,1640
```

This message tells you that an input has been placed on top of another input. In the case of negative block numbers, it may be of no consequence, but we show the message just in case it is important.

Example:

```
Blocks have been overlaid: Module 1,16,03 Block 22419 FC 62 S1  
and Module 1,16,03 Block 22423 FC 62 S7 overlaid at 2500,1640
```

This message was triggered by a set of spare blocks.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

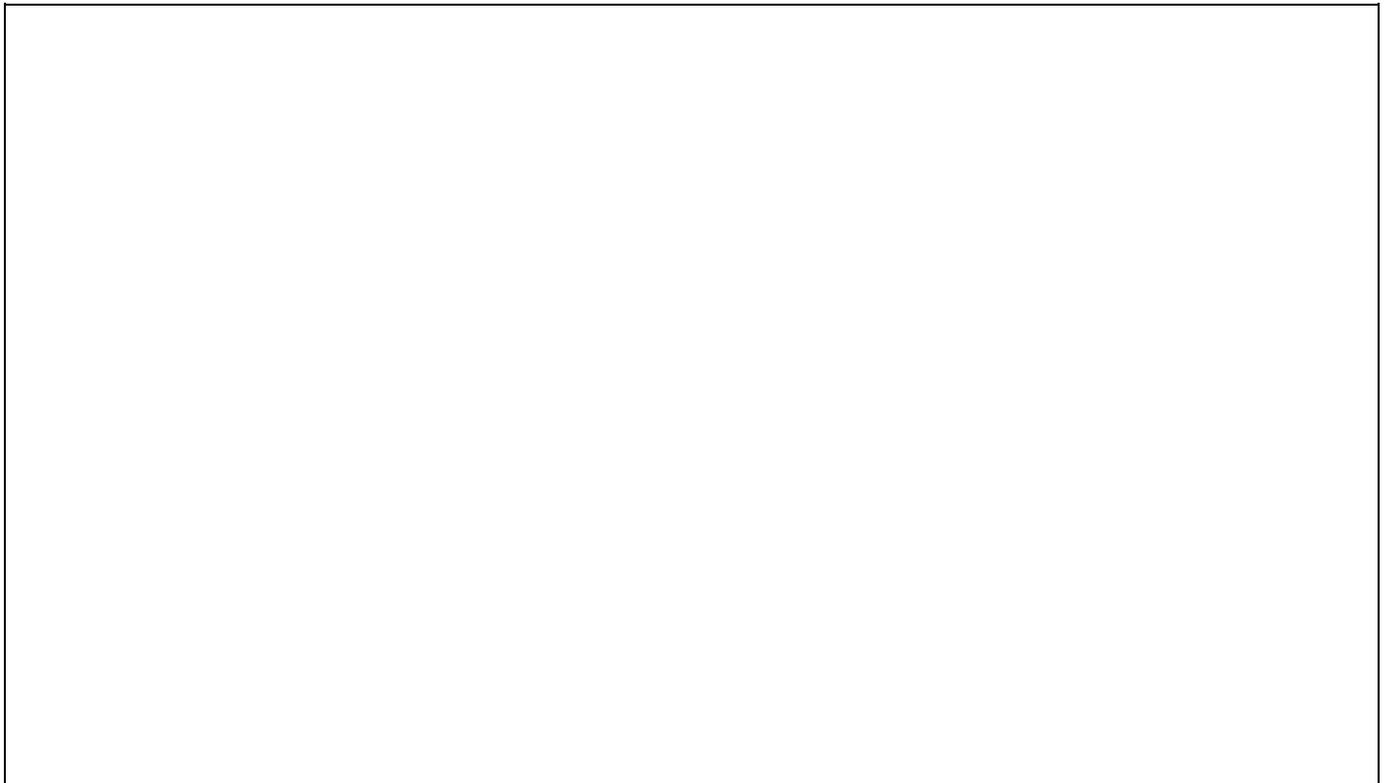
This error type was originally reported in error file **DBDOC-CHECK OVERLAID.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.128 Integrator disabled by spec value [ERROR 298]

Module 1,01,02 Block 1001 integrator disabled by S4 value

This INTEG: Integrator [FC166] block has a value of zero for S4, so it will never output anything but the preprogrammed initial value on S3.

An integrator block with S4 unwired gets the default value 0 applied to that specification. This means that the output is the value of what is wired to S3, and it does not integrate. If the integrator is intended to function, a constant [1] (or a Boolean signal) must be wired to S4.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC FUNCTION DISABLED**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.129 Internal error: get_block() non-block tag [INTERNAL 049]

Internal error: get_block() called on non-block tag FUNCTION_0131_

If you get this message, please contact us to make sure there are no problems with the build.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL GET BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.130 Internal error: Unknown hotspot [INTERNAL 016]

```
Internal error: unknown hotspot type 99
```

DBDOC has encountered an internal error.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.131 Invalid block number on line [INTERNAL 246]

```
Invalid block number on line 0
```

This message says that DBDOC found an invalid block it can't interpret.

If DBDOC is unable to interpret a block in your system, please contact GMCL so we can look into the problem.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID NUMBER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.132 Invalid CAD file: Too short [INTERNAL 010]

```
Invalid CAD file: too short (124 bytes)
```

This message means that a CLD/CAD file was found that we cannot process because the CLD/CAD file is too short.

Please try to open the file with your Composer or CAD system. If it seems to be fine, send it to us for analysis. If not, there may be a problem to be addressed at your end.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID CAD FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.133 Invalid CAD file: Zero length [INTERNAL 009]

```
Invalid CAD file: zero length
```

This message means that a CAD file was found that we cannot process.

Please try to open the file with your CAD system. If it seems to be fine, send it to us for analysis. If not, there may be a problem to be addressed at your end.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID CAD FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.134 Invalid header or bad magic number [INTERNAL 10011]

```
Invalid header or bad magic number - ABCDEF01 23456789 - skipping file
```

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.135 Invalid loop number for block [INTERNAL 247]

```
Invalid loop number 256 for Module 2,40,31 Block 21581
```

This message says that DBDOC found an invalid loop number it can't interpret.

The maximum value for a loop number is 255. if the value is greater than 255, DBDOC cannot interpret it.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID NUMBER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.136 Invalid loop: high byte used [INTERNAL 248]

```
Invalid loop number 256: using high byte (1) for Module 2,40,31 Block 21581
```

This message means DBDOC found a loop number it can't interpret. We have used the high byte, which might be correct.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID NUMBER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.137 Invalid record length [INTERNAL 003]

```
Invalid record length 512
```

This message means that a CLD/CAD file was found that we cannot process.

Please try to open the file with your Composer or CAD system. If the file seems to be fine, please send it to us for analysis.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL INVALID CAD FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.138 IREF does not have an input [ERROR 096]

```
IREF NoSource does not have an input in Module 1,01,02
```

This message means that if you compile the modules, you would have no source for the indicated IREF. The OREF needed to resolve an IREF was not found.

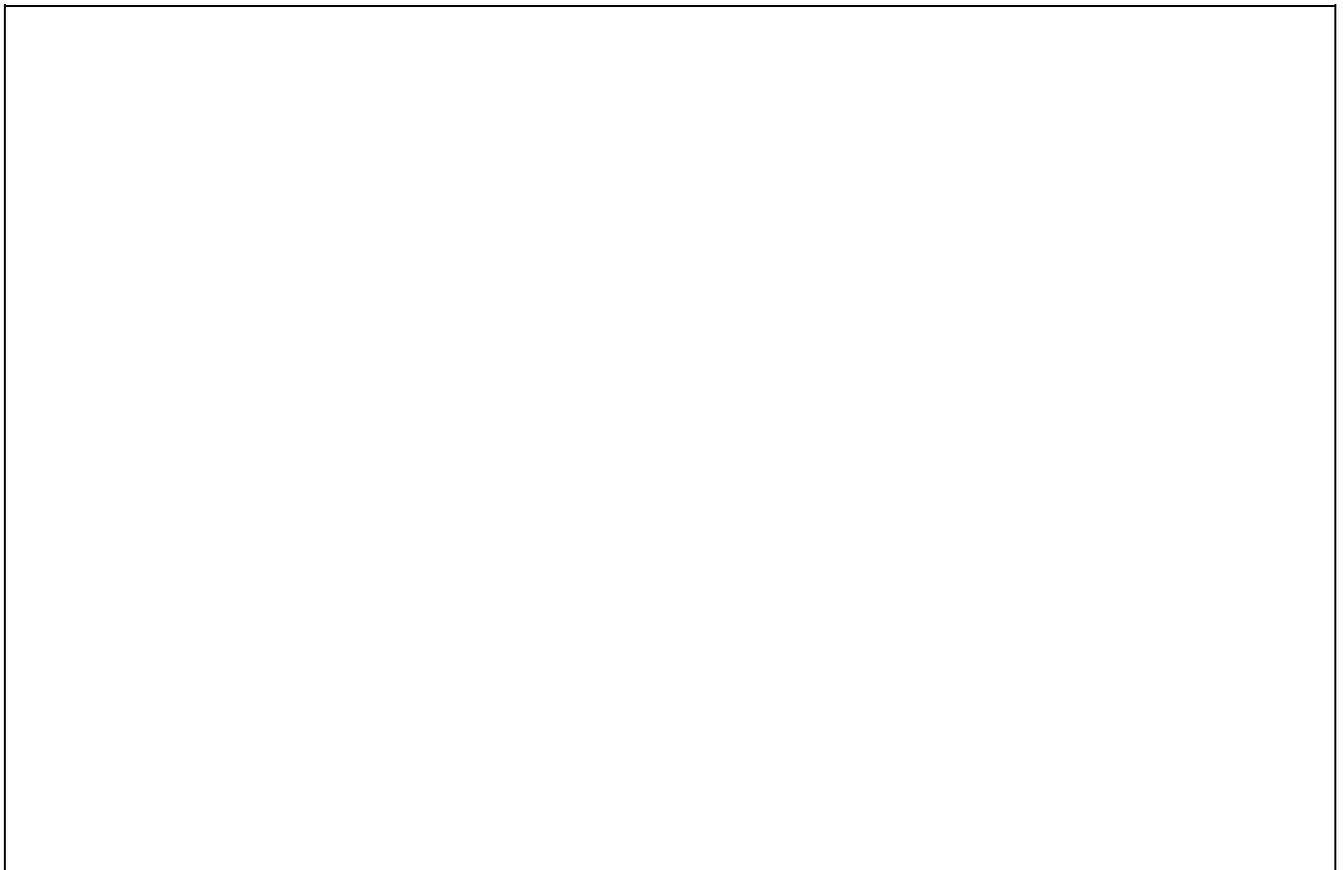
The module would not compile under Composer, WinCAD or DOS CADEWS. Best practice is to define the output reference first, then add the input references. Sometimes these messages show places where the output references were deleted.

If the OREF does exist, look for a gap in the wiring. Also, you may get a message telling you the connection is wired in Composer only.

Example:

```
IREF FV-40223A does not have an input.
```

In the example, see that OREF "PV-40223A" must have existed at one time to make the cross-reference work, but it no longer can be found.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC IREF HAS NO INPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.139 IREF GlobalIRef import mismatch [INFO 10019]

```
IREF GlobalIRef makes Module 1,01,02 Block 12455 FC 45 import Module 1,01,02  
Block 12455 instead of Module 1,01,02 Block 12455
```

These messages are special. They show that in the CLD file, the input reference text forces the importing of a block that is not the same as what the specifications indicate.

This suggests that nobody checked the block that is being imported, having Composer make the connection to an existing valid external reference.

The only way to check them is to walk through the old and new blocks and see either patterns or reasons for the differences. Here are the four possibilities:

1. The WinCAD reference was correct, but changed to a correct reference in Composer.
2. The WinCAD reference was wrong, but the Composer reference using the IREF makes a correct reference, eliminating an error that did exist.
3. The WinCAD reference was correct, but the Composer reference using the IREF makes an incorrect reference, causing an error that did not exist.
4. The WinCAD reference and the Composer IREF reference are both wrong, which is not very likely.

Cloning logic in Composer can also cause this message as follows:

1. Blocks are created without valid reference text, then copied. The specs will be duplicated.
2. Input reference text is added to the blocks, but the output references to not yet exist. The block numbers remain in the import block specs.
3. The output references are created, allowing Composer to resolve the links.

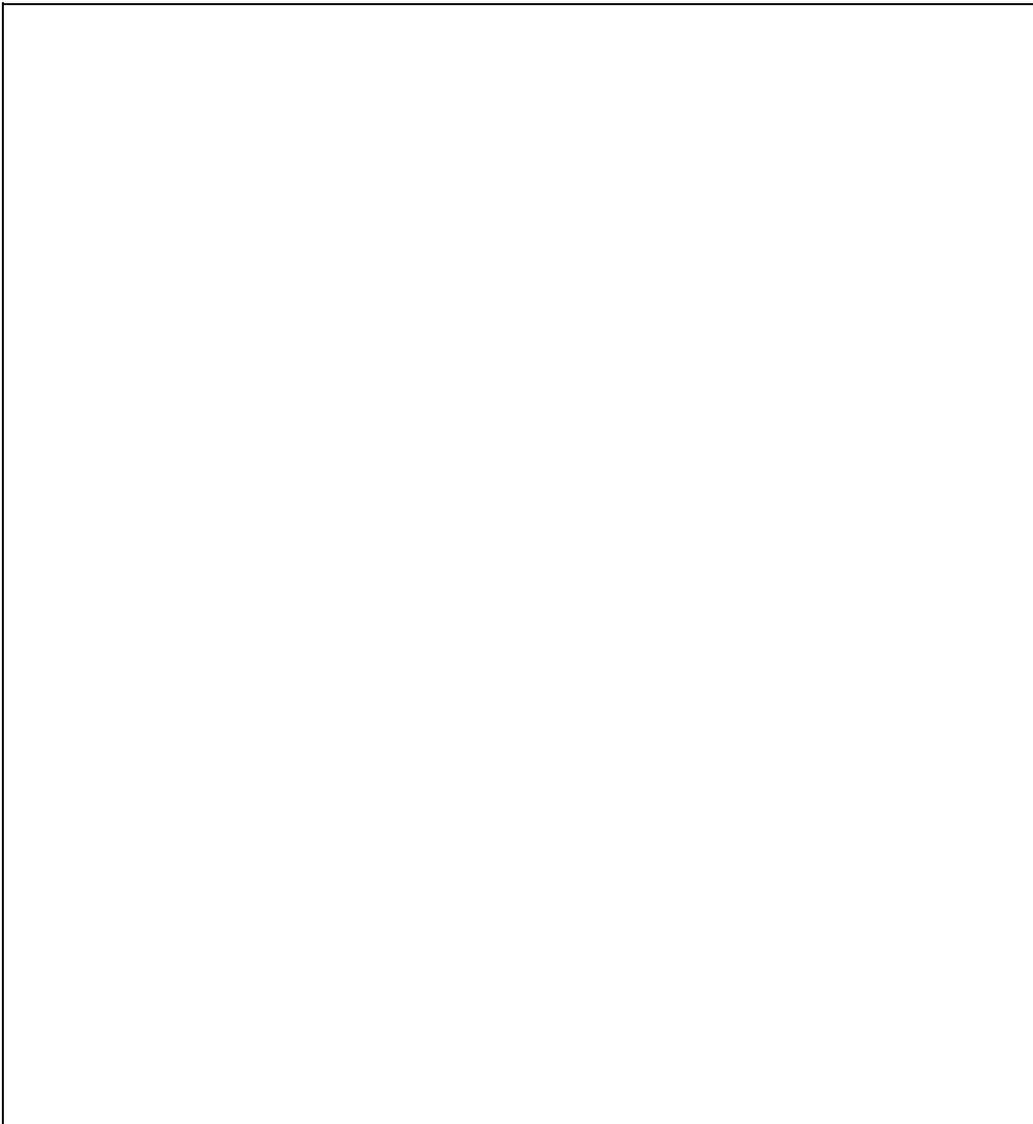
In this case, the message tells you that it is possible that the import text is not correct, as it was not entered after the output reference was created.

Removing and reentering the input reference text should make the message disappear.

Example:

```
Parsing file H:\CLIENT~1\BAR\ACTIVE~1\EBY3FF5.CLD...  
IREF **1D VVP HP STEAM makes Module 2,20,02 Block 1631 FC 26 import Module 2,53,04 Block 3678  
instead of Module 2,53,04 Block 3684.
```

This Composer diagnostic tells you that the input reference on an import block overrides the specifications in the block that used to be there. As shown here, it can indicate an error. The output reference is wrong.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO OVERRIDDEN SPEC.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.140 IRef not found in IRefRes table: set to blank [INTERNAL 10013]

```
IRef (EBY12AB.CLD,17,0) not found in IRefRes table: description set to blank
```

This message indicates that the Composer project .EBP file is corrupted. The IREF indicated was looked for in the table of reference data extracted from Composer and not found.

The CLD file contains a key (GUID - Globally Unique IDentifier) that should have been found in the master reference table, but was not. Composer might have techniques for repairing this.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G.

Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.141 IRef not found in IRefRes table: skipping [INTERNAL 10012]

```
IRef (EBY12AB.CLD,17) not found in IRefRes table: skipping
```

This message means we cannot handle a file. Please look at it with the real tools and determine if it seems legitimate. If so, send it to us. There are still surprises out there.

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

Note: if the file DBDOC-CHECK_FILE_OPEN_FAILURE is also present, sometimes they may both be resolved if the project is rebuilt from the Wizard stage.

```
File x contains no records -- skipping
Blank MTEXT record on line nnnn -- skipping
Invalid magic number ... corrupt file will be skipped
Skipping database TRUSAGE.DBF
File x contains no records
Non-local exit from file D:\FOGBUGZ TEST DATA\SMM\SMM\MODELS\SUBMODELS\U1TOPBAR.m1
Error parsing submodel U1TOPBAR, skipping
```

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.142 Legacy err message [INTERNAL 10005]

```
Legacy err message
```

This is a catch-all category for unconverted Decomposer messages. All Composer problems must be solved in files from which Decomposer synthesizes CAD sheets.

Messages stating that a file cannot be opened will disappear where they are meaningless.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **Decomposer.err**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.143 Legacy fatal message [INTERNAL 10006]

Legacy fatal message

This is a catch-all category for unconverted Decomposer messages.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.144 Legacy warning message [INTERNAL 10004]

Legacy warning message

This is a catch-all category for unconverted Decomposer messages.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.145 Library missing [COSMETIC 026]

Library SAMPLE missing: included 10 times

This message gives you a count of the number of sheets that calls the missing library. This is a cosmetic issue because no shapes are actually called from the library.

We include this message to avoid creating a large number of cosmetic messages.

The message probably appears because the root path for the library location was used.

To make DBDOC use the user libraries located in the modules, start from the Wizard and mark the user library query as omitted. If you do this, then DBDOC uses the same rules as WinCAD – it looks in the module folder for the library requested, moving up the tree to the root. This will catch the libraries where the INFI 90 system compilation finds them.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC LIBRARY NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.146 Linked Block Already Linked [ERROR 317]

```
Module 1,27,02 Block 1196 FC117 S12 = Module 1,28,03 Block 2222 FC117 is already  
linked to Module 1,09,01 Block 4444 FC117
```

With function code block chains, the next, or linked block address cannot be attached to any other block.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.147 Linked Block Connected To Self [ERROR 319]

```
Module 1,27,02 Block 1196 FC117 S12 = Module 1,27,02 Block 119 cannot be linked  
to itself
```

A function code block cannot be linked to itself.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.148 Linked Block Has Wrong FC [ERROR 316]

```
Module 5,07,09 Block 3640 FC119 S12 = Module 5,07,09 Block 3667 FC42 must be  
FC119
```

With certain function code block chains, the next, or linked block address must be a specific function code.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. **Most errors can now be viewed in the Hyperview Error Browser.**

6.6.2.149 Live wire not connected [CHECK 199]

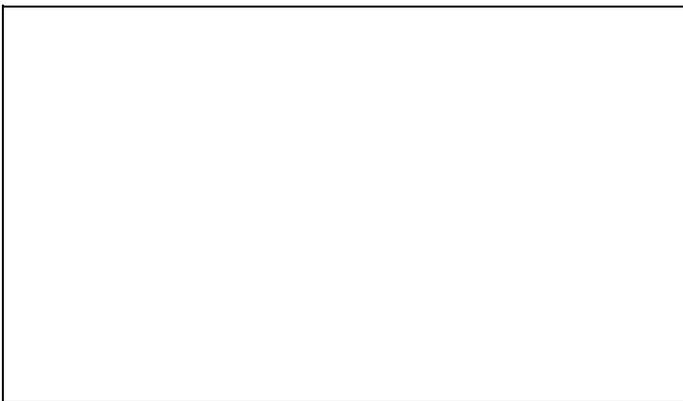
```
Live wire from Module 4,10,03 Block 4250 (FC 80) does not connect to anything -  
(3380,2340) (3380,2260) (3560,2260) on 41003H1D.CAD
```

Sometimes, a line is left attached to a source that is no longer used. Deleting the line will eliminate the message. Also this message could be saying that a signal has no connection at the coordinates shown. This message means that a signal was processed that did not end up being fed anywhere.

It could mean that a signal has no connection at the coordinates shown, but there are number of possible cosmetic causes as well:

1. A common cause is that an IREF is used to bring a visual line from an external diagram to what looks like an input on an AI: Analog Input [FC27] or DI: TCS Digital Input [FC43] block. There is actually no input for these blocks, so the message may be ignored in these cases.
2. Sometimes, a line is left attached to a source that is no longer used. Deleting the line will eliminate the message.
3. They may tell you that a signal is brought out from the named IREFs but not used. This shows up if people delete logic without deleting the IREFs that feed it.

In this example, the wire from (3380,2340) to (3380,2260) does not connect with the wire from (3380,2260) to (3560,2260):



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK LIVE WIRE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.150 Loop out of range [ERROR 164]

Loop 261 out of range (0-250) for Module 1,01,02 Block 1234 (FC 37)

This message means that the value of the loop is smaller or larger than the allowed value range.

It is possible that a spec has been entered incorrectly.



Loop must be between 0-250.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SPEC OUT OF RANGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.151 M1 display file not found (not configured) [CHECK 195]

M1 display file ADFS not found (not configured) in USFa_INTK01N_m

The graphic calls for another graphic which was not found. This is not really an error, but is a weakness for the operators using the graphic system.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.152 Macro is present in multiple libraries [INFO 10031]

Macro MACRO is present in both SHAPE and MACROS libraries (EBY1234.MCR, EBY5678.MCR)

This message is designed to improve the development of DBDOC by asking you to collaborate with us. We would like to better understand duplicated macros and shapes, so if you can explain how the messages for your system got there, and anything meaningful, please let us know.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO MULTIPLE MACROS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.153 Mismatched tagname [CHECK 205]

Mismatched tagname -- using index: testTag specifies index 1 (tag testName) and tagname testTag (index 24)

This message means that the tagname in the SODG graphic file does not match the tag index which is being used. In MCS or OIS 20 graphics, the danger is that a conversion to tagname keyed graphics (WinTools or Conductor VMS) will be done without noticing that the tagname is not correct.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISMATCHED TAGNAME.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.154 Mismatched tagname (not in database) [CHECK 204]

Mismatched tagname -- using index: testTag specifies index 1 (tag testName) and tagname testTag (not in database)

This message means that the tagname in the SODG graphic file does not match the tag index which is being used. In MCS or OIS 20 graphics, the danger is that a conversion to tagname keyed graphics (WinTools or Conductor VMS) will be done without noticing that the tagname is not correct.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISMATCHED TAGNAME.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.155 Missing associated file [INTERNAL 285]

Associated variable (or dynamic) file for Data.G not found

The file containing variable (.VAR) or dynamic (.DIAG) definitions for a graphic was not found. The missing file is needed to properly link hotspots on this graphic.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.156 Missing definitions file [INTERNAL 292]

Missing definitions file tagtypes.csv

File tagtypes.csv is missing from DBDOC program files. This usually indicates that there is a problem with your DBDOC installation. Please contact us for help fixing it.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL BAD DBDOC FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.157 Missing dynamic target [INTERNAL 275]

The dynamic IDPT_VAR1 in FILE1234.G has no link target

A graphic is trying to link to a tag but does not specify which tag to link to.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.158 Missing graphic target [INFO 314]

```
M1 display file graphic: does not have a target defined at 0,0,,0,0 in testStack
```

The target for a vector is missing.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.159 Missing LBR Report [INTERNAL 000]

```
Missing LBR Report
```

This message says that a CLD/CAD sheet called for a user library that was not found. This is a cosmetic issue because no shapes are actually called from the library.

The message probably appears because the root path for the library location was used.

To make DBDOC use the user libraries located in the modules, start from the Wizard and mark the user library query as omitted. If you do this, then DBDOC uses the same rules as WinCAD ? it looks in the module folder for the library requested, moving up the tree to the root. This will catch the libraries where the INFI 90 system compilation finds them.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.160 Missing M1 not defined [CHECK 267]

```
M1 display file graphic:302 not defined in testStack
```

This message means that the graphic indicated calls for another graphic, which was not found.

The usual sources of this error are:

1. Failure to build all graphics in the DBDOC document. If the graphic is not built, the DBDOC process will not find it. This is normally the case when there are a lot of messages in this file.

2. Graphics becoming obsolete and being eliminated, without correction of other graphics that call for the obsolete graphic.
3. Graphics being planned and anticipated that have not yet been built or installed.
4. Errors in pop-ups.

When there are a lot of these errors showing up, it normally means that all the graphics needed were not built, or the graphics are not used. It suggests that operator graphic keys and touchpoints may be "dead" because those graphics are not in the system. In Hyperview, go to "MISSING GRAPHICS / Index of Unconfigured Graphics" in the "System Information" Chapter to see where these graphics are referenced.

You can find where the graphic is linked using either the Audit Window or the Index of Unconfigured Graphics in the System Information chapter.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.161 Missing string introducer [INFO 10040]

```
Missing string introducer at offset 4660
```

This overview intentionally left blank, by order of Geoff Michaels.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.162 Missing symbol or sub-model [BUILD 099]

```
Missing symbol or sub-model MISSING
```

This message means that a call was found for a symbol or sub-model DBDOC did not find when the graphic was being compiled. Our DBDOC picture of the graphic may not be valid if you have missing symbols or submodels.

The usual sources of this error are:

1. Failure to include all symbols used in the DBDOC paths
2. The symbol may be obsolete, but the graphics using it are unchanged
3. The symbol may have been lost or not shipped by a supplier

We suggest you find these missing symbols and then rerun the wizard to include the symbols into the system.

Example:

```
--Scanning graphic 6-CAS002.DR  
Missing symbol or sub-model S-PVALVR.  
Missing symbol or sub-model S-PVALVU.
```

DBDOC makes links to these missing symbols for you. For the example above, you can see this here.



The Missing Symbols Index lists all missing symbols, and each symbol has a list and link to the graphics it is supposed to be found on. DBDOC thus makes it easy to find where these symbols are on the graphic so you can solve the problem that the missing symbol represents.



If you end up with numerous missing symbols, it is probable that your build is not configured properly. Work with us to resolve the problem so DBDOC will be the most use to you.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD MISSING SYMBOL.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.163 Module mismatch: does not match module in file [ERROR 10021]

Module mismatch: Module 1,01,02 Block 12455 FC 45 module 4 for IREF GlobalIRef (S5) does not match module 6 in file -- skipping

The input reference refers to a module that is not the one indicated by the module specification (S2) in an AIL/B or DIL/B function.

This message means that the input reference refers to a module that is not the one indicated by the module specification (S2) in an AIL/B or DIL/B function.

```
--Parsing file E:\...\RI_UNI~4\EBY1E5C.CLD...  
Module 8 for IREF MILL GROUP F IN SERVICE~ (S3) does not match module 0 (S2) in file -- skipping.  
Module 4 for IREF ANY W.U. VLV IN SERVICE~ (S3) does not match module 2 (S2) in file -- skipping.  
Module 4 for IREF EF ANY IGN OR W.U. TRIPPED~ (S4) does not match module 2 (S2) in file -- skipping.
```



Looking at the logic, you can see that what has happened is that an unused DI/L block has been put on a boneyard sheet. The natural outputs, blocks 813 and 814, are not used anywhere else.

When you run into a DBDOC error message of this type, it makes sense to get rid of the offending text eventually in a safe, effective manner. In this Composer case, the reference text should be deleted, which would make the messages go away.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC IREF DOES NOT MATCH MODULE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.164 Module mismatch: does not match Module in file -- skipping [ERROR 10038]

```
Module mismatch: Module 1,01,02 Block 12455 FC 45 Module 1,01,02 for IREF  
GlobalIRef (S5) does not match Module 1,02,04 in file -- skipping
```

This message means we cannot handle a file. Please look at it with the your tools and determine if it seems legitimate. If so, send it to us. There are still surprises out there.

The input reference refers to a module that is not the one indicated by the module specification.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC IREF DOES NOT MATCH MODULE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.165 Module out of range [ERROR 166]

```
Module 42 out of range (0-32) for Module 1,01,02 Block 1234 (FC 37)
```

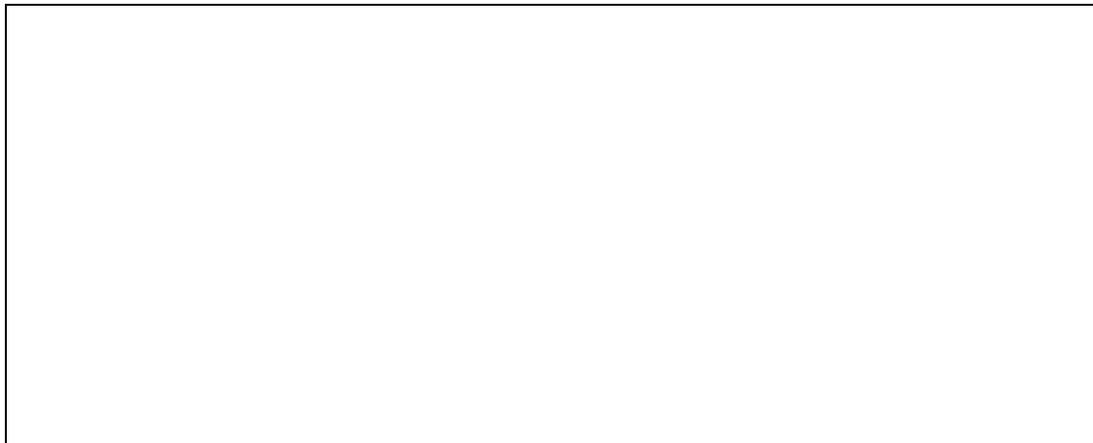
This message means that the value of a module was smaller or larger than the allowed value range.

A common cause is that S2 is the intended value for the module and S1 is intended block value.

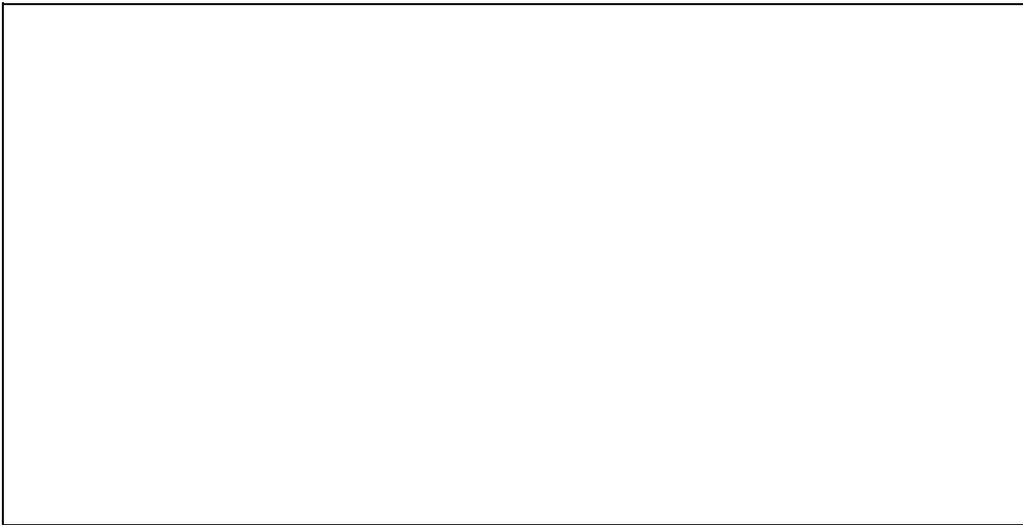
This error can trigger other errors:

```
Module 8713 out of range for Module 2,50,04 Block 1699 (FC 121).  
Module 8605 out of range for Module 2,50,04 Block 1698 (FC 121).  
Internal error in LPMToString: module 713 out of range
```

Note that 713 is the lower part of the specious value of S1=module=8713. Correcting the value of S1 and S2 (swapping them) will fix the Internal error, too.



```
Module must be between 0-32.
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SPEC OUT OF RANGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.166 More than one input to connector [ERROR 106]

```
More than one input to connector at 1240,2310: Module 1,01,02 Block 1001 (FC 1)
and IREF (REFERENCE)
```

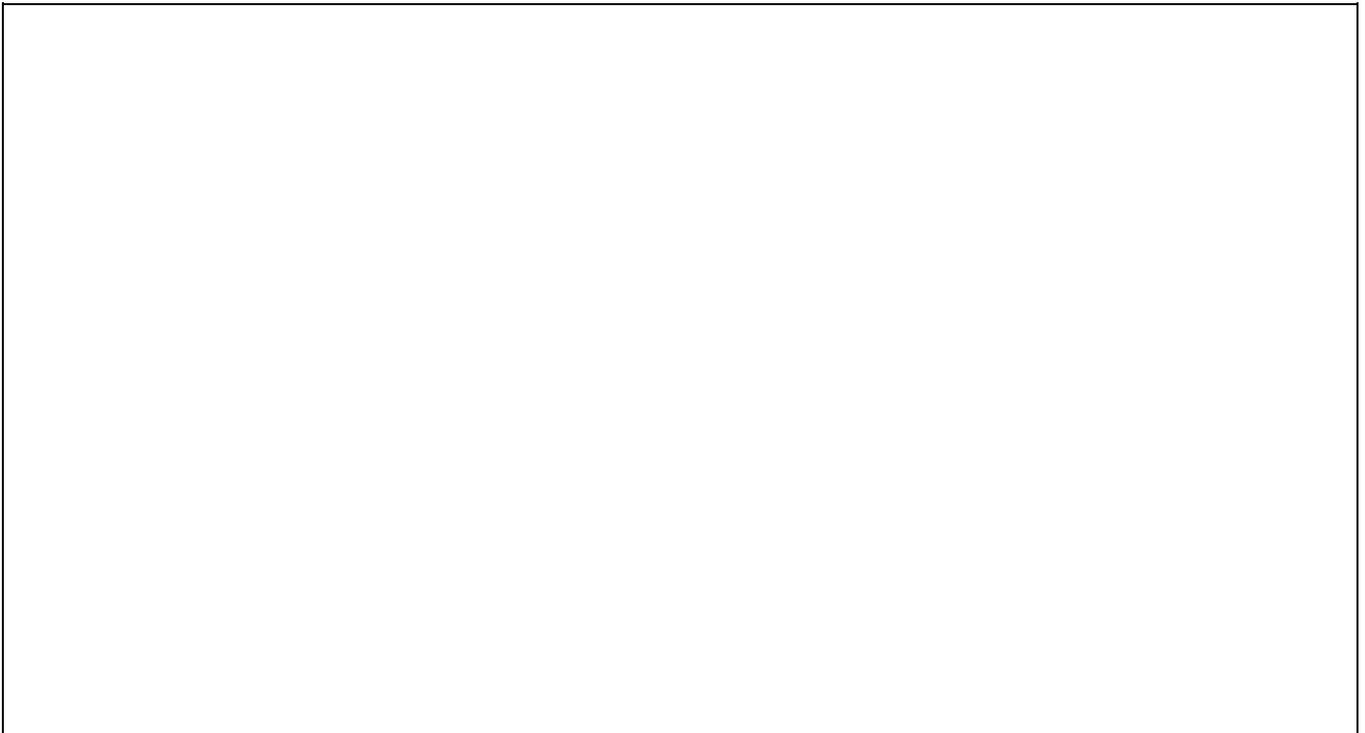
This message means that signals from more than one source are being brought into a connector at the coordinates indicated. This message is often cosmetic.

Opening up the sheet in Composer or WinCAD will allow you to remove the extraneous lines and connectors until a single path remains.

When this message mentions two different sources, it is clear that only one works. You can find out which one and resolve accordingly.

```
--Scanning drawing 9570357A.CAD
```

```
More than one input to connector at 1600,2600: Block 1788 (FC 129) and Block 1788 (FC 129).
```



You can see that the extraneous line from block 1788 to S1 of block 1793 causes this message.

It appears that the signal from the two function codes shown are coming into a connector at the coordinates indicated. More than one input to connector at 6860,5960: Block 1606 (FC 30) and Block 1607 (FC 66).



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC MULTIPLE INPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.167 Multilevel alarms disabled [ERROR 062]

```
Multilevel alarms configured but disabled for Module 11,03,04 Block 427 FC 177  
[1-LI-1067-M]
```

This message tells you that a DAANG block appears to be configured to support multilevel alarming, but it is disabled.

There are a number of specific parameter values required to make DAANG: Data Acquisition Analog [FC177] blocks work. This message tells you that one or more of the 2nd high, 3rd high, 2nd low and 3rd low alarms are specified by non-zero values of S22, S23, S26 and S27 and that it seems that the extra alarm limits are intended to be in use.

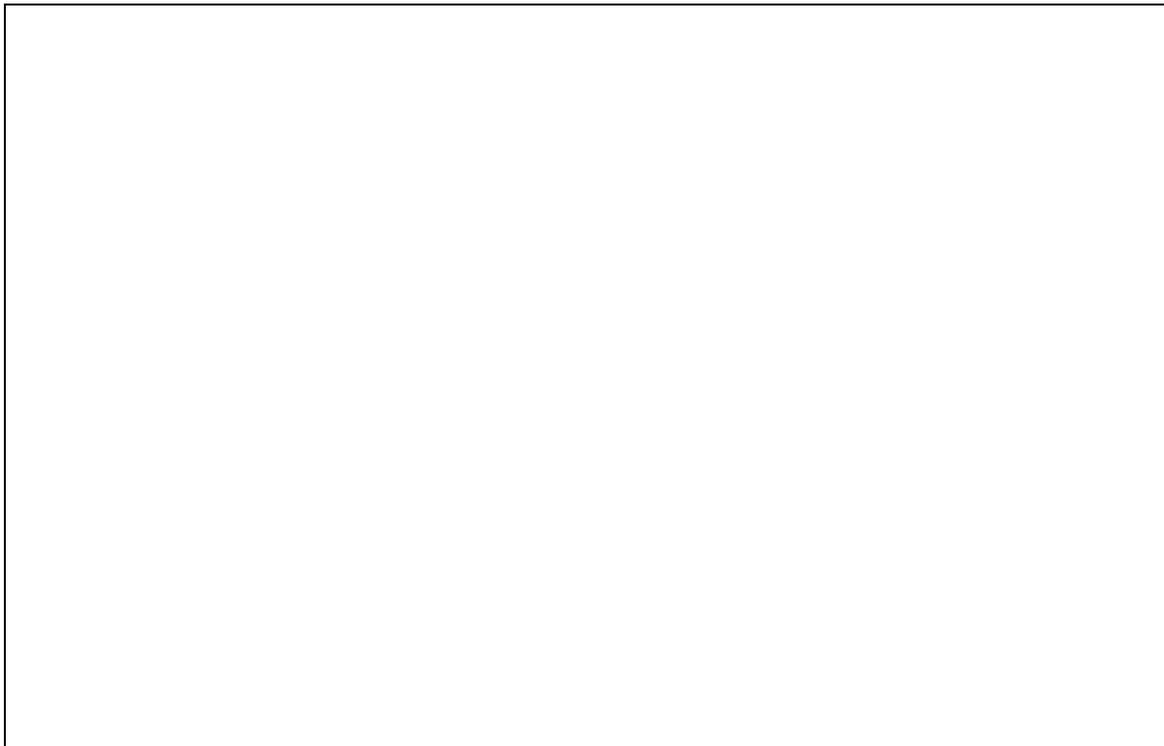
Bit 2 - Value 4 - Multilevel alarming enable:

- 0 = one high level alarm, one low level alarm
- 1 = three high level alarms, three low level alarms

If the alarm differences are configured, and S20 does not have Bit 2 set, we warn you that multilevel alarming is not enabled.

Example:

```
Multilevel alarms configured but disabled for Module 11,01,04 Block 427 FC 177.
```



We infer that the second and/or third alarm limits are intended to be used by comparing them (if not zero) to the span of the block. If they are less than or equal to the span, we assume that the the extra alarm status checking might be desired.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC DAANG SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.168 Multiple slave FCs [ERROR 153]

```
Multiple slave FCs on Module 1,01,02 slave 3: see slave report (Module 1,01,02
Header) for details
```

This message means that DBDOC's analysis of the configuration of your slave FC specs suggests that there may be a problem with too many slaves chained together or with more than one slave on the same channel.

In the example, you can see that Channel 17 has a problem. Two pairs of DIGRP: Digital Input Group [FC84] blocks (blocks 720/730 and 740/750) have the same slave address 17. 18 and higher are unused, so the solution probably involves configuring the second pair to be the next slave.

```
--Module 31,30,04 Slave Report
Multiple slave FCs on Module 31,30,04 channel 17: see slave report for details.
```



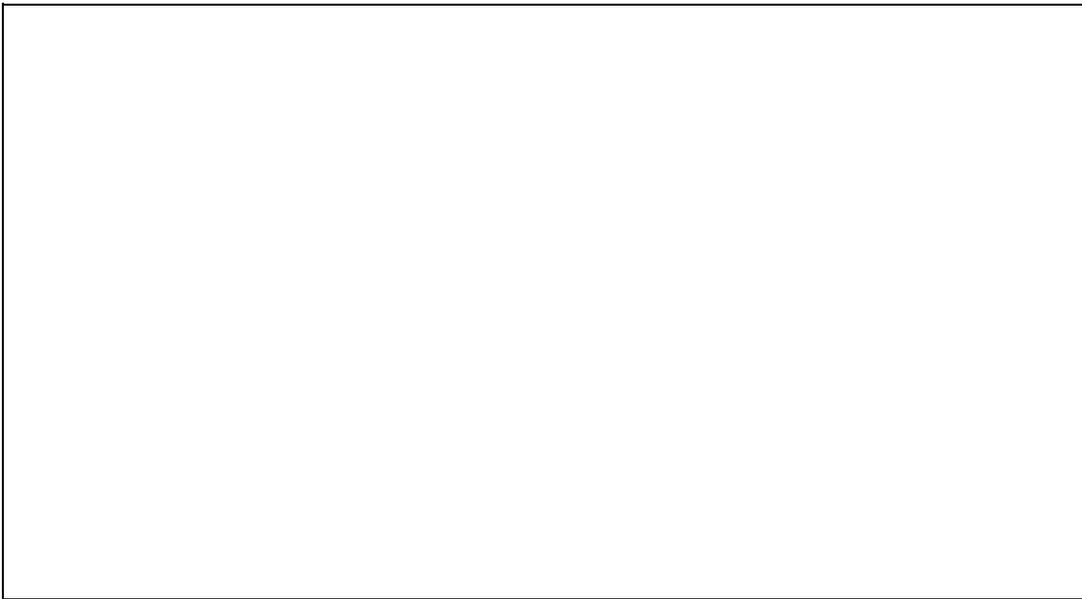
This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SLAVE CONFIGURATION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.169 Multiple slave FCs: Too many FC 132's [ERROR 155]

Multiple slave FCs on Module 1,01,02 slave 10: 4 FC 132's, limit is 3 (Module 1,01,02 Header)

DBDOC's analysis of the configuration of your slave specs suggests that there are too many slaves on an AIS/FBS: Analog Input/Slave [FC132] block.



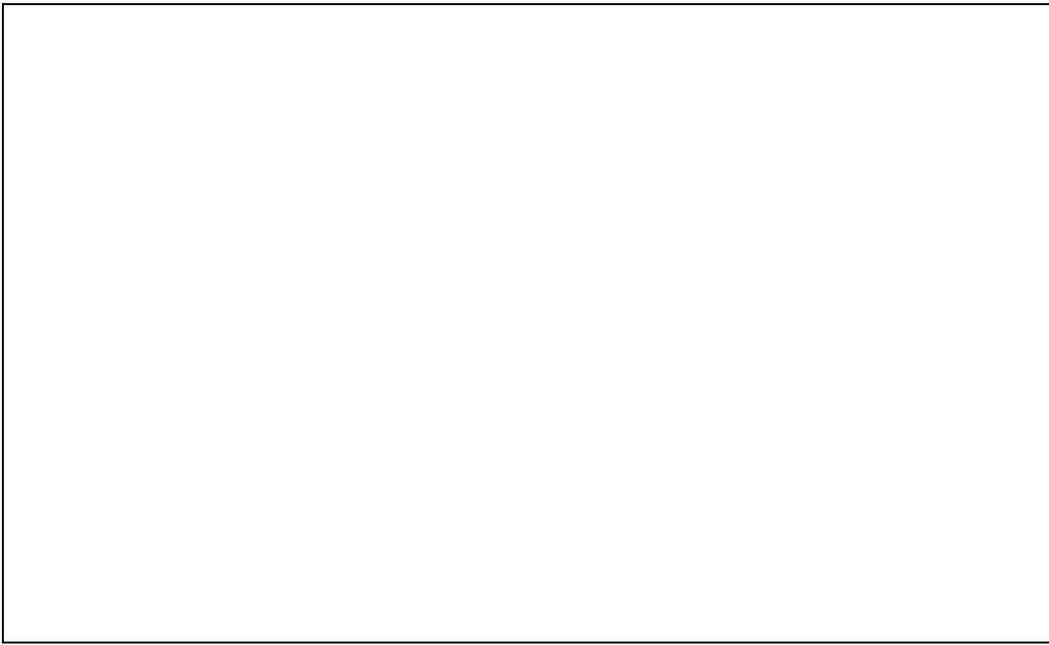
This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SLAVE CONFIGURATION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.170 Multiple slave FCs: Too many FC 149's [ERROR 154]

Multiple slave FCs on Module 1,01,02 slave 15: 3 FC 149's, limit is 2 (Module 1,01,02 Header)

DBDOC's analysis of the configuration of your slave specs suggests that there are too many slaves on an ASO: Analog Output/Slave [FC149] block.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SLAVE CONFIGURATION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.171 Multiple targets [ERROR 328]

```
Shortcut graphic name GraphicNickname has 2 targets: targets
```

Nicknamed vector shortcut has multiple targets.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

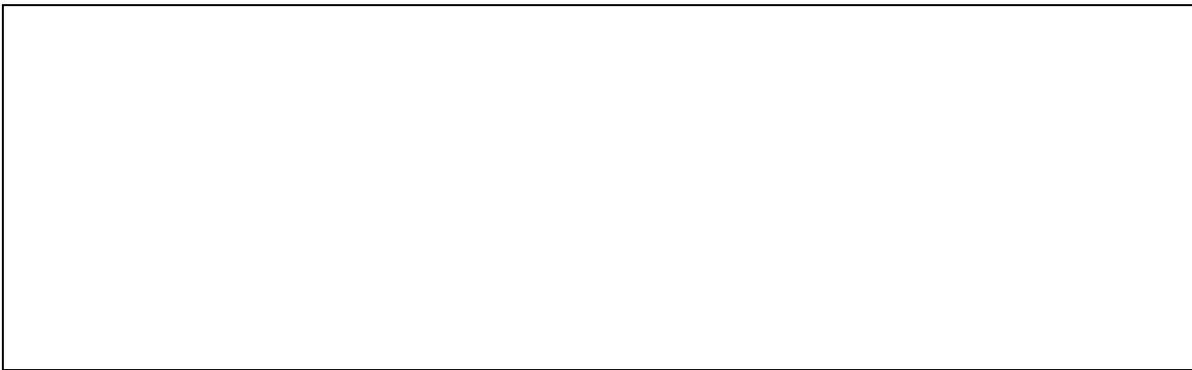
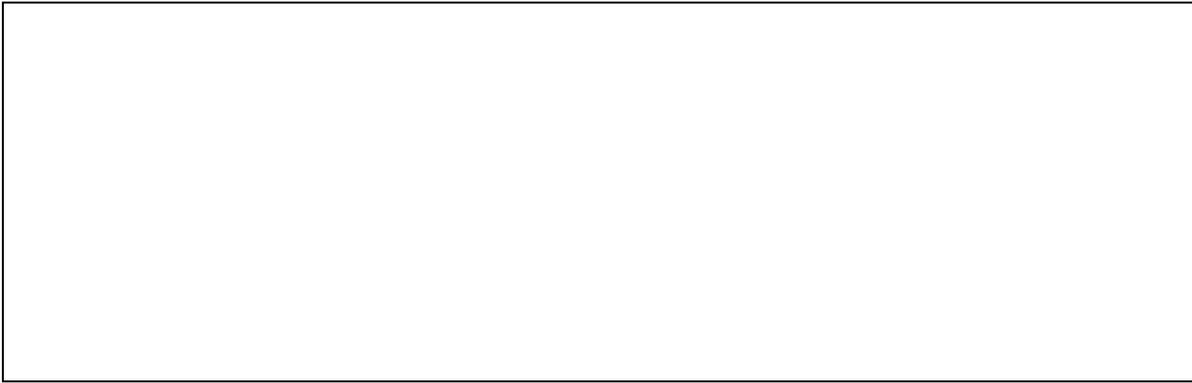
6.6.2.172 Multiple TSTALM [ERROR 084]

```
Multiple TSTALM: Module 1,01,02 Block 1001 is tested multiple times (mode 2)
```

When multiple TSTALM blocks test the same block, it usually indicates an error in cloning the logic. If this is the case then a block that is intended to be tested will have been missed.

Specifically, Spec S1 is can be missed when the block numbers are changed in copied logic.

Multiple TSTALM: Module 1,27,02 Block 21160 is tested multiple times (mode 2)
Module 6,70,03 Block 1529 is tested by multiple TSTALM blocks (mode 2).



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TSTALM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.173 Multiple wires at source: assuming connector [COSMETIC 219]

Multiple wires at a source: assuming a connector at 4400,2700

There are often a few places worth checking with WinCAD. They represent unusual wiring, but we cannot tell if they show a problem. We would welcome feedback on what you find this message to really mean.



`This point has multiple wires in a single location.`

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC MULTIPLE WIRES.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.174 Negative block number [CHECK 037]

`Negative block number Module 1,01,02 Block -5 encountered`

Negative block numbers should not be encountered in your CLD or CAD files, it is an indication of a module that will not compile. If you run into them in a DBDOC build, it means either that you are building a prototype module that is not actually in the process, or that block numbers from a prototype CLD or CAD sheet have been missed.

Example:

```
--Scanning drawing 30500E7  
Negative block number Module 3,05,01 Block -33 encountered
```



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK NEGATIVE BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.175 Negative string length [INFO 10041]

```
Negative string length -5 -- changing to 0
```

The CLD being processed is corrupted. No CAD sheet will be synthesized.

The CLD being processed is corrupted. We only have found this situation in files from one client. In that set of data, 27 CLD files are skipped with the message "Non-local exit called", suggesting that we do not have a good set of CLD files.

The user should examine any CLD with this message using Composer.

No CAD sheet will be synthesized.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.176 Nested inhibit [CHECK 069]

```
Nested inhibit: tag CZRCH100 is inhibited by CXPCH100 but also inhibits 2 other tags: CITCH100, CSTCH100
```

Nested alarm inhibits have been detected. In the database for a graphics system, alarms on a tag are inhibited by one tag, but at the same time the tag is set to inhibit alarms on another tag. Alarm inhibit action may be unpredictable.

Unreliable operation of alarm inhibits is not desirable for the operators.

DBDOC lists up to 10 other tags:

```
--Scanning database file B7_TAGDATA_TAGDATA.DBF  
Nested inhibit: tag GI41Q-9137 is inhibited by HS41Q-8106  
but also inhibits 8 other tags: UI41Q-9118L, UI41Q-9118M,  
UI41Q-9119L, UI41Q-9119M, UI41Q-9120L, UI41Q-9120M, UI41Q-9121L, UI41Q-9121M
```

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK NESTED INHIBIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.177 Never found library file [COSMETIC 011]

```
Never found library file sample.lbr
```

This message says that a CLD/CAD sheet called for a user library that was not found. This is a cosmetic issue because no shapes are actually called from the library.

The message probably appears because the root path for the library location was used.

To make DBDOC use the user libraries located in the modules, start from the Wizard and mark the user library query as omitted. If you do this, then DBDOC uses the same rules as WinCAD – it looks in the module folder for the library requested, moving up the tree to the root. This will catch the libraries where the INFI 90 system compilation finds them.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC LIBRARY NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.178 No SHAPE.LBR was found [BUILD 012]

```
CAD files were specified in configuration, but no SHAPE.LBR was found. All further CAD files will be skipped.
```

This message means CLD/CAD files were specified in the configuration, but no SHAPE.LBR file was found.

To resolve this problem, rebuild your project specifying a SHAPE.LBR file.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.179 No source: Imported block not found [ERROR 110]

```
No source: Module 3,32,04 Block 7630 imported - module built
```

This message shows you that a block is being imported from a module that was built, but the block does not exist in the CAD or CLD files. The CLD/CAD sheets for that module were built, so the block reference appears to be an error.

Example:

No source: Module 1,10,02 Block 1733 imported - module built

If the blocks actually exist, this message tells you that they are not in the files you built with DBDOC. Sometimes, this means the CAD or CLD files are not up to date.



The other possibility is that the module CAD or CLD files are simply a stub and there are no configuration drawings for the blocks. If the blocks actually exist, you can use **BuildPlus/Tools/Project Options/Define Blocks with External Sources** to define them and eliminate the error message.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC BLOCK IMPORTED SOURCE NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.180 No source: Other [ERROR 113]

No source: Other: Module 1,01,02 Block 1001 used - configuration built

A block appears with this message if it appears not to have a source and is not in any of the other documented "source not found" categories.

If you build a project with no configuration (CLD, CAD or LAD files), this message tells you that no checking could be done to verify that blocks used have sources. Tags, for example, cannot be verified if you do not have the configuration built.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC NO SOURCE OTHER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.181 No source: Used in display [ERROR 111]

```
No source: Module 3,32,06 Block 958 (3COS-NOXOBJWT) used in display - module built
```

This message means that two things are true:

- The tag refers to a block that does not exist, but that should have been found in a module that was built.
- The tag is used on at least one graphic.

If there is, in fact, no source for these tags, the operator graphics cannot display a correct value for them.

Example:

```
No source: Module 4,50,02 Block 1703 (CRD-MO-618) used in display - module built
```

In this example, the tag named exists in the database, and was used on a graphic. However, the source block for the tag was not found on CLD or CAD sheets that were built.

The other possibility is that the module CAD or CLD files are simply a stub and there are no configuration drawings for the blocks. If the blocks actually exist, you can use **BuildPlus/Tools/Project Options/Define Blocks with External Sources** to define them and eliminate the error message.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG DISPLAYED SOURCE NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.182 No source: Verify [COSMETIC 112]

```
No source: Verify: Module 1,01,02 Block 1001 used - module built
```

A block appears with this message if it appears not to have a source, and is only in a Verify file. This means that the verification resulted in the block being eliminated.

Note that if the verify file has not been resolved, and you get this message, it means the block does exist.

We build verify files into DBDOC to make it easier for you to do the verifications as the links to the blocks are very useful. However, sometimes the DBDOC build happened after the corrections were made, which means that all the offending blocks have been fixed. If this is the case then the message can be ignored.

In particular, if a block is only mentioned in a verify file, and the block is not in the CLD/CAD sheets (the only way to get this message) it means that the verification resulted in the block being eliminated. Note that if the verify file has not been resolved, and you get this message, it simply tells you what you already know - the block does exist.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC NO SOURCE VERIFY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.183 No tag database specified for console [BUILD 108]

```
No tag database specified for chapter DIRECT.DRS
```

This message indicates that you have not chosen a tag database for a graphics chapter. Without a tag database, neither tag name nor tag index keyed systems will be linked to anything. Your graphics may look okay, but they will not have any links.

Your DBDOC project should be rebuilt to include a database.

Note that DBDOC compiles other files, including .ab, .rtu and .x, that can generate block links for which the source ought to be found.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD NO TAG DATABASE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.184 No valid sections found in XDC file [BUILD 109]

```
No valid sections found in XDC file System.XDC: possibly corrupt/invalid
```

This message indicates that the .XDC file used had no information in it that we could analyze. This can happen when a bad file copy occurs, resulting in a corrupt .XDC file.

It is possible to have a problem in the DBDOC Configuration Wizard where you think you should find an .xdc file, but none is presented to you. If that happens, it means that DBDOC did not accept the structure of the file. Please zip it for us to maintain full information about it, and try to extract another copy. Send both to us

so we can see them and try to explain what has gone wrong.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD NO XDC FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.185 No wired input [COSMETIC 097]

```
No wired input: Module 1,01,02 Block 1001 FC 66
```

This message means that a block has been found that has no input attached to it. You normally can ignore these messages as a cosmetic issue.

We have seen that this condition can be caused in WinCAD and DOS CADEWS by connecting a source to a F(x) block, then deleting the line. The result is that S1 has a value 0. Composer restores the value to 5.

With some older firmware revisions, it is possible that the configuration will compile cleanly, but will not run when it is loaded into the module.

As of "DBDOC 11.2 Feature Update", this message is only generated when the build includes SODG graphics and a trend database.



This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

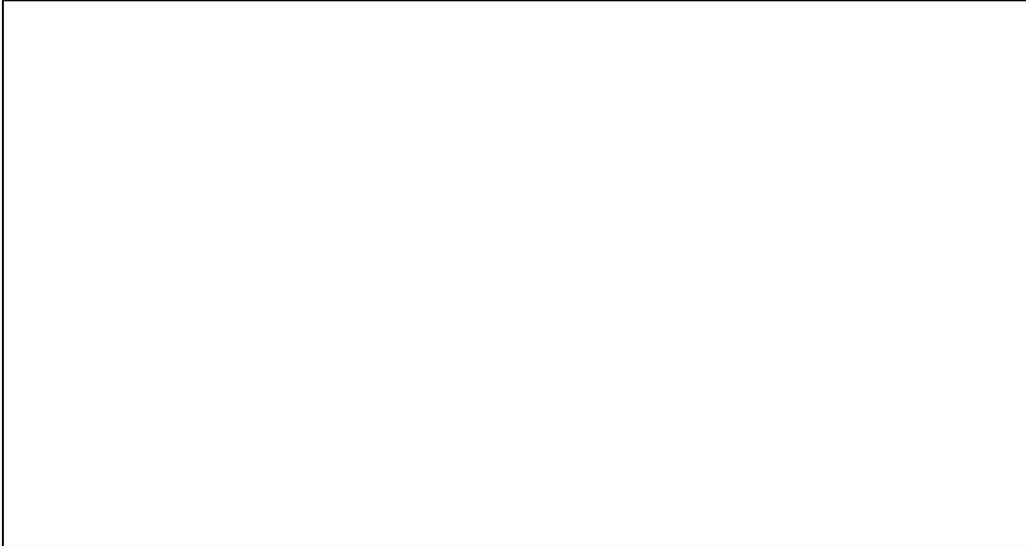
This error type was originally reported in error file **DBDOC-COSMETIC NO WIRED INPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.186 Node out of range [ERROR 165]

Node 256 out of range (0-255) for Module 1,01,02 Block 1234 (FC 37)

This message means that the value of the node is smaller or larger than the allowed value range.

It is possible that a spec has been entered incorrectly.



Node must be between 0-255.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SPEC OUT OF RANGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.187 Non-local exit called -- CLD resolution [INTERNAL 10043]

Non-local exit called -- CLD resolution of EBY1234.CLD

This message means that a file contains information that DBDOC cannot handle.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.188 Non-local exit called at offset [INTERNAL 10042]

```
Non-local exit called at offset 4660
```

This message means that a file contains information that DBDOC cannot handle.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.189 Non-local exit from file [INTERNAL 060]

```
Non-local exit from file sample.txt: error message at offset 0x00000100
```

This message means that a file contains information we cannot handle. Please send us examples with this error.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL NON-LOCAL EXIT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.190 Non-orthogonal rotation [INTERNAL 038]

```
Non-orthogonal rotation: 45 degrees at 2300,1150
```

This message means that the CLD/CAD sheet or graphic contains an element rotated at an angle that is not a multiple of 90 degrees. Presently, we do not handle this. Please send us examples of how you use this feature.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL NON-ORTHOGONAL ROTATION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.191 Non-orthogonal shear rotation [ERROR 327]

Non-orthogonal rotation: Shear(197.4325, 17.43246) degrees at 1827,400 on graphic

This message means that the graphic contains an element rotated at an angle that is not a multiple of 90 degrees. Presently, we do not handle this.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL NON-ORTHOGONAL ROTATION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.192 Not enough inputs [ERROR 133]

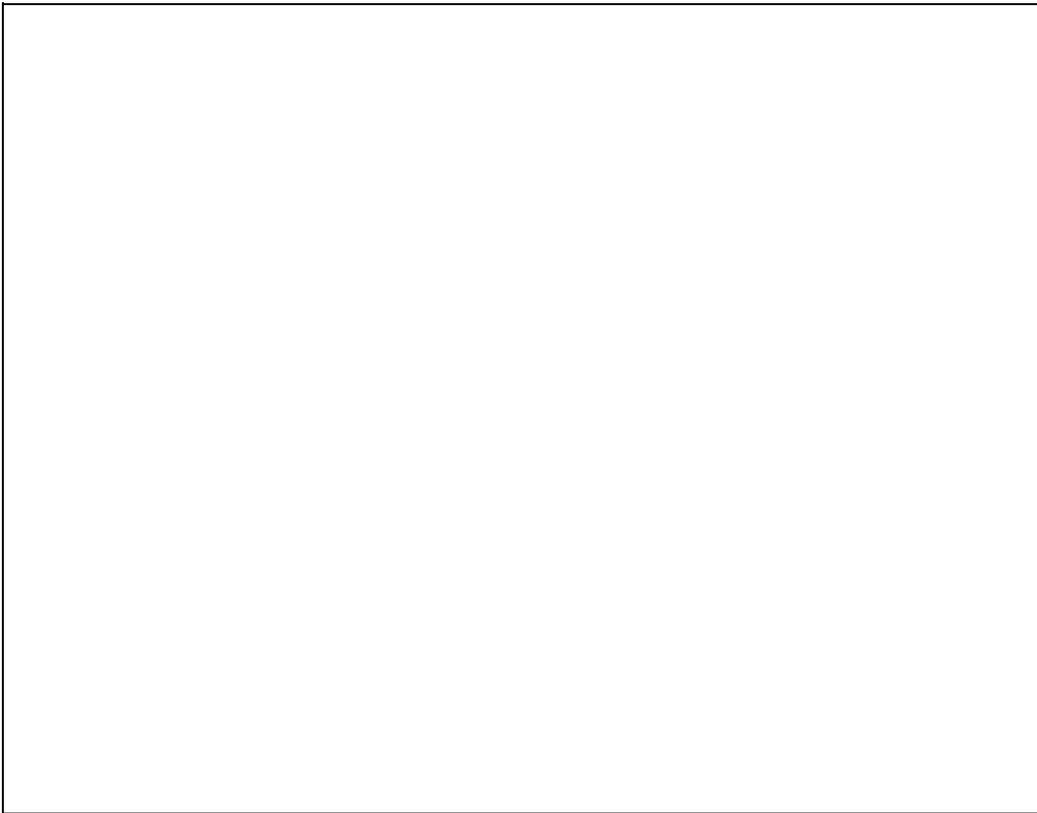
Insufficient number of inputs (5 wired, 8 required) in FC 36 Module 1,01,02 Block 1001

The function block requires more inputs than have been provided. NOTE: DO NOT SET S9 to 0. If you do this, the output will be set to 1.

A usual example is of a QOR: Qualified OR (8-Input) [FC36] block used as an AND operator, with S9 greater than the number of inputs.

```
--Module 4,10,04 QOR Report
```

```
Insufficient number of inputs (2 wired, 3 required) in FC 36 Module 4,10,04 Block 1344.
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TOO FEW INPUTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.193 Only handling first tag [CHECK 048]

Only handling first tag in expression: "PU901CDS_AS.C_APV + PU902CDS_AS.C_APV"

The graphical expression only has a link generated for the first tag in the expression. If there are other tags, there is no link to them.

Example:

```
Only handling first tag in expression: "PU901CDS_AS.C_APV + PU902CDS_AS.C_APV"
```

For this example, you have a link for PU901CDS_AS, but none for the second tag, PU902CDS_AS.

DBDOC does not evaluate graphical expressions in Conductor NT and PPB. The message allows you to make sure we have not missed any errors.

```
Scanning graphic 18_DECARBONATOR.M1 (number 1801)
```

```
Only handling first tag in expression: "19_HS_2105_RMC.EB_FB2_T == 19_HS_2105_RMC.EB_FB2LS1_T".
```

```
Only handling first tag in expression: "19_HS_2105_RMC.EB_FB2_T == 19_HS_2105_RMC.EB_FB2LS1_T".
```

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK GRAPHICAL EXPRESSION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

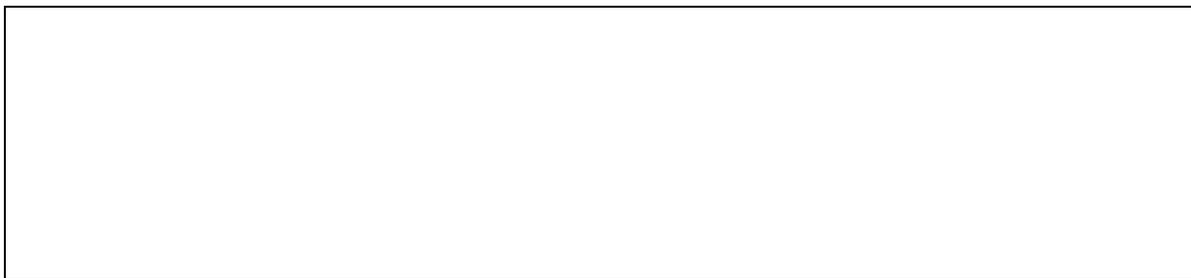
6.6.2.194 OREFs differ in case only [CHECK 185]

The following OREFs differ in case only: da4ao498,DA4A0498

The IREFs and OREFs reported in this message do not work in DBDOC. While they would work with WinCAD they would not work with Composer. We suggest modifying these OREFs and IREFs to have the same case, using the DBDOC report as a guide. This way you do not leave a pitfall for yourself or someone else if a switch to Composer occurs.

Example:

```
--Module 1,04,02 OREF Report  
The following OREFs differ in case only: da4ao498,DA4A0498
```



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK CASE DIFFERENCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.195 Out of expected range [ERROR 10026]

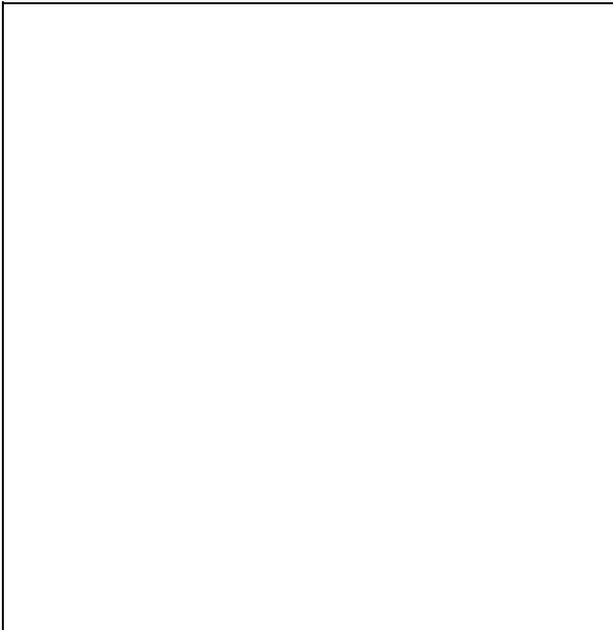
Out of expected range -360..360: Module 1,01,02 Block 12345 (FC23 S3) integer value 720

This message means that a CLD file has a value for a specification that is apparently illegal. We have found this as uncorrected damage from early versions of Composer after the verify with update feature was used.

Most of these messages occur when you switch to Composer version 2.1. They can be ignored for the most part.

```
--Parsing file E:\PROJECT\ANC_CO~2\EBY0884.CLD...  
Block 9399 S1 integer value 65534 (real value 0.000000) too big for short.
```

From the picture, you can see that somebody must have put 0.0 in S1 of block 9399 (a real number) instead of 0 (an integer). A real number does not have the same structure as an integer, so DBDOC tells you about it. We do not know if it works, but some errors of this type have found actual problems for clients.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.196 Outdated file format [INTERNAL 206]

```
Outdated file format for FCConstraints.txt - has obsolete format: obsolete = 1
```

This message means that the DBDOC file is in an outdated file format. Contact us to get a new file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.197 Override on input [INFO 122]

```
Override on input LABEL, Module 1,01,02 Block 1001: value forced to 1
```

In a ladder system, this message shows that the configuration loads the variable shown as being in override. This may not be a problem, but gives you a checklist.

In a CAD or Composer system, this message is generated by our rung block processing. You will want to verify that the logic makes sense.

Example:

```
Override on input S10, Module 1,11,02 Block 6265: value forced to 1
```

This message tells you that a rung block input has been overridden to make its value 0 or 1. This is not an error, but a checklist. The people responsible for the plant usually want to keep track of all overrides. This report exists for that purpose, so the overrides are not forgotten and invisible.

You can see from the example that this can just be a simple means to avoid setting the controlling spec to the correct value. In the example, specifications 11 through 21 should be set to 000, which would disable the unwired inputs.



Note how Hyperview shows overridden inputs with the value they are set to. In this case, the unwired inputs are set to "1", so the AND chain will work.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO OVERRIDE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.198 Override on output: forced to a constant [INFO 151]

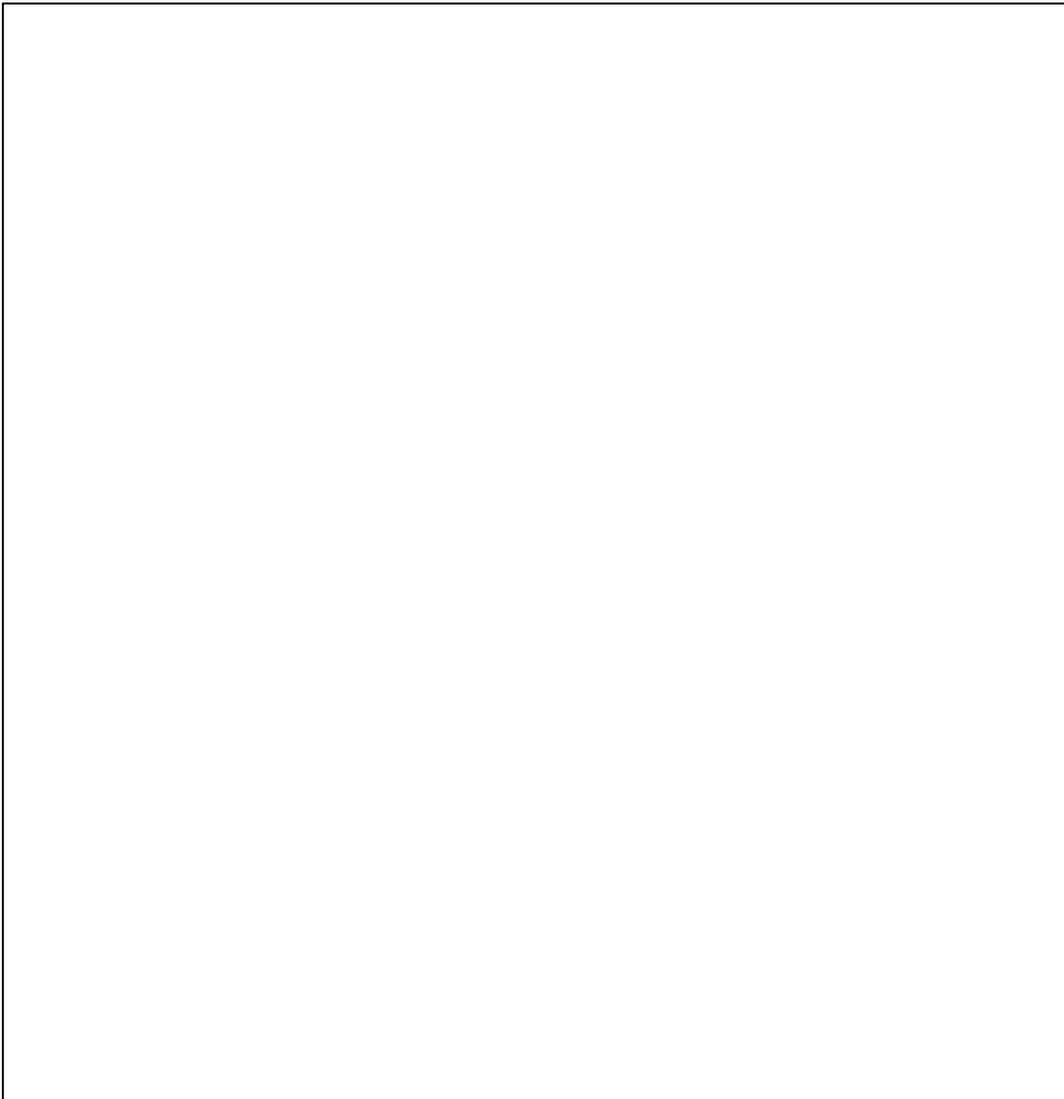
```
Override on output: ladder rung Module 1,01,02 Block 1001 output is forced to 1
```

The indicated rung block has its output forced to 0 or 1. The logic will not respond to changes in the inputs.

This is not an error, but a checklist. The people responsible for the plant usually want to keep track of all overrides.

Example:

```
--Post Processing drawing 1020881C.CAD  
Override on output: ladder rung Module 1,42,02 Block 6224 output is forced to 0.
```



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO OVERRIDE ON OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.199 Override on output: held [INFO 150]

Override on output: ladder rung Module 1,01,02 Block 1001 output is held

The indicated rung block has its output forced to hold its value. DBDOC indicates this as an H for hold. The logic will not respond to changes in the inputs.

This is not an error, but a checklist. The people responsible for the plant usually want to keep track of all overrides.



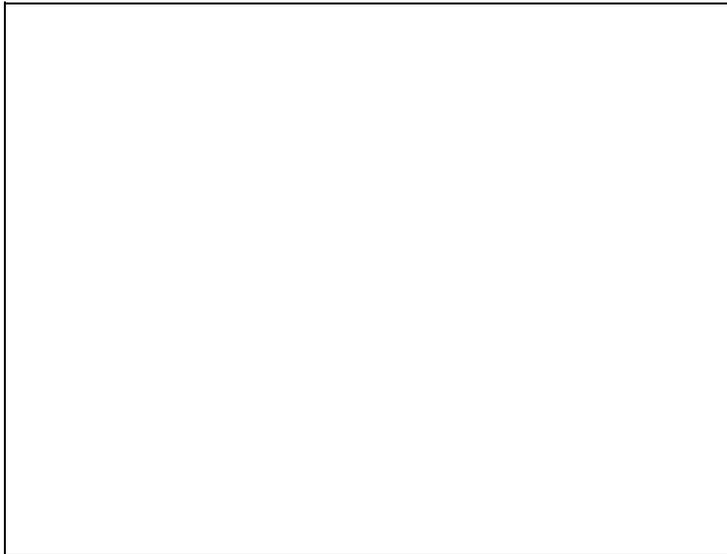
This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO OVERRIDE ON OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.200 Override on output: invalid code [INFO 152]

Override on output: ladder rung Module 1,01,02 Block 1001 output has invalid code 4

The indicated rung block has its output forced to an invalid code. The logic will not respond to changes in the inputs.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO OVERRIDE ON OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.201 Parse error in file [INTERNAL 115]

```
Parse error in file attributes.txt, reverting to static
```

This means there is a problem with one of DBDOC's text files. You will need to contact us to get a new version of the file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL BAD DBDOC FILE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.202 PDF security prevents embedding [BUILD 324]

```
Cannot embed PDF file due to PDF security settings: sample.pdf
```

The security settings in the PDF file prevent DBDOC from embedding it in your system snapshot. You can still add it to your project as a linked/external PDF file.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.203 PID block uses derivative with PV from discontinuous input [ERROR 307]

```
PID Module 1,01,02 Block 1002 uses derivative with PV from discontinuous input
Module 1,01,02 Block 1003 PID Module 4,02,02 Block 1425 uses derivative with PV
from discontinuous input Module 4,02,02 Block 6188. Intermediate blocks: Module
4,02,02 Block 1403 [40202G6]
```

This PID (or APID) block has a non-zero derivative spec and input from a discontinuous source.

These messages are generated if FC18 or FC19 have S8 adapted or not equal to 0, or FC 156 has S14 adapted or not equal to 0, and tracing back the source of S1 through up to 6 intermediate blocks, there are potential discontinuous sources such as FC26, FC47, FC93, FC114, FC121, FC137, FC165, or FC178.

PID and APID algorithms should never use a derivative term if the PV is affected by an exception-reported input PV.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INFO PID GAIN-SPAN.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.204 PID Controller has all active tuning constants zero [ERROR 173]

```
PID Controller Module 1,01,02 Block 1001 has all active tuning constants zero
```

A controller has its gain constants all zero, which will force the output to be 0.0.

- APID: Advanced PID Controller [FC156]
 - ◆ S11 is the overall gain multiplier K and must never be 0.
 - ◆ S12, S13 and S14 are the KP, KI and KD constants and must never all be 0 (unless one or more of these specs are adapted, in which case there will be no message).
 - ◆ S12 must never be 0 if the algorithm is "classical", not "noninteracting", that is, if S18 ends in 0 or 2.
- PID: PID Error Input [FC18] and DEL PID: PID Control (PV and SP) [FC19]
 - ◆ S5 is the overall gain multiplier K and must never be 0.
 - ◆ S6, S7 and S8 are the KP, KI and KD constants and must never all be 0 (unless one or more of these specs are adapted, in which case there will be no message).

The effect of either of the above sets of conditions is that the output is 0.00.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TUNING CONSTANTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.205 PID Controller has primary gain spec zero [ERROR 172]

PID Controller Module 4,14,03 Block 4222 has primary gain spec S12 zero

A controller has its overall gain spec set to zero, which will force the output to be 0.0 unless it is tracking.

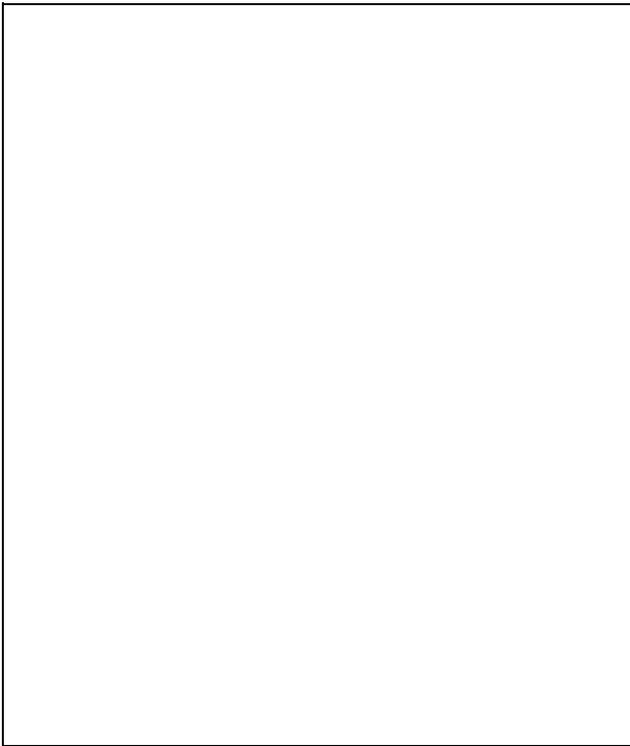
- APID: Advanced PID Controller [FC156]
 - ◆ S11 is the overall gain multiplier K and must never be 0.
 - ◆ S12, S13 and S14 are the KP, KI and KD constants and must never all be 0 (unless one or more of these specs are adapted, in which case there will be no message).
 - ◆ S12 must never be 0 if the algorithm is "classical", not "noninteracting", that is, if S18 is 0, 2, 10 or 12.
- PID: PID Error Input [FC18] and DEL PID: PID Control (PV and SP) [FC19]
 - ◆ S5 is the overall gain multiplier K and must never be 0.
 - ◆ S6, S7 and S8 are the KP, KI and KD constants and must never all be 0 (unless one or more of these specs are adapted, in which case there will be no message).

The effect of either of the above sets of conditions is that the output is 0.00.

Note that the output is not 0.0 under the following conditions for FC 156 (APID).

- S4 is wired (this is the TR tracking input)
- S5 is wired and has a value 1 (this is the TS track switch input)

The image shows live data when tracking is switched on.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

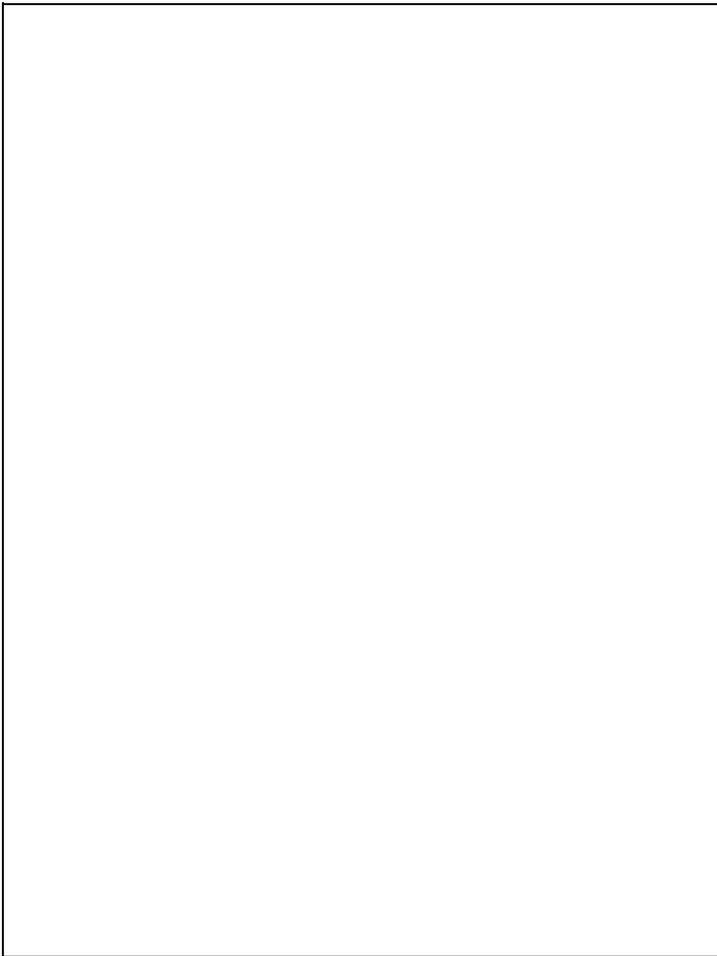
This error type was originally reported in error file **DBDOC TUNING CONSTANTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.206 PID lag spec zero [CHECK 177]

```
PID Controller Module 1,01,02 Block 1001 has derivative lag spec S15 zero --  
derivative component will be ignored
```

This message means that a PID Controller block has its derivative lag spec S15 set to zero. The derivative term will be applied directly.

If S15 is 0.0, the value 1.0 will be used internally. Also if S14 is 0.0, S15 is ignored.



With S15 as 0.0, the value of S14 is divided by 1, and the derivative is applied all at once. Therefore There is no lag in effect.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.207 Point has no description [COSMETIC 251]

Point EXPANA1 has no description

The referenced tag is in the Honeywell PNT database only. There is no corresponding tag in the INFI 90 database. The point has an empty description field.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC MISSING PNT TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.208 Possibly corrupt GPI mdb File [ERROR 315]

```
Possibly corrupt GPI file 12405_CORMON.mdb -- skipping
```

The .mdb file does not contain a table with a BlockNumber or Tag column.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.209 Possibly corrupt HGS mdb File [CHECK 320]

```
Possibly corrupt HGS file 12405_CORMON.mdb -- skipping
```

The .mdb file does not contain a table with a BlockNumber or Tag column.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.210 Possibly corrupt HPG mdb File [CHECK 321]

```
Possibly corrupt HPG file 12405_CORMON.mdb -- skipping
```

The .mdb file does not contain a table with a BlockNumber or Tag column.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.211 Problem communicating with dongle [INTERNAL 045]

Repeated problems communicating with hardware key (dongle)

DBDOC has encountered repeated problems communicating with hardware key (dongle).

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

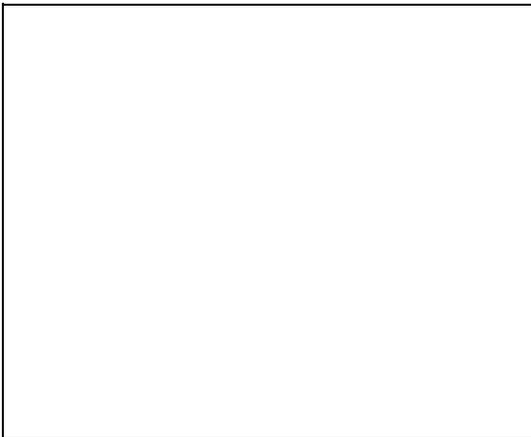
This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.212 QOR forced to 1 [INFO 135]

QOR output forced to 1 (5 wired, 8 required) in FC 36 Module 1,01,02 Block 1001

A QOR output has been forced to 1. The message is intended to encourage documentation of forced logic.

A QOR block functions in such a way that if the value of S10 is 0, then the output is 1 if the number of logic 1 inputs is greater than or equal to the value of S9. With the value of S9 at 0 as well, the number of input 1's will always be greater than or equal to 0. therefore the output value of the block will always be 1.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO QOR OUTPUT FORCED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

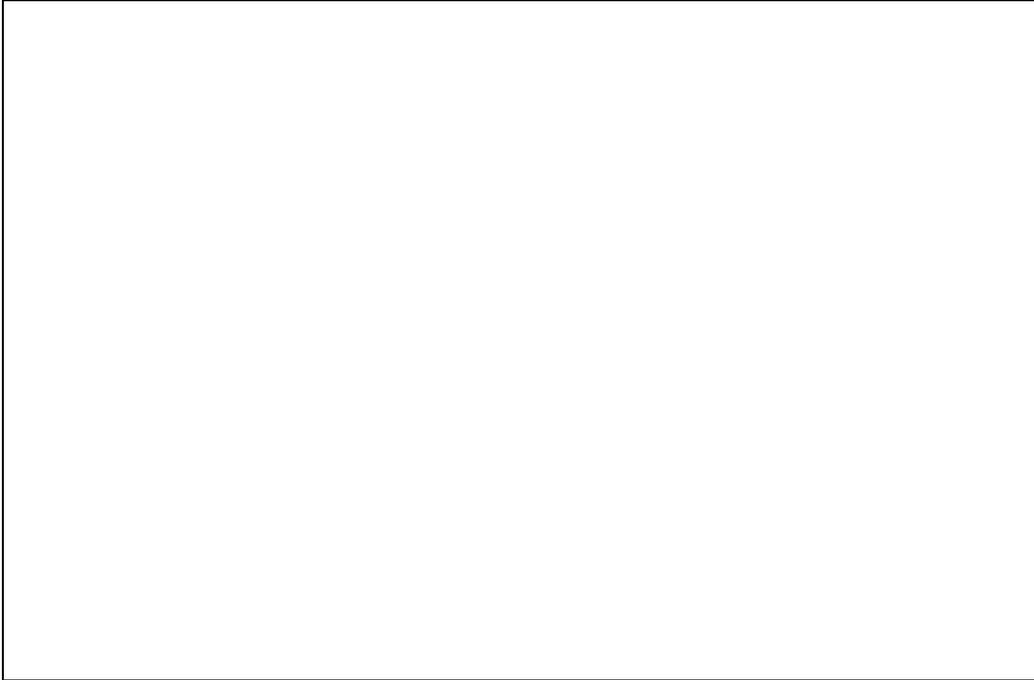
6.6.2.213 QOR used as a NOR [INFO 134]

QOR being used as a NOR: possibly unintended in FC 36 Module 1,01,02 Block 1001

This QOR gives a logical 1 only if all inputs are 0. This might be unintended.

The DBDOC build has found logic that is unexpected.

QOR being used as a NOR: possibly unintended in FC 36 Module 1,11,02 Block 2999.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO QOR USED AS NOR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.214 Record failed to find matching format [INTERNAL 210]

Record 26: Failed to find matching format for 30FQI170RS, defaulting to first format

This message means we have had some failure in creating a format template for the tag named (this relates to the .fmt files created by DBDOC in the project folder). Please call this to our attention.

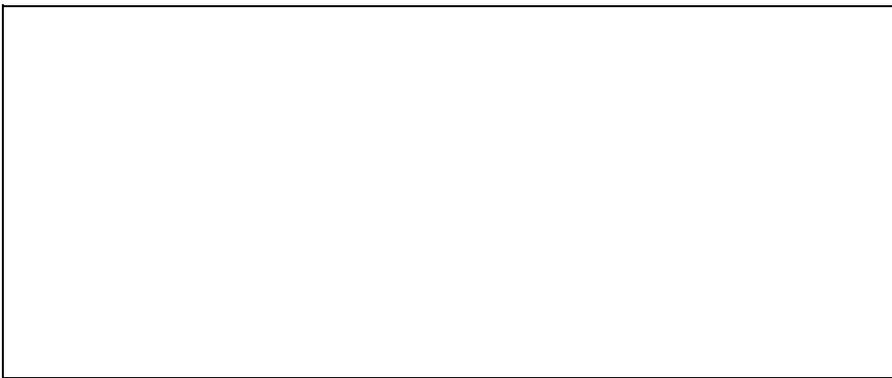
This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL MISSING FMT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.215 REDAI both inputs same block [CHECK 072]

```
REDAI Module 20,03,02 Block 4673 with both inputs from same block Module  
20,03,02 Block 452 cannot synthesize quality
```

When a REDAI block has both inputs from same block, the block does nothing because the output tracks the input value and quality. If the input has quality, so will the output. However, if the input does not have quality, the output will never be changed to bad quality, so testing it is misleading. The specifications allow bad quality to be introduced on detection of a rate of change problem. If both S5 and S6 are reasonable, the quality will be set bad if the rate of change (units per second) of the inputs exceeds the sum of S5 and S6 (the deadband parameter). Clearly, the deadband should be less than the rate of change limit. If you see this message with what looks like a valid value of S5, you will find that S6 is not a valid deadband value.



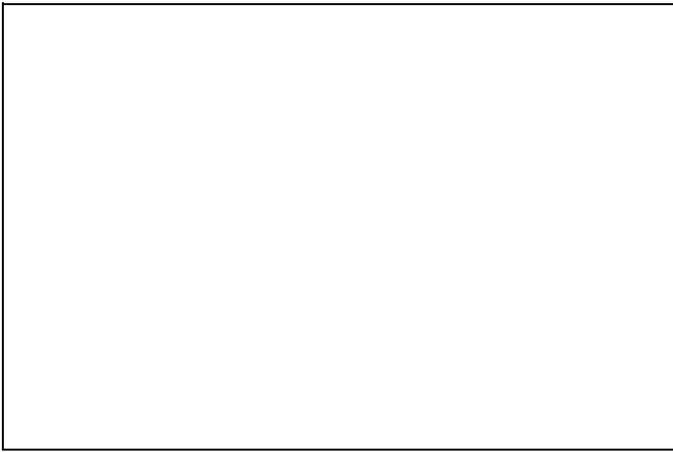
This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.216 REDAI deadband exceeds rate limit [CHECK 075]

```
REDAI Module 1,01,02 Block 1000 has deadband S6 larger than rate limit S5
```

If the deadband exceeds the rate limit, the change to bad quality can not be reversed reasonably.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.217 REDAI invalid deviation S4 [CHECK 074]

```
REDAI Module 1,01,02 Block 1000 has invalid deviation limit S4
```

The REDAI deviation limits may not be 0.0 or a negative value.

S4 is the deviation limit, which has the same units as S1 and S2.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.218 REDAI quality is never tested [CHECK 175]

REDAI Module 1,01,02 Block 1000 reports or synthesizes quality but quality is never tested

This message is most important when the quality synthesized is not ever used in a tag that is displayed. Bad quality may be alarmed, but it has no direct effect in the configuration.

DBDOC does not necessarily track the quality all the way to its use, so it is possible that the block is eventually wired into a test quality block. This can be checked relatively easily by following the path to the places the quality is eventually lost. If a test quality block is found somewhere along the path, then this error is false, and we would recommend hiding it in the error browser.

On the other hand, it is possible that the quality is lost. This can lead to very wrong data being used in your system, so it is definitely worth looking into.



In this case, the REDAI block gives quality to the output value, and as it is entered into the F(X) block, it loses that quality.

This causes the values to be very wrong.

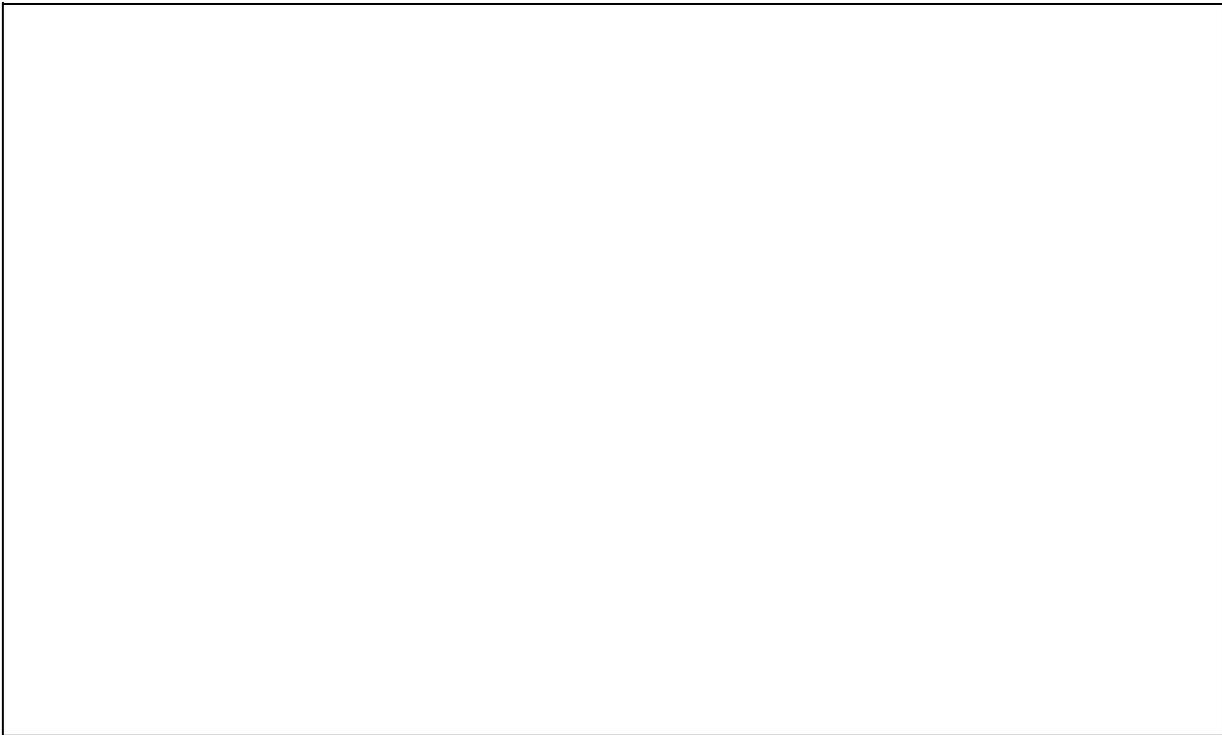
This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.219 REDAI select input hardwired [CHECK 073]

```
REDAI Module 1,01,02 Block 1000 has select input S3 wired from constant block  
Module 1,01,02 Block 1001
```

The select input is designed to switch between two inputs. When it is wired to a constant, no switching is possible. This is why DBDOC draws this to your attention.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

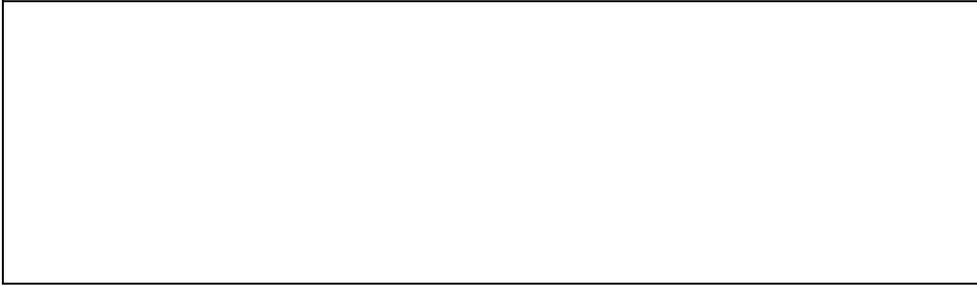
This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.220 REDAI select input wired but ignored [CHECK 079]

```
REDAI Module 1,01,02 Block 1000 has select input S3 wired from Module 1,01,02  
Block 1001 but ignored
```

This means that the only effect of the block might be to switch an unconnect (0) value into the output.

A REDAI: Redundant Analog Input [FC96] block switches between S1 and S2 values based on the value of the S3 input. When S1 and S2 come from the same block, however, this block has no effect. As this block is doing nothing, it could be made spare.



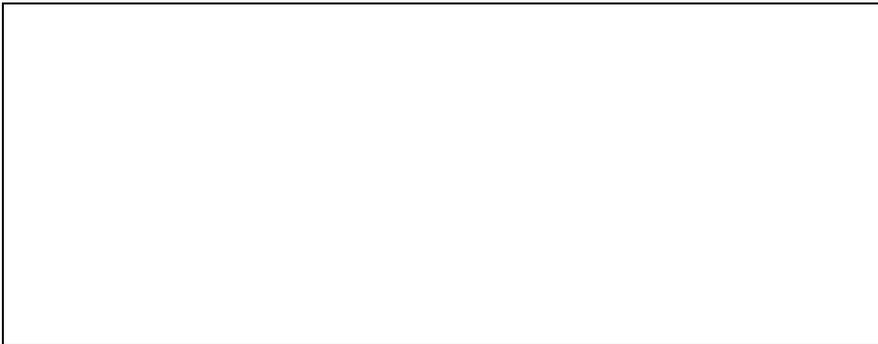
This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.221 REDDI both inputs from same block [CHECK 076]

```
REDDI Module 20,09,02 Block 2462 has both inputs from same block Module 20,09,02  
Block 1525
```

A REDDI block that has both inputs from same block does nothing.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.222 REDDI select input hardwired [CHECK 077]

```
REDDI Module 1,01,02 Block 1000 has select input S3 wired from constant block  
Module 1,01,02 Block 1001
```

The select input is designed to switch between two inputs. When it is wired to a constant, no switching is possible. This is why DBDOC draws this to your attention.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK REDAI REDDI.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.223 Referenced M1 display file not found [CHECK 194]

```
M1 display file TR101 (ref'd as graphic:8101) not found (not configured) in  
TLINK
```

The graphic calls for another graphic which was not found. This is not really an error, but is a weakness for the operators using the graphic system.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK MISSING DISPLAYS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.224 Regex compilation error [INTERNAL 039]

```
Regex compilation error: Compilation failed
```

DBDOC encountered a regex compilation error - the compilation failed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.225 Regex execution error [INTERNAL 040]

```
Regex execution error: Execution failed
```

DBDOC has encountered a regex execution error - the execution failed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

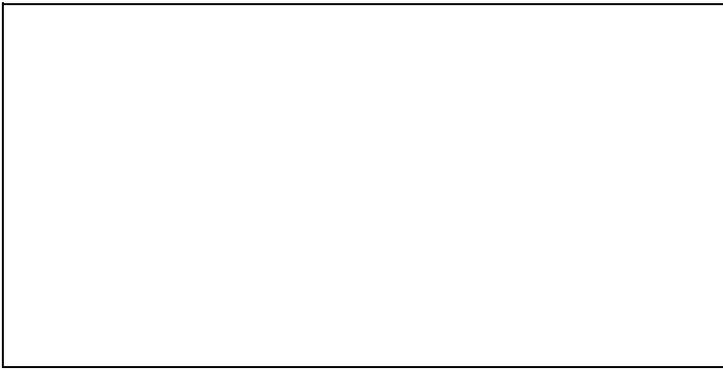
6.6.2.226 REMSET limits out of order [ERROR 261]

```
REMSET limits out of order: Module 1,40,31 Block 24520 S3 low limit 1.0 greater than S2 high limit value 20.0 (adapted)
```

This message means that the REMSET: Remote Manual Set Constant [FC68] block has an S3 low limit greater than the S2 high limit value.

This may indicate a problem.

In the image below, the S3 value greater than an S2 value. Because of this, the range of acceptable values does not exist.



This block allows an operator to enter a constant to the control scheme. S2 and S3 are the limits of the range of values the operator can enter.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC REMSET LIMITS OUT OF ORDER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.227 Requested color index out of range [INTERNAL 193]

```
Requested color index Test Font out of range (0 - 255)Test
```

This message means that a file appears to use a color or font index outside the range that we know how to handle. Note that a strange number might also mean that you have a version of graphics that we need to learn more about.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL REQUESTED INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

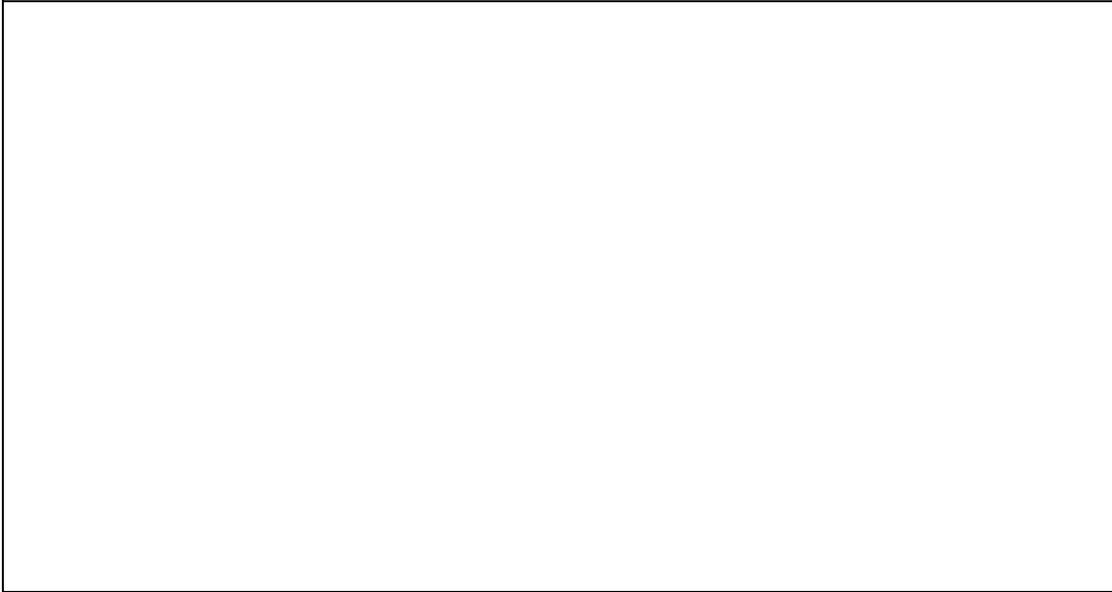
6.6.2.228 RESTR block order wrong [ERROR 065]

```
RESTR block Module 1,04,03 Block 159 not greater than input Module 1,04,03 Block 516
```

A RESTR: Restore [FC140] block that is out of order will save the previous value of the block it supports. This is an invalid practice and can be an error.

The restore block must be located at a block number greater than the function block being restored (S1).

The restore function code saves and restores values. However, the restore block must come after (be a higher block number) than the block being restored. If it precedes the block it is told to backup, the value restored will be that from the last cycle. This will cause all restored values to be false.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RESTORE BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.229 RESTR block will never save [ERROR 309]

```
RESTR block Module 1,01,02 Block 1000 will *never* save because S2 is unwired  
and S3 is wired to constant 0
```

This RESTR has one or both of S2 and S3 unwired or wired to constant logic 0. This block will never save

A RESTR block will perform the save operation if, and only if, there is a logic 1 on both S2 **and** S3. If this never happens then when the restore operation is performed, the value restored will likely not be useful.

For best results, logic should be used to set S2 and S3 at times appropriate for a save operation. This might be on a time schedule, or when the value to be saved has changed (or changed enough) to make it useful to save.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RESTORE BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.230 RESTR block will update to NVRAM every cycle [INFO 308]

```
RESTR block Module 1,01,02 Block 1000 will update to NVRAM every cycle
```

This RESTR has S2 and S3 both wired to constant logic 1, so that it will update every cycle.

Depending on what function code a RESTR block is saving, and on the hardware you are using, a single RESTR block can take up to a few milliseconds to execute. If **all** of your RESTR blocks execute every cycle, you may experience poor performance. The worst example we have seen had a set of RESTR blocks that took 480 milliseconds to execute, but they were being run on a 250 millisecond cycle. As a result, the cycle was not completing before being run again.

It will be better to use logic to stagger the execution of your RESTR blocks. The usual method would be to execute the RESTR block only when you know that the value has changed, or has changed enough for you to care about.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.231 Rung block did not use all inputs to compute output [ERROR 127]

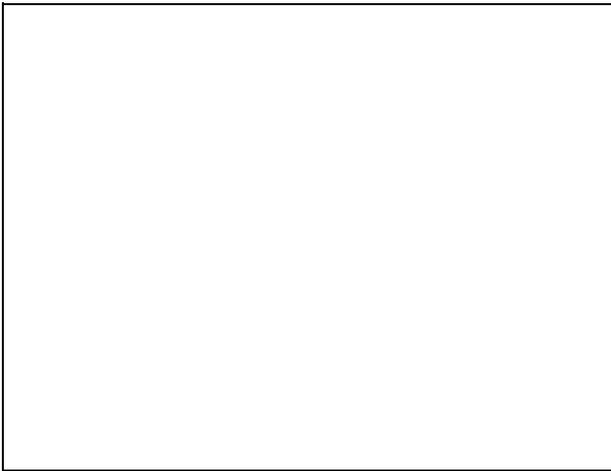
```
Rung block Module 1,01,02 Block 1001 did not use all inputs to compute output
```

Rung blocks can have parameters that appear incorrect. The logic implied by the specifications may appear to be in error.

Example:

```
--Post Processing drawing 1010274C.CAD  
Rung block Module 1,42,02 Block 3093 did not use all inputs to compute output.
```

This message means that the calculations have not been specified properly. A partial result has been created, but not used.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RUNG BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.232 Rung block spec is unwired but used [ERROR 125]

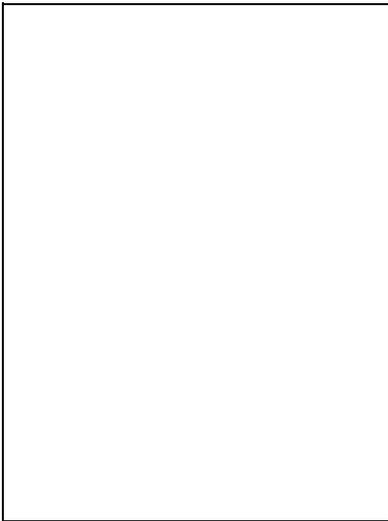
```
Rung block Module 1,01,02 Block 1001 S3 is unwired but used by S8 value 031
```

Rung blocks can have parameters that appear incorrect. Unwired inputs may have non-zero parameter values, as if they were to be used. The logic implied by the specifications may appear to be in error.

Messages like these can often be ignored. They usually mean that the person doing the work left inputs drawn in, but unused, but sometimes it means there are controlling specs operating on empty inputs.

For example,

```
--Post Processing drawing 14004Q4  
Rung block Module 10,40,04 Block 8510 S18 is unwired but used by S8 value 022
```



When a block has both this error 125 and "Error 124: Rung block spec is wired but unused", it usually means the source was connected to the wrong input, or the specification values need to be changed:

```
--Post Processing drawing 10902D2  
Rung block Module 20,09,02 Block 5057 S7 is unwired but used by S2 value 010  
Rung block Module 20,09,02 Block 5057 S8 is wired (block 1055) but unused by S3 value 000
```



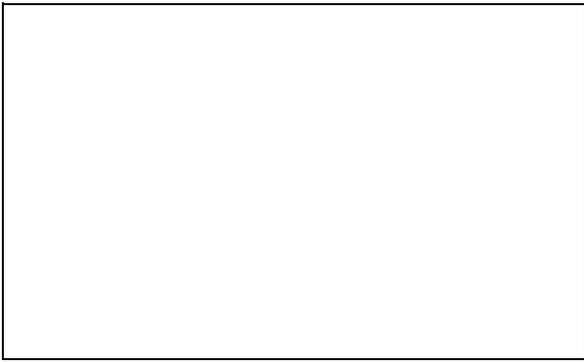
This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RUNG BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.233 Rung block spec is wired but unused [ERROR 124]

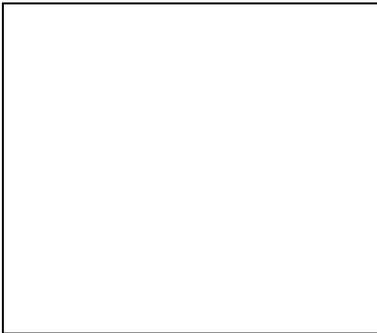
```
Rung block Module 1,01,02 Block 1001 S3 is wired (block 1234) but unused by S8  
value 032
```

These are always worth checking, as they mean that the input looks like it is used, but it is not. This may be what is intended, in which case the line should probably be deleted. Otherwise, it could show you a potential problem. In the case of block 0, a logical 0 constant is wired into the rung block but it is not needed because the controlling specification is 0. If you tidy this up, the CLD/CAD sheets will be easier to understand.



When a block has both this error 125 and "Error 125: Rung block spec is unwired but used", it usually means the source was connected to the wrong input, or the specification values need to be changed:

```
--Post Processing drawing 10902D2  
Rung block Module 20,09,02 Block 5057 S7 is unwired but used by S2 value 010  
Rung block Module 20,09,02 Block 5057 S8 is wired (block 1055) but unused by S3 value 000
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RUNG BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.234 Rung block spec is wired from constant [ERROR 126]

```
Rung block Module 1,01,02 Block 1001 S3 is wired from constant block 1
```

This message is a warning about unusual logic.

Rung blocks can have parameters that appear incorrect. Wired inputs may have controlling parameter values of 0, so they are unused. The logic implied by the specifications may appear to be in error.

We have not found any instances of this error in any of our test builds. If you should get this error, please contact GMCL. This will allow us to test the error in greater detail, and help identify the problems you may be having. Thank you for your patience.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RUNG BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.235 Rung stack not initialized [ERROR 306]

```
Rung block Module 3,14,15 Block 926 logic invalid because S2 is an OR operation
```

A Rung Block stack was not initialized before the first OR operation

GMCL has determined that the execution of each Rung Block starts with a 1 on the stack. If the first operation is an OR, the result of the operation is always 1, regardless of the input to the OR operation. The first operation should never be OR. Use PUT (xx0) instead.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RUNG BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.236 Same-bus input cannot read value in other loop [ERROR 10016]

```
Same-bus input Module 1,01,02 Block 12455 (FC 45) attempts to read value in  
other loop 3
```

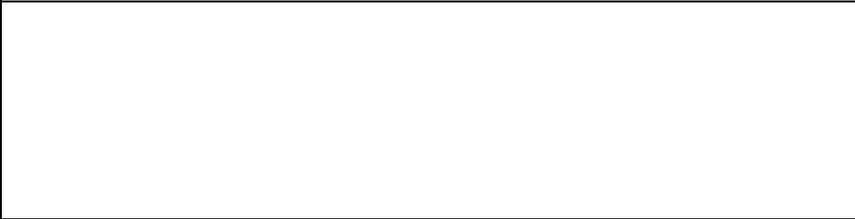
This message tells you that an IREF (Input Reference) exists in the Composer project. However, the function block cannot import it, because it is outside the scope of what it can import.

For example, if an AI/L or DI/L import block has an IREF that exists only in another Loop, you will see this message. The reference cannot tell what is being imported, because these functions cannot import from another loop, only the same one.

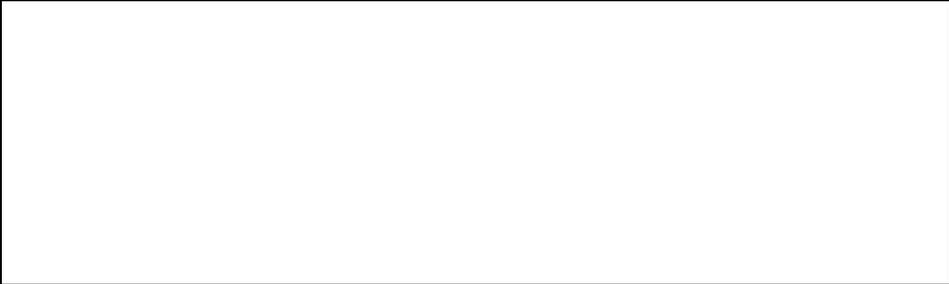
```
--Parsing file H:\...\EBY45EB.CLD... CAD Names: 2300251A.CAD  
External output reference is unreachable for IREF **1AOIK CB CLSD on Module 2,30,02 Block 1652:  
cannot import from Loop 2 PCU 20.
```



Here is the actual source, as determined by the specifications.

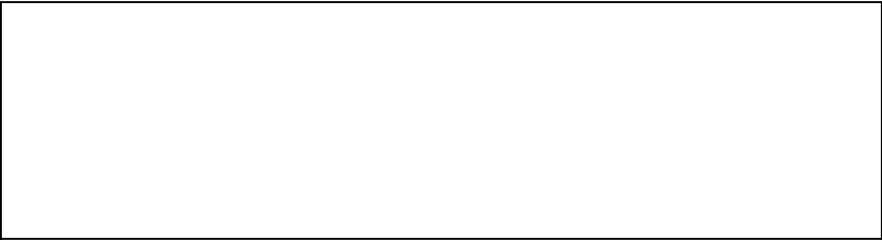


Here is the source that the input reference indicates.



```
--Parsing file H:\...\EBY3DAC.CLD... CAD Names: 2320454A.CAD  
External output reference is unreachable for IREF L30D_MSG on Module
```

2,32,04 Block 1584: cannot import from Loop 2 PCU 22."



Here is the block that is being imported.



Here is the source indicated by the input reference L30D_MSG.



It is possible to see how Composer, in being robust enough to import files from WinCAD, also allows errors to persist in the input references.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNREACHABLE OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.237 Same-bus input cannot read value in other PCU [ERROR 10017]

```
Same-bus input Module 1,01,02 Block 12455 (FC 45) attempts to read value in  
other PCU 3
```

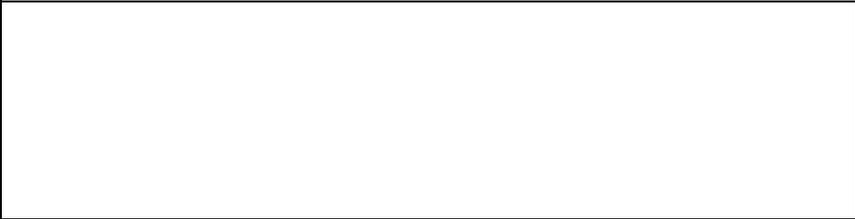
This message tells you that an IREF (Input Reference) exists in the Composer project. However, the function block cannot import it, because it is outside the scope of what it can import.

For example, if a Bus Import (AI/B, DI/B, AIL/B or DIL/B function) has a reference in a different PCU, you will see this error. These functions can only import blocks from the same PCU, so the reference text is flawed if it is in a different PCU.

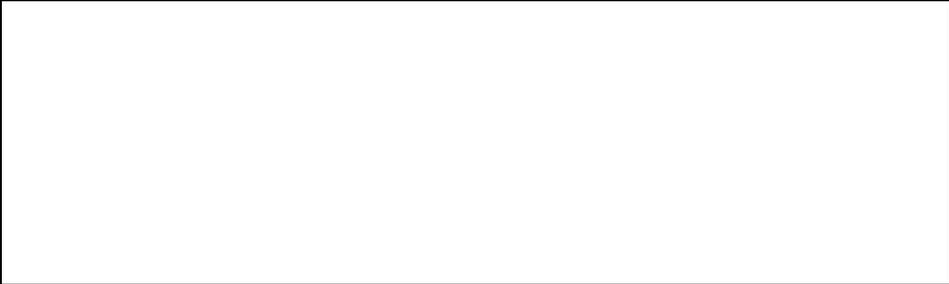
```
--Parsing file H:\...\EBY45EB.CLD... CAD Names: 2300251A.CAD  
External output reference is unreachable for IREF **1AOIK CB CLSD on Module 2,30,02 Block 1652:  
cannot import from Loop 2 PCU 20.
```



Here is the actual source, as determined by the specifications.

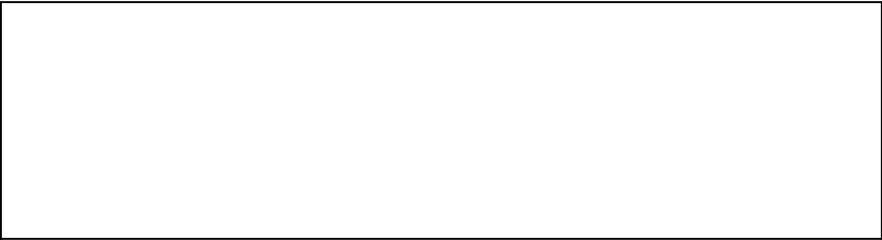


Here is the source that the input reference indicates.

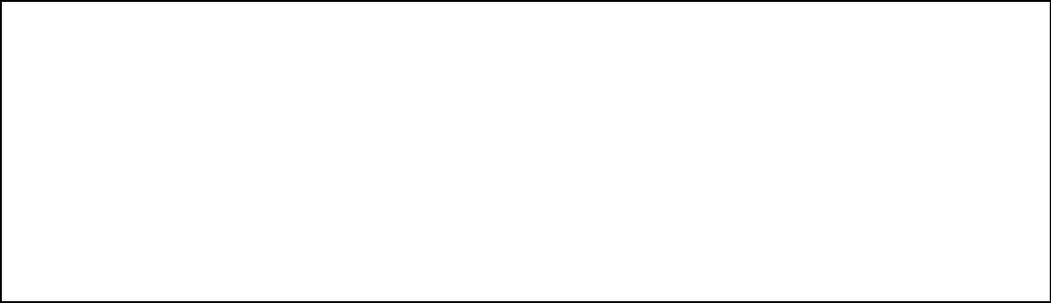


```
--Parsing file H:\...\EBY3DAC.CLD... CAD Names: 2320454A.CAD  
External output reference is unreachable for IREF L30D_MSG on Module
```

2,32,04 Block 1584: cannot import from Loop 2 PCU 22."



Here is the block that is being imported.



Here is the source indicated by the input reference L30D_MSG.



It is possible to see how Composer, in being robust enough to import files from WinCAD, also allows errors to persist in the input references.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNREACHABLE OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

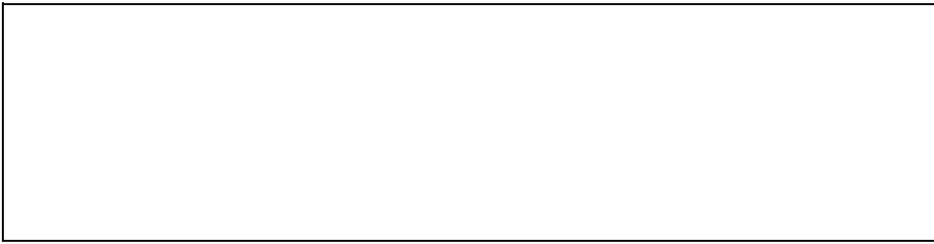
6.6.2.238 Same-loop input attempts to read value in other loop [ERROR 10018]

```
Same-loop input Module 1,01,02 Block 12455 (FC 45) attempts to read value in  
other loop 3
```

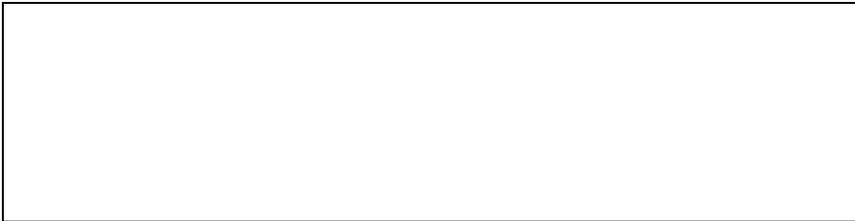
This message tells you that an IREF (Input Reference) exists in the Composer project. However, the function block cannot import it, because it is outside the scope of what it can import.

For example, if an AI/L or DI/L import block has an IREF that exists only in another Loop, you will see this message. The reference cannot tell what is being imported, because these functions cannot import from another loop, only the same one.

```
--Parsing file H:\...\EBY45EB.CLD... CAD Names: 2300251A.CAD  
External output reference is unreachable for IREF **1AOIK CB CLSD on Module 2,30,02 Block 1652:  
cannot import from Loop 2 PCU 20.
```



Here is the actual source, as determined by the specifications.



Here is the source that the input reference indicates.

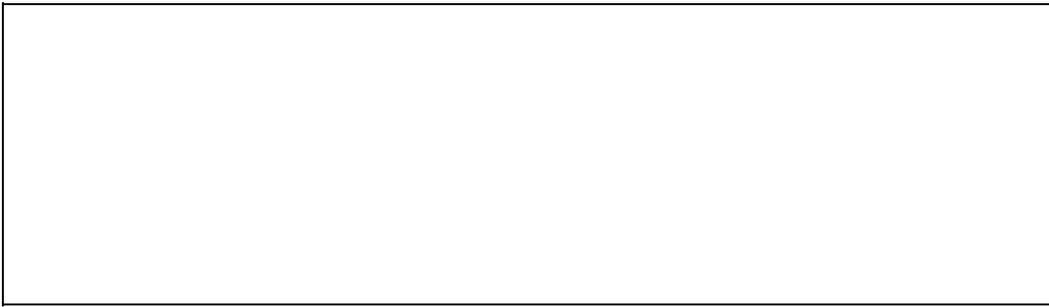


```
--Parsing file H:\...\EBY3DAC.CLD... CAD Names: 2320454A.CAD  
External output reference is unreachable for IREF L30D_MSG on Module
```

2,32,04 Block 1584: cannot import from Loop 2 PCU 22."



Here is the block that is being imported.



Here is the source indicated by the input reference L30D_MSG.



It is possible to see how Composer, in being robust enough to import files from WinCAD, also allows errors to persist in the input references.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNREACHABLE OREF.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.239 Segment control block auto-sequencing enabled [INFO 302]

```
Segment Control Block Module 197,23,03 Block 1234 S15=1 has auto-sequencing
enabled
```

Function blocks will not be executed in block number order.

The *Function Code User Manual* says:

S15 ? SEQUEN

(Auto sequencing signal: 0 = off, 1 = on) If this specification equals one, the module finds and saves the most logical execution order of the function blocks and will execute them in that order, despite block numbers. Auto sequencing helps prevent loopbacks. Loopbacks occur when a block requires the output of a higher numbered block to complete its execution. The segment must then go through two or more execution cycles before the output of the first block is correct. If the auto sequencing function is off (zero), blocks are executed in ascending numerical order.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO SEGMENT CONTROL BLOCKS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.240 Segment control block bad priority [ERROR 300]

```
Segment Control Block Module 197,23,03 Block 1234 priority S3=314 is not in the
range 0..255
```

Segment priority S3 must be in the range 0–255.

The *Function Code User Manual* says:

S3 ? SPRI

(Segment priority) Assigns execution priorities to up to eight active segments. An active segment is one that is ready to run or is running. If two or more segments are active, the processor will run the highest priority segment. Segment priorities should be selected from zero to seven with zero being the lowest priority segment.

In spite of this recommendation, segment priorities up to 255 are permitted.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.241 Segment control block check cycle time [ERROR 303]

```
Segment Control Block Module 197,23,03 Block 1234 S2=0.1 minutes is greater than
2 seconds
```

Cycle time is unusually short or long.

The *Function Code User Manual* says this:

S2 ? CYCTIM

Sets the target segment execution cycle in time units selected with the ones digit of S1. In each segment, blocks execute in a predefined order, selected with S15. A cycle consists of one execution of the blocks plus any idle time (cycle time remaining after the cycle has been executed). Cycle time is the length of time from the start of one cycle to the start of the next cycle.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.242 Segment control block checkpoint period is too small [ERROR 304]

```
Segment Control Block Module 197,23,03 Block 1234 S4=3 times S2=.0001 minutes is less than 20 milliseconds
```

The checkpoint period S4 times the cycle time S2 must be greater than 20 milliseconds. The *Function Code User Manual* has information about how to calculate a value for S4.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.243 Segment control block duplicate priority [ERROR 301]

```
Segment Control Block Module 197,23,03 Block 1234 S3=7 but Module 197,23,03 Block 1233 already has that priority
```

Another Segment Control Block in this module specifies the same priority. (The two blocks have the same value of S3.)

The *Function Code User Manual* says:

S3 ? SPRI

(Segment priority) Assigns execution priorities to up to eight active segments. An active segment is one that is ready to run or is running. If two or more segments are active, the processor will run the highest priority segment. Segment priorities should be selected from zero to seven with zero being the lowest priority segment.

So it does not explicitly prohibit two segments with the same priority, but neither does it say exactly what happens when two have the same priority.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.244 Skipping file: Error parsing submodel [INTERNAL 055]

```
Error parsing submodel sample.m2, skipping
```

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.245 Skipping file: File contains no records [INTERNAL 059]

```
File NSTAT051.DIS contains no records
```

This message identifies a file that DBDOC believes contains no records. Please send us a copy of it if the file does contain records.

Example:

```
File x contains no records -- skipping  
File x contains no records
```

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.246 Skipping file: File too short [INTERNAL 058]

```
File NSTAT051.DIS too short -- skipping
```

This message identifies a file that we cannot process because the file appears to be too short. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.247 Skipping file: OpenDesign Exception [INTERNAL 052]

```
OpenDesign Exception: Sample Exception - skipping
```

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.248 Skipping file: Possibly corrupt A/B PLC file [INTERNAL 057]

```
Possibly corrupt A/B PLC file -- skipping
```

This message identifies an Allen-Bradley PLC file that we cannot process, and may possibly be corrupt. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.249 Skipping file: Unable to read file [INTERNAL 054]

```
Unable to read file SAMPLE.DWG: C:\DIR1\DIR2\SAMPLE.DWG - skipping
```

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.250 Skipping file: Unknown M1/M2 file type [INTERNAL 056]

```
Unknown M1/M2 file type -- skipping file sample.m2
```

This message identifies a M1/M2 file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.251 SODG aliasing error [ERROR 160]

```
SODG aliasing error: Tag name TAGNAME is alias for GUID  
{12345678-ABCD-EF01-2345-6789} (tag TAGNAME2) not found in database DB1.DBF
```

This is a rare message that indicates a Composer database issue.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NAME NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.252 Source in same module [ERROR 157]

```
Source in same module: AI/I Reference Module 1,01,02 Block 1001 (Tagname)  
[Alias, Alias2]
```

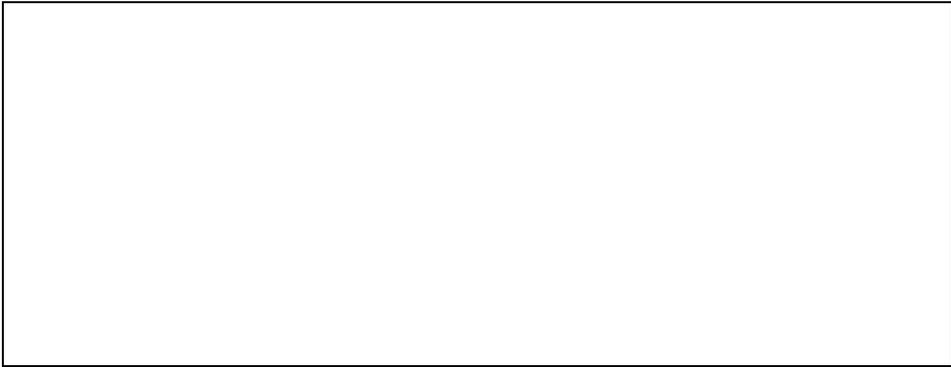
This message means that an exception report input is being attempted from the local module. In INFI 90 Loop, this might cause an "NPMError" when the module is started. Since there was no error in Plant Loop, this has at times been a significant problem.

Example:

```
--Generating index: Module 1,34,02 Block Index  
Source in same module DI/L Reference MM0310A Module 1,34,02 Block 3365 DOSING EAST MAIN CONTROL
```



A bit of searching for block 3365 shows that the problem is that S1 should be 4, not 2. Module 1,34,04 Block 3365 is the intended source.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SOURCE IN SAME MODULE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

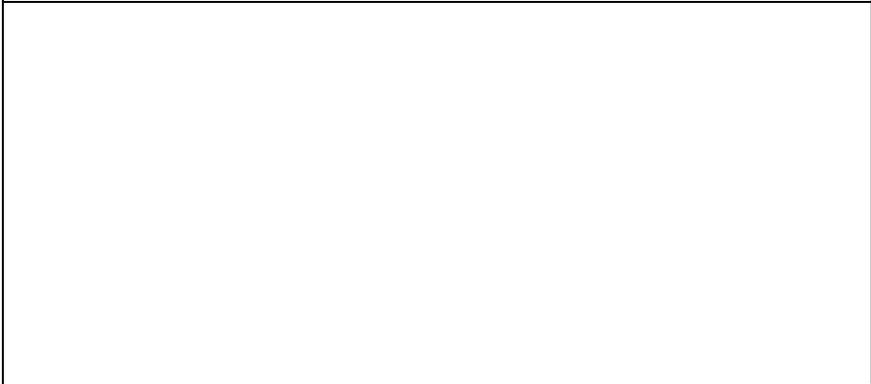
6.6.2.253 Source wired to another source [ERROR 158]

Source may be wired to another source: Module 1,01,02 Block 1001

This message shows a wiring error. We find it when sources appear to be wired together.

Example:

```
--Scanning drawing 3100399 PAF A AUX LOGIC (3/3)  
Source may be wired to another source: IREF (PAF A FAN BRG 2 VIB H H).  
Source may be wired to another source: IREF (PAF A FAN BRG 1 VIB H HH).
```



The circled area shows an incorrect connection between S1 and S2 of block 3722.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SOURCE TO SOURCE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.254 Span is zero for dynamic bar [COSMETIC 231]

```
Span is zero for dynamic bar BLANK TAG_1 on CHOCOLATE FACTORY OVERVIEW --  
suppressing
```

This message means that the dynamic bar cannot be displayed by DBDOC because the span is zero. A non-zero value is required to display a bar graph.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC DYNAMIC BAR SPAN ZERO.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.255 Spare block used in downstream wiring [CHECK 168]

```
Spare block imported by Module 1,01,02 Block 1001 and then used in downstream  
wiring
```

A spare block is apparently used in the logic. This is usually an oversight.



The block 3200, on the right, is a spare input, as is shown by the specs below, (S9 is 9999). However it is still connected to all the blocks listed on the left. This message is also generated for FC205 and FC206 when

S5=31999.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK SPARE BLOCK USED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.256 Spec input index out of range [ERROR 018]

```
Spec input index out of range for Module 1,11,30 Block 2522 (FC 2): 0
```

This message means that we cannot process the function code correctly at this time.

This is a rare message. When we have encountered it, the CLD/CAD files have been corrupted. We would appreciate getting copies of any CLD/CAD files that show this problem.

Note, however, that the usual reason for this message appearing is that there are duplicate sources for a block. Our processing gets confused because the specs we find are attached to the wrong block. Thus, you should first check to make sure that there is only one "Source" entry in the block index.

If more than one exist, the problem is that you have not included only the functioning CLD/CAD files for the module. You should fix the build to eliminate the duplicate sources and probably get rid of these message at the same time.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC SPEC INPUT INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.257 Spec is illegally adapted [ERROR 083]

```
Spec is illegally adapted: Adapt block Module 1,01,02 Block 1001 modifies S5 of  
Module 1,01,02 Block 1234 (FC 30)
```

This message means that illegal targets are being used for adapt blocks, specifically specs that exist but that are documented as not being adaptable.

This usage can seem harmless, because the problem is latent. Once the adapt block has been configured and loaded, it will work until a cold reload.

Impact: If you compile these blocks, you will get no errors. Also, if the block already is in the redundant

module, you can reload with no problem. However, when you download a module with this logic in it for the first time and change mode to Execute, the module will go into Error mode (flashing Green status LED) showing one of these error indications:

- LED 1+3 ON. Module Status Report: Config Error - Undefined Block, or
- LED 2+3 ON. Module Status Report: Config Error - Input Data Type Incorrect,
- with xx = Block # Making Reference (xx = Block Number of the Adapt).

Solution: Change mode to Configure and correct the error, compile and download again. Of course, take care of all errors of this class first.

Note: Some of these messages can be especially difficult to comprehend. We had clients experiment with an adapt block that was applied to an alarm specification, documented as illegal. Here were the results:

- If the adapt situation already existed, the redundant module could be loaded and the module would go into execute properly.
- If the adapt situation was new, as when the logic was cloned on another sheet, it would compile and load. However, in the new situation, the module would fail to go into execute mode.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ADAPT EXECUTE ERROR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.258 Specification out of valid range [ERROR 299]

```
Module 3,07,03 Block 850 FC45 S2 value 5 is out of the valid range [0,2]
```

The specification is not in the documented range.

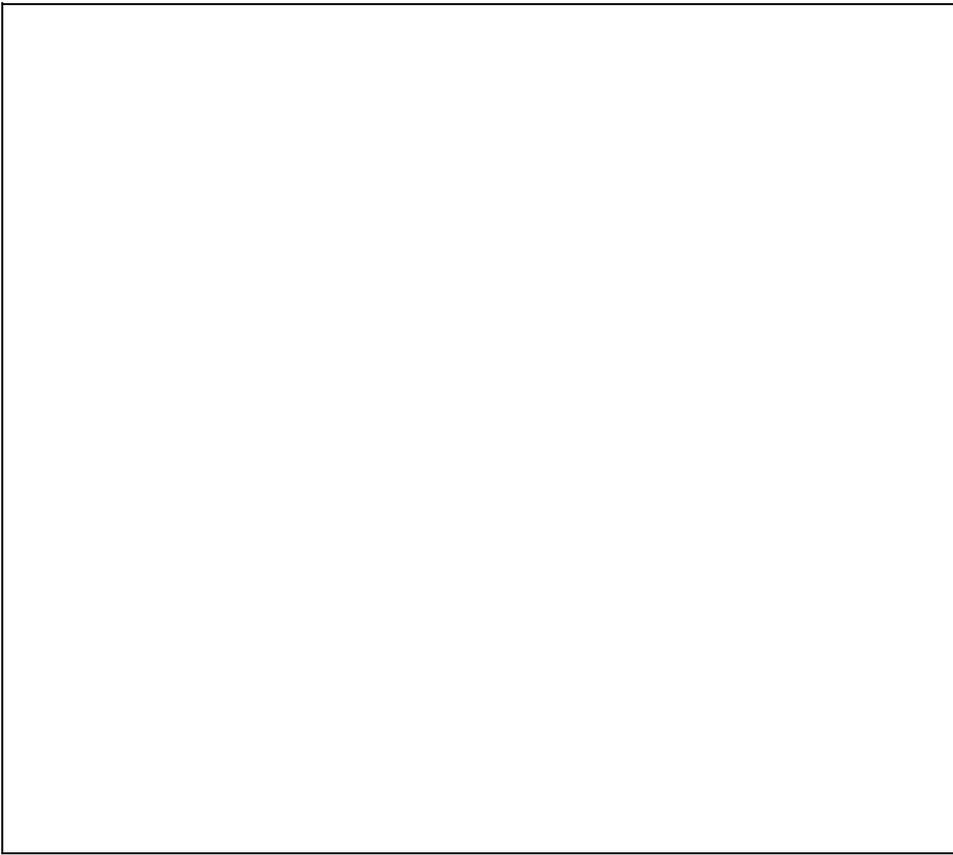
Specifications are tested against documented ranges. Also, invalid values can be allowed because their use appears to be innocuous. Some messages definitely show blocks that do not function. Please let us know of any messages that apply to valid usage, even though it is not documented. The same applies to feedback about real errors that are detected.

Example 1 DO/L block with illegal alarm state value:

```
Module 5,03,02 Block 3835 FC45 S2 value 3 is out of the valid range [0,2]
```

You can see the error, and also how a simple search finds the complementary tag, with a valid specification

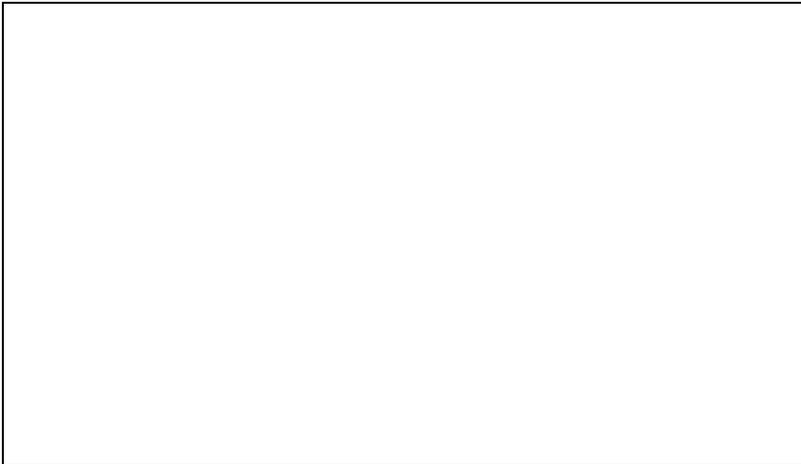
value.



Example 2 Rung block with illegal override specification:

```
Module 1,01,02 Block 13567 FC111 S1 value 10 is out of the valid range [0,3]
```

We flag the bad specification as some sort of overridden output, but the new message is simple and direct. In any event, it is absolutely clear that the problem here is that S2 should be 10, and S1 simply 0.



This block does not do what it is supposed to.

Example 3 Tens digit of S2 must be 0 for FC35

Module 2,01,04 Block 2500 FC35 S2 value 011 is out of the valid range [000,102]

For FC35, S2 can be within the range 000 to 102, but the tens digit must be 0.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.259 SQLite bind error [INTERNAL 208]

```
SQLite bind error (0): bind error
```

Our programs use SQLite in internal processing. These messages are indications that our compiler encountered data that violated our programming assumptions.

We might need to ask for files that cause these messages for analysis.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SQLITE MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.260 SQLite error [INTERNAL 207]

```
SQLite step error: out of memory
```

Our programs use SQLite in internal processing. These messages are indications that our compiler encountered data that violated our programming assumptions.

We might need to ask for files that cause these messages for analysis.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SQLITE MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.261 SQLite error [INTERNAL 081]

```
SQLite step error: out of memory at Bailey.c line 1234
```

Our programs use SQLite in internal processing. These messages are indications that our compiler encountered data that violated our programming assumptions.

We might need to ask for files that cause these messages for analysis.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SQLITE MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.262 SQLite FTS Indexing Batch Failed [ERROR 313]

```
SQLite FTS indexing failed on 1 topics, beginning with TOC
```

SQLite FTS Indexing Batch search Failed on database

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.263 SQLite FTS Indexing Failed [ERROR 312]

```
SQLite FTS single-query indexing failed, trying to index searchable topics in batches
```

SQLite FTS Indexing search Failed on database

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.264 ST parse error [INTERNAL 346]

Unsupported ST syntax at line 15 (could not parse expression)

This message indicates a failure of our Structured Text program parser at the given point in the document. Likely this is due to us not handling the syntax used.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

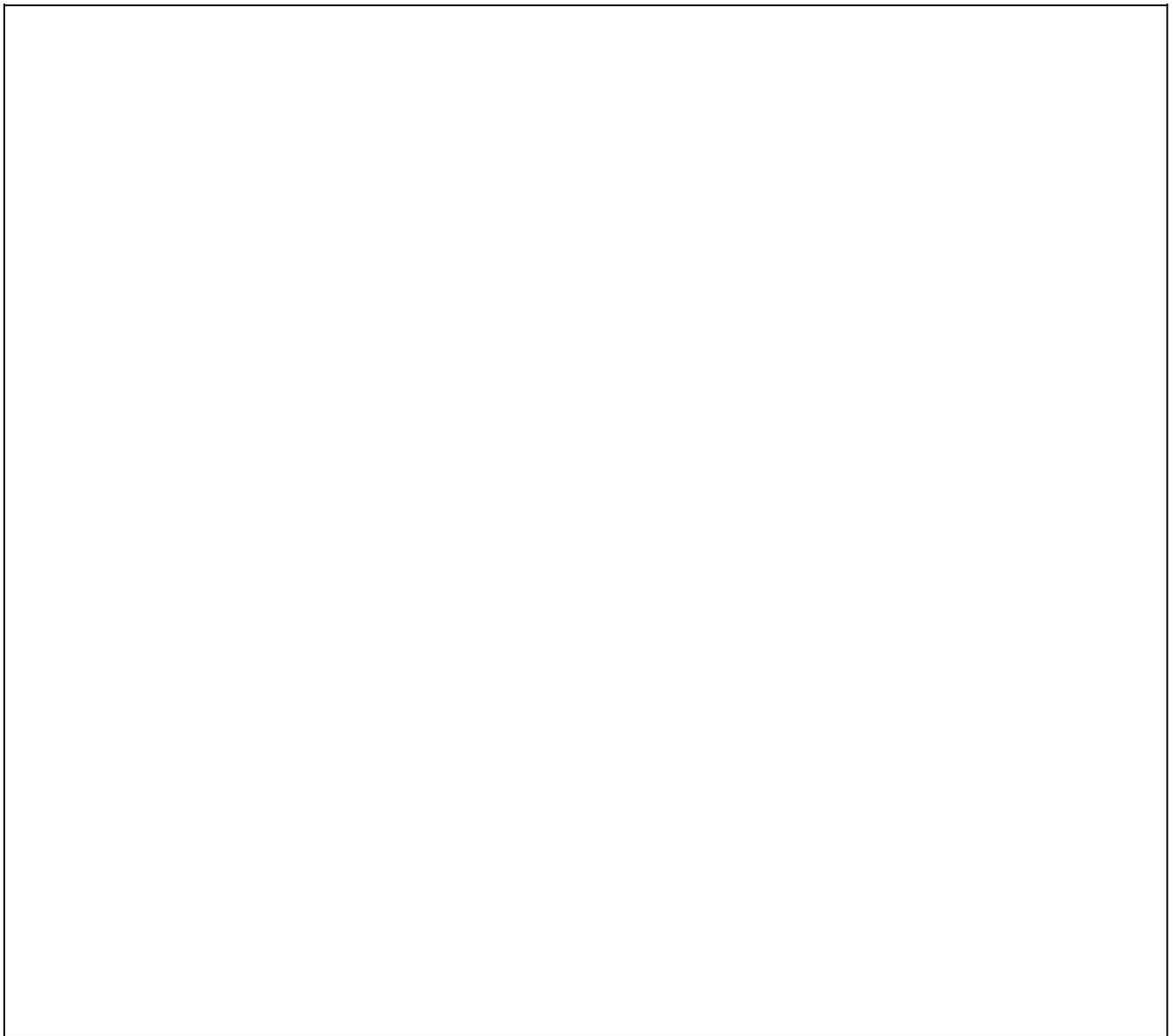
6.6.2.265 Station span * PID gain greater than 125.0 [INFO 179]

Station Module 1,01,02 Block 1001 span 20.0 [S10] and APID Module 1,01,02 Block 1501 gain 6.5 [S11] product 130.0 is greater than 125.0

The overall gain constant of a PID (FC 18 or 19) or APID (FC 156) controller is designed to scale PV / SP values into a range of 100%. When this is done, the proportional gain constant, in particular, is 1.0 for a gain of one control algorithm.

These messages show where that situation does not exist. These blocks can be tuned, but the tuning constants will not be "normalized" and thus will be subject to misinterpretation.

The product of S11 on the APID and the S10 on the M/A MFC/P must fall in the range of 80 to 125. This error occurs when the product is below that range.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO PID GAIN-SPAN.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

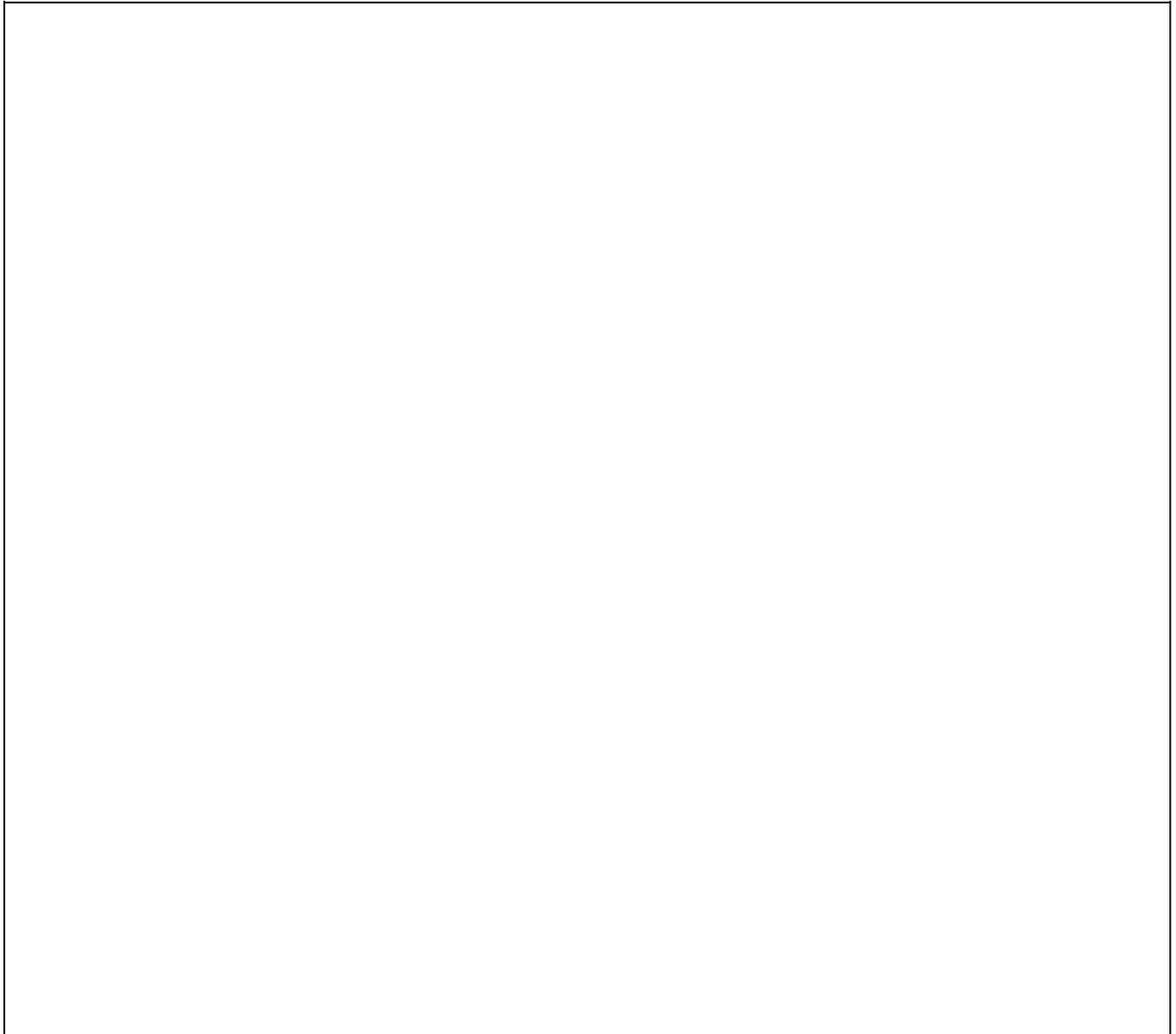
6.6.2.266 Station span * PID gain less than 80.0 [INFO 178]

Station Module 1,01,02 Block 1001 span 20.0 [S10] and APID Module 1,01,02 Block 1501 gain 3.5 [S11] product 70.0 is less than 80.0

The overall gain constant of a PID (FC 18 or 19) or APID (FC 156) controller is designed to scale PV / SP values into a range of 100%. When this is done, the proportional gain constant, in particular, is 1.0 for a gain of one control algorithm.

These messages show where that situation does not exist. These blocks can be tuned, but the tuning constants will not be "normalized" and thus will be subject to misinterpretation.

The product of S11 on the APID and the S10 on the M/A MFC/P must fall in the range of 80 to 125. This error occurs when the product is below that range.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO PID GAIN-SPAN.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.267 Symbol flag detection failed [INTERNAL 287]

Symbol flag detection failed during tokenization in ABC12345.G

The DBDOC build system was unable to tell whether this file was a symbol or a graphic. When this happens, the build system treats the file as a graphic.

If you see this error, we would appreciate it if you contact us with details so that we can improve DBDOC's ability to tell the difference between symbols and graphics.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.268 System error total exceeded [BUILD 60001]

```
System maximum error count exceeded: Further error messages are suppressed
```

The build contains a very large number (>20000) of error messages. Because of this, not all of them are shown in the error browser.

(We believe no one would walk through the whole list, but they are listed in DBDOC_SUMMARY.ERR and DBDOC.ERR.)

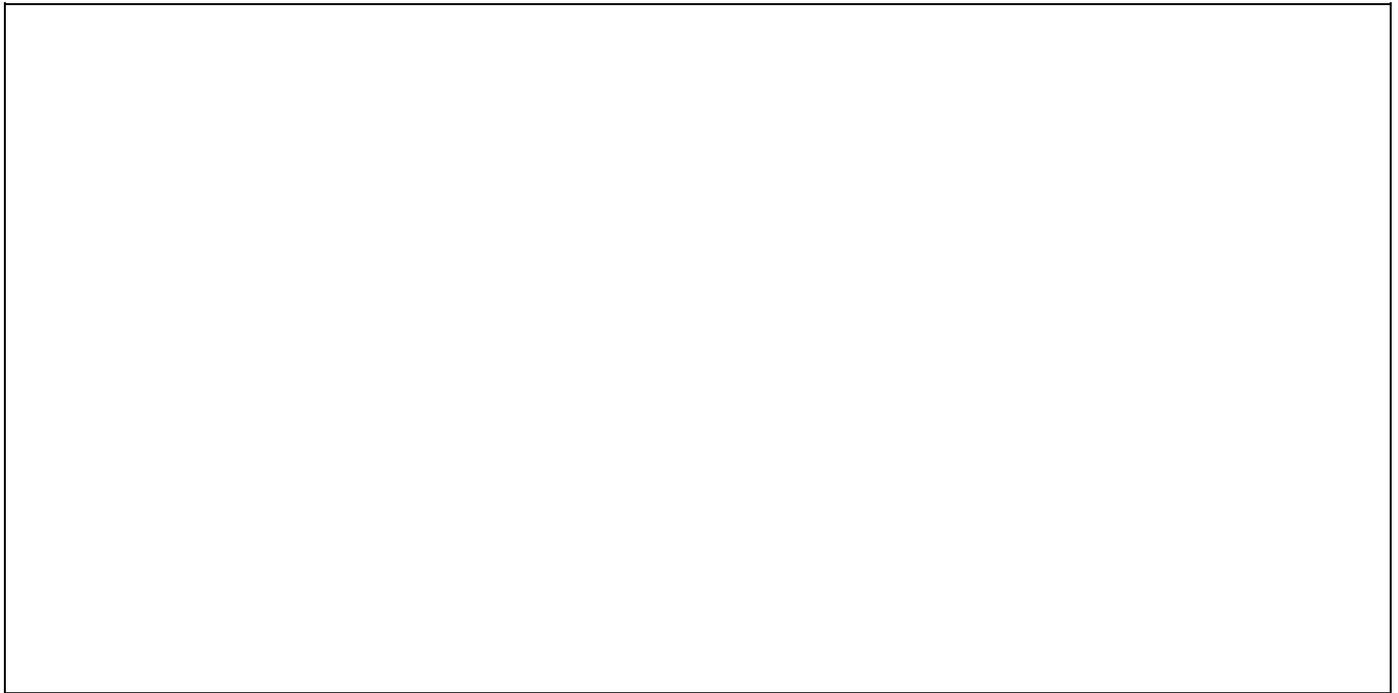
This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD ERRORS LIMITED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.269 Tag atom not upper case [CHECK 264]

```
Tag atom UndefTag should be upper case
```

The INFI 90 graphics packages are based on Microsoft Access techniques using tagnames as keys, which must be all in uppercase. When they are not all upper case, DBDOC treats them as undefined, and places them in the Index of Undefined Tags. They may not work as expected.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK UNKNOWN ATOM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.270 Tag failed to find matching format [INTERNAL 209]

```
Tag '30FQI170RS' Failed to find matching format for RCM, defaulting to first  
format (Record 26)
```

This message means we have had some failure in creating a format template for the tag named. Please call this to our attention.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL MISSING FMT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.271 Tag index not found [ERROR 159]

```
Tag index not found in AACRT003.DIS: 640
```

This is a message that should always be investigated. It means that graphics are being controlled by a tag

index with blank tagnames (except in OIS 20 and MCS systems). This is dangerous because maintaining a correct set of tag index values proves difficult over the long run.

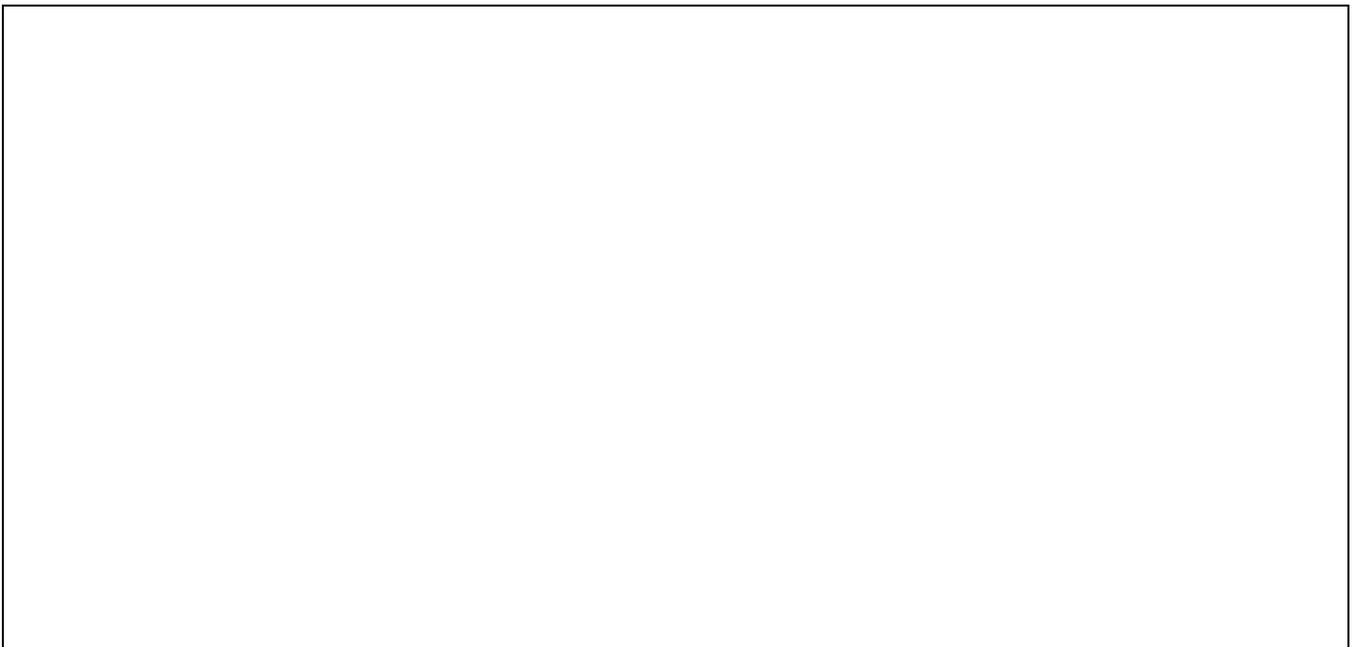
In most SODG systems, any tag index message means that the tagname is blank. You should consider this to be worth a lot of attention, as it means that you can lose the connection to the intended tag easily.

The value of this message for pop-ups is substantial, as it is the easiest way to find out that pop-ups that have been keyed by index refer to tags that no longer exist.

DBDOC has techniques for finding and eliminating blank tagnames.

```
--Scanning graphic AACRT003.DIS
```

Tag index not found in AACRT003.DIS: 640



This means that a display element references a blank tagname with tag index 640, but no tag with index 640 was found in the database. That part of the graphic simply does not work. DBDOC cannot tell you what tag index 640 is supposed to be. You have to find that in your archives. The message tells you that the tag index is used in at least one graphic, and it does not exist in the database.

However, searching for places that the undefined tag TAG_00640 is used might be useful. In the image, that is not the case, because you can see that the problem is simply in a trend group. If you find the tag index used on a graphic that gives you some additional clues, it might help resolve it. DBDOC links directly to all of them. In the example, you might find that graphic AACRT061.DIS contains enough information for you to identify the problem and correct it.

It is for this reason that DBDOC synthesizes an undefined tag TAG_00640 (in the example) and creates the links to it – it might help sort it out.

In INFI 90, a SODG graphic can be controlled either by the tagname or the tag index.

The latter method (tag index) is the only way for OIS 20 and MCS consoles to display graphics. For these graphics, the tag index MUST relate to an entry in the tag database that is meaningful. In those old systems, graphics could be cloned by directly changing the tag index, so that it was no longer related to the tagname in the graphic .DR (or symbol .DY) file. This resulted in many graphics and symbols with mismatched tag names and index values.

In the case of Conductor VMS, WinLDG (WinTools SODG graphics) and PCView / LAN90, the tagname is the correct key to what is displayed. In fact, in PCView / LAN90, the tag index has no meaning as a concept.

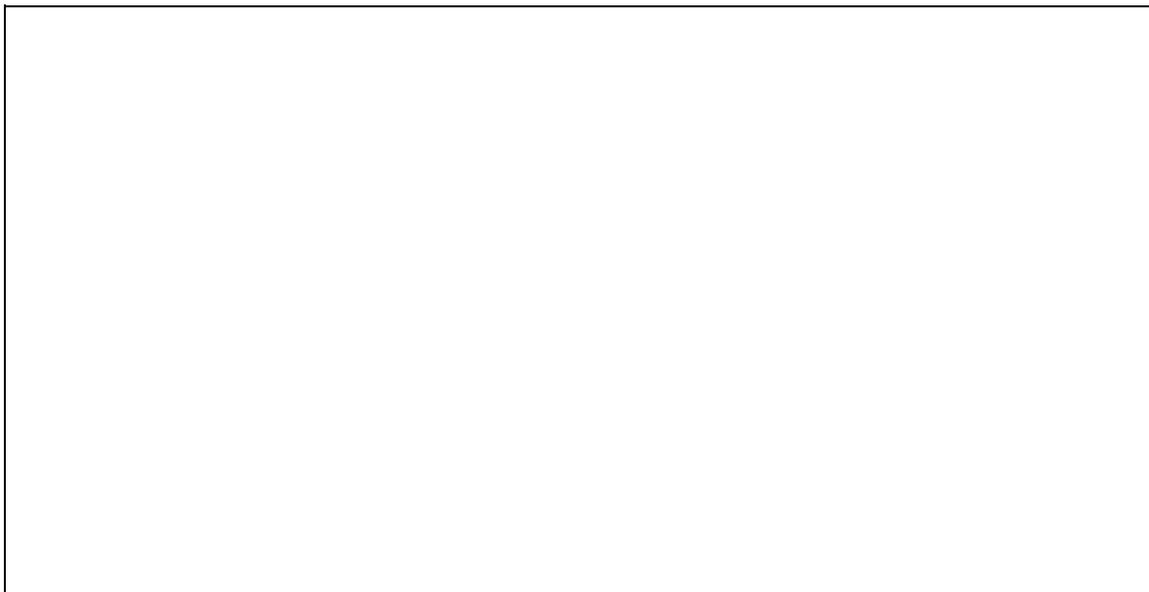
When tag index keyed versions of SODG graphics were upgraded, and the tagname did not match the tag index in the .DR or .DY file, some procedures blanked the tagname. From that point on, only the tag index was in the file to tell what tag to display at that position.

The tag index, however, is subject to re-indexing. If this is done, any graphic using tag index only will be affected negatively.

Note, also, that even a few of these errors might show you that a serious problem exists. If index values are not correct for a few that come out here, it can mean that many more values are wrong. There is no message because the tag index points to a tag that exists, but that tag is no longer the correct one. Consider a simple example with tag index values 2 to 5000 in use. If the index values get corrected to be 1 to 4999, all graphics will have been broken.

Finding tags with blank tagname

In a SODG graphics system, search for the text "blank tagname" (without the quotes). If you find any, you know that your system has graphics links being made by tag index instead of tag name. This is not stable in the long term.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.272 Tag index not found [ERROR 272]

```
Tag index 423 not found in 31s1045-.dt at 1942,3444
```

This message means that a symbol has a tag index that is not found in the tag (or trend) database. The name of the tag might be blank or not found in the database either.

The value of this message for pop-ups is significant, as it is the easiest way to find out that pop-ups that have been keyed by index refer to tags that no longer exist.

This message can be important in conversion of SODG systems to Symphony Plus.

Note that tag index values 1, 6 and 100 are built into ABB standard symbols, so must be ignored if they show up.

The conversion of SODG graphics, even ones using names as keys, is done using tag index values in .DT text files. Any errors in tag name to index mapping will cause problems.

This message can be found in unused symbols. The recommended technique is to use DBDOC builds to remove unused symbols iteratively, so that it is clear that errors apply to symbols that are used.

DBDOC has techniques for finding and eliminating blank tagnames.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.273 Tag index not found in database [ERROR 123]

```
Tag name TAGINDEX_06554 not found in database in MSDD31.DY
```

This message says that a tag was only identified by tag index, and that tag index was not found in the tag database. It can also refer to TRENDINDEX_nnnnn with the same sense. In both cases, there is no tag name to use, only a numerical index value that was not found in the assigned database.

This message should only apply to MCS and OIS SODG graphics, the most archaic form. PCView, WinTools and Conductor VMS all use tagname and trendname to make links.

However, the conversion from any SODG graphics system to Symphony Plus appears to need correct tag index values, so this error is very important if graphics are being cleaned up for conversion.

Example:

```
--Scanning graphic AC-T1 AIR COMPRESSOR TRND (AC-T1)  
Tag name TAGINDEX_07631 not found in database in trndpvfl.DY
```

This message is closely related to Build error 67: Tag name not found in any database

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NAME NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.274 Tag mismatch [ERROR 101]

```
Tag mismatch for 16HY570: REJECTS.WATER_EXTRACTOR._16HS570 in AC 800M;  
REJECTS.WATER_EXTRACTOR._16HS570a in AC 800M; Tag mismatch for TAG1: Module  
1,01,02 Block 1234 in DB1.DBF, DB2.DBF; Module 1,01,02 Block 1235 in DB3.DBF
```

This message may indicate multiple AC 800M elements with the same tag name.

Or during the scanning of the various databases, a tag was found that was defined with different combinations of loop, PCU, module and block. DBDOC depends on a unique tagname corresponding to a unique block number in a module.

Often, you will find that Composer or WinCAD actually have the correct definition of the tag. The reason for this is that they automatically generate a tag definition when an exception report block is created. In the console, on the other hand, the correct parameters have to be entered manually.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.275 Tag name not found in any database [ERROR 067]

```
Tag name UNDEFTAG not found in any database
```

This message means that a tag name was found in the graphics file but was not in any database. These undefined tags represent "deadspots" in the graphics - places where you would expect a tag to be displayed

but it is not.

Sometimes this message happens when a graphic is brought from somewhere else, used as a prototype, and left on the system. In this case, you simply move the inappropriate files into another folder and ignore them in the DBDOC build. It also occurs when a tag is created in Composer or WinCAD, but is not moved into the tag database.

```
--Scanning graphic AACRD00.DIS  
Tag name TC44165-CO not found in any database.  
Tag name TC44165 not found in any database.
```

This graphic has one or more tags on it that are not in the database and do not work.

A message such as this indicates that the tag specified is referenced in a submodel included by another submodel:

```
--Scanning graphic 1-BA-U1ELECTRICALTABLE.M1  
Tag name 1BAT10 CE031 not found in any database  
in LYB_TagTextDescrip included by LYB_AnalogTableAOL.
```

When many tags are not found on a single graphic, it means that graphic is very much broken, and is probably not in use.

See how the Audit Window actually shows you the type of data being displayed for the undefined tag.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NAME NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.276 Tag name not upper case [ERROR 263]

Tag name UndefTag should be upper case

The INFI 90 graphics packages are based on Microsoft Access techniques using tagnames as keys, which must be all in uppercase. When they are not all upper case, DBDOC treats them as undefined, and places them in the Index of Undefined Tags. They may not work as expected.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

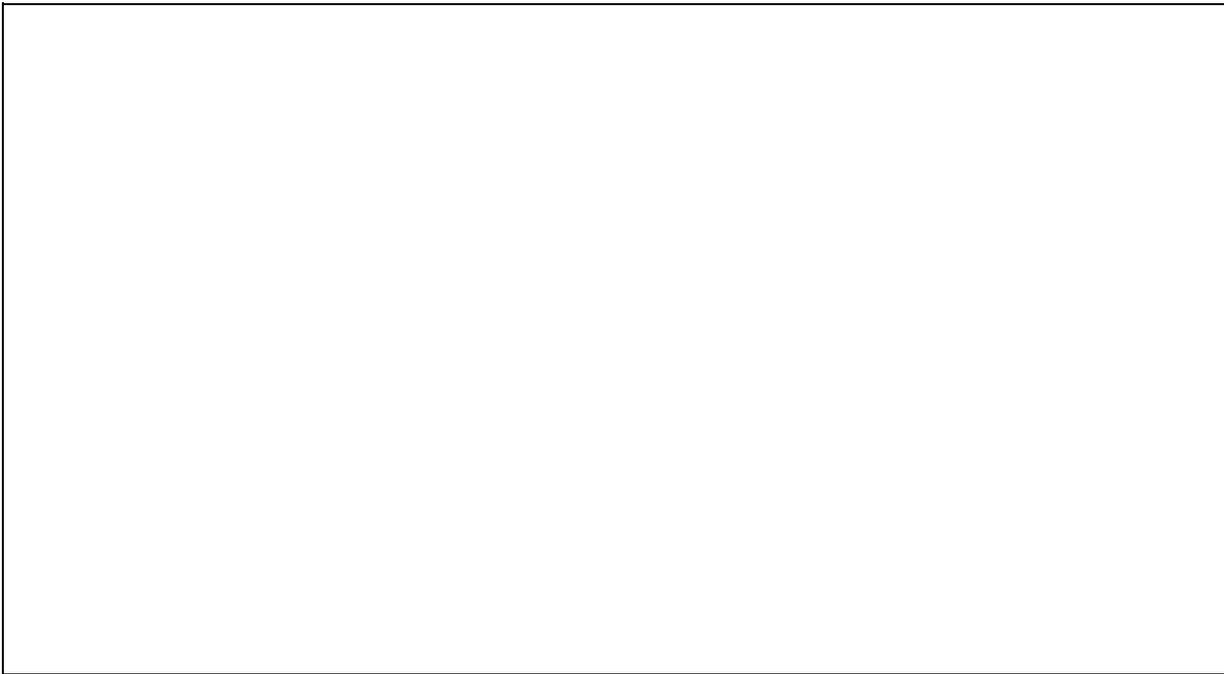
This error type was originally reported in error file **DBDOC TAG NAME NOT UPPER CASE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.277 Tag not an exception report [ERROR 256]

Tagged but not exception-reported: Module 1,10,31 Block 22935 (19) is tagged
12LC03050 in TAGDATA.DBF

A tag name has been defined for a block output that does not generate an exception report, but the tag is not used on any graphic.

The tag should be corrected, as third-party HMI systems, alarming and history might depend on it. It cannot serve any purpose as it is.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NOT EXCEPTION REPORTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.278 Tag not found by name [INTERNAL 293]

Tag UNDEFTAG on DOCUMENT was not found by name, searching by index

DBDOC was unable to find an S+ tag by name.

In most cases, this error will not be generated. The error that is usually generated when an S+ tag is not found is Tag not found in any database [ERROR 284].

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.279 Tag not found in any database [ERROR 284]

Tag UNDEFTAG on DOCUMENT not found in any database

This message means that a tag was found in the graphics file but was not in any database. These undefined tags represent "deadspots" in the graphics - places where you would expect a tag to be displayed but it is not.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NAME NOT FOUND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.280 Tag not found in correct database [ERROR 288]

```
Tag UNDEFTAG not found in database DB1.DBF (defined in DB2.DBF) in A.G included  
by B.G
```

The tag mentioned was not found in the database used with the graphic. It does not work. This message tells you that you likely have a definition of what the tag should be to guide you.

This can mean:

- Your build is flawed because the wrong database is associated with the console graphics.
- A tag was created in Composer but not moved into the tag database.
- In systems with multiple consoles and multiple taglists, it can indicate omitted tags that exist but that have not been added to the taglist where they are required. When individual taglists exist, you have to make sure that all tags you need in each console are in the taglist for that console.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG FOUND IN OTHER DATABASE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.281 Tag not found where specified [ERROR 068]

```
Tag name UNDEFTAG not found in database DB1.DBF (defined in DB2.DBF) in A.DY  
included by B.DY
```

The tag name mentioned was not found in the database used with the graphic. The database mentioned has the tag needed but it was not associated with the console graphics.

This can mean:

- Your build is flawed because the wrong database is associated with the console graphics.
- A tag was created in Composer or WinCAD but not moved into the graphics database.
- In systems with multiple consoles and multiple taglists, it can indicate omitted tags that exist but that have not been added to the taglist where they are required. When individual taglists exist, you have

to make sure that all tags you need in each console are in the taglist for that console.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG FOUND IN OTHER DATABASE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.282 Tag referenced in PNT database has no LPMB info [COSMETIC 250]

```
Tag XXXXXXXX referenced in PNT database has no LPMB info
```

The referenced tag is in the Honeywell PNT database only. There is no corresponding tag in the INFI 90 database. It has no loop, PCU, module or block data.

Example:

```
--Scanning block file 90103.BLK  
Tag 2HC2016T referenced in PNT database has no LPMB info.
```

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

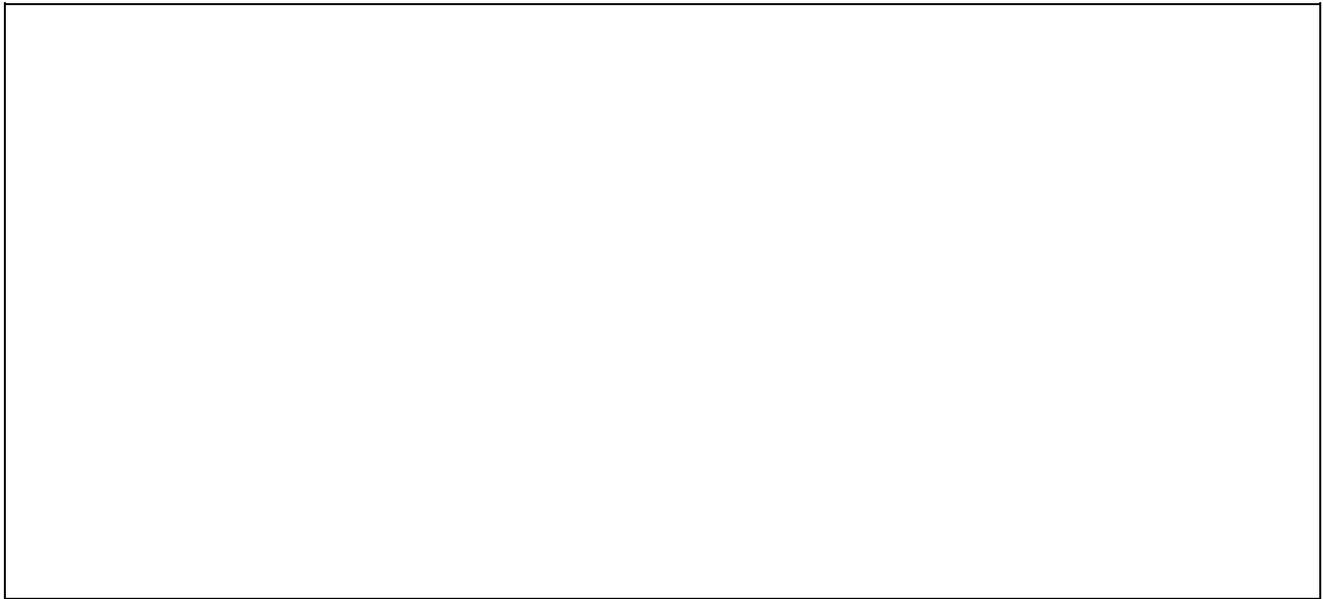
This error type was originally reported in error file **DBDOC-COSMETIC MISSING PNT TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.283 Tag renames block [ERROR 144]

```
Tag Tagname 2 renames Module 1,01,02 Block 1001 (already named Tagname 1 in  
DB1.DBF)
```

We report if the same block has more than one tagname associated with it. This is not likely intended as it wastes a relatively limited resource.

When two tags define the same block in a single database, it can mean that the loop, PCU, module or block of one of the tags is incorrect. The usual meaning of this message when the databases are different is that one entry is obsolete and does not reflect a correction.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC RENAMED BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.284 Tag type mismatch [ERROR 102]

Tag type mismatch: Module 1,01,02 Block 1001 (FC FC 31) is tagged TAGNAME in DB3.DBF with mismatching type ANALOG

The block shown has a tag defined for it, but the tag type does not match the block function code. Some types of mismatches may not be problems, but others are and the tag will not work for the operator.

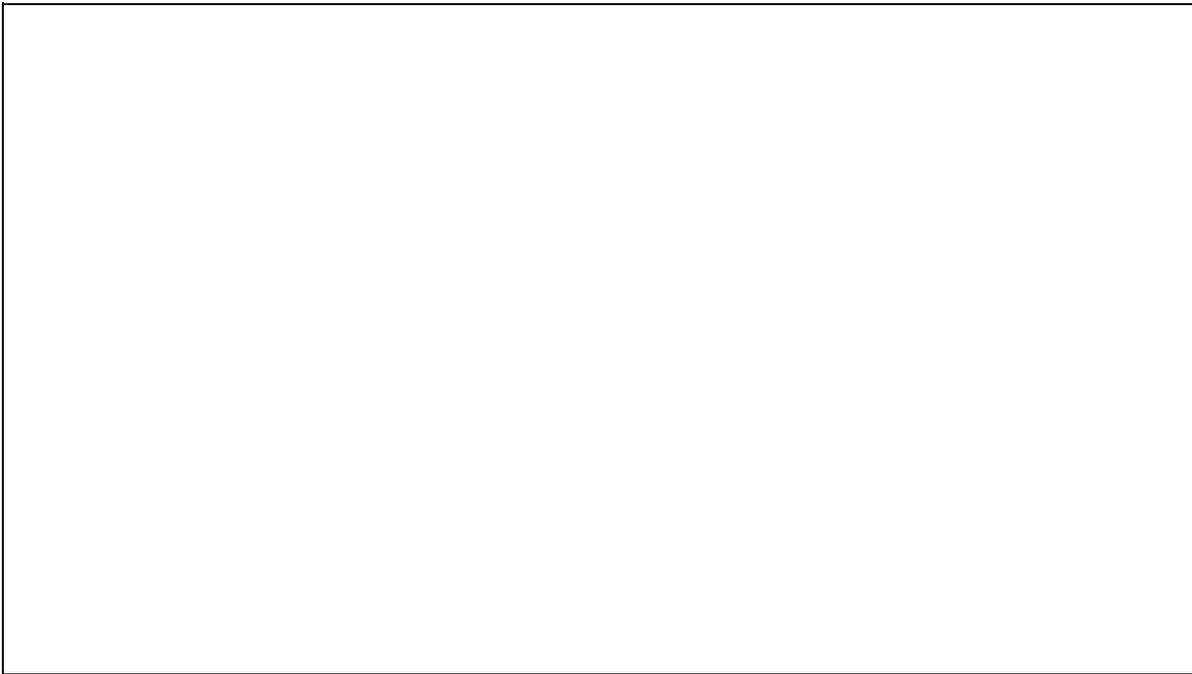
Example:

```
--Generating index: Module 1,01,17 Block Index  
Tag type mismatch: (FC 45) is tagged by tag LI45163 with mismatching type Analog.
```

These tags are in use, but are misdefined. They cannot be used on graphics, alarms or trends.

The snippet from the CLD/CAD sheet source shows that the block is a digital block but the tag definition says it is an ANALOG tag, which is why there is a note about a mismatch.

If an operator tries to make changes on a tag that is defined with the wrong type, those changes will not work.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG TYPE MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.285 Tag used in graphic not exception-reported [ERROR 141]

Tag used in graphic but not exception-reported: Module 1,01,02 Block 1001 (FC 34) is tagged Tagname in DB4.DBF

This message tells you that a tag is defined with a block that is not an exception report block. The tag does not work if it is used on graphics, for alarming, or for history.

The primary function of the message is to show you tags that are on graphics, but not exception-reported.

However, this message includes cases where the only connection to a graphic comes because the tag definition has a primary display defined or alarm vectoring is implemented. You will want to make sure that there is not an operational problem caused by the fact that a desired or required alarm status will not get displayed. You might be counting on it, but it will never happen.

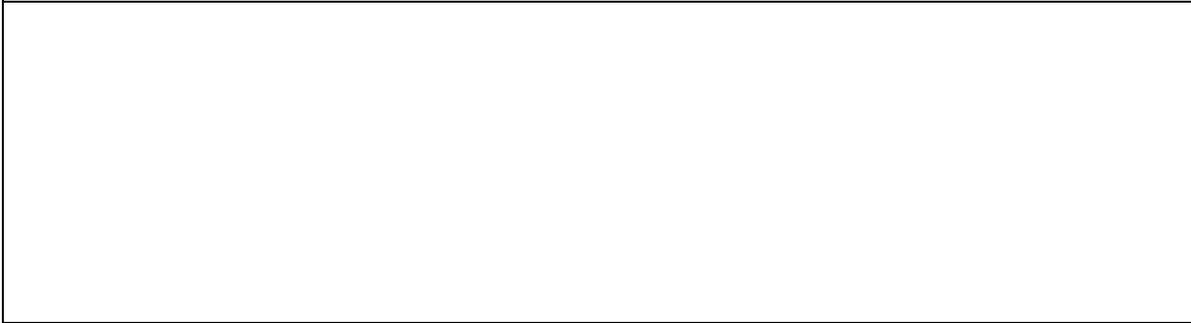
You may fix the tag by:

1. making it an exception report,
2. changing the block number, or
3. deleting it.

Example:

```
--Generating index: Module 4,50,02 Block Index
Tag used in graphic but not exception-reported: Module 4,50,02 Block 3630 (FC 40)
is tagged LC45862-OVR in TAGDATA.DBF
```

For MSDVDR: Multistate Device Driver [FC129], only output block N is exception reported, other output blocks of FC 129 may generate this message.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TAG NOT EXCEPTION REPORTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.286 Tag/trend mismatch [ERROR 105]

```
Tag/trend mismatch: tag TAGNAME (DB1.DBF) refers to Module 1,01,02 Block 1001
while trend refers to Module 1,01,02 Block 1005
```

This message means that during the scanning of the trend databases in a PCView SODG graphics system, a trend tag was found that was defined with a tag that did not match, or it does not have matching tag in any associated database.

In PCView, a trend tag is only meaningful if it is defined with the same block as its source tagname.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC PCV TREND MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.287 TagAtom mismatch [CHECK 229]

```
TagAtom mismatch: SIC9007.EB_ALMSTATUS is used in a Conductor NT systemSIC9007
```

This message means that a tagatom was encountered that does not match the graphics system built.

The tagatoms in Conductor NT and Process Portal are quite different from each other. If we are processing one type and run into tagatoms from the other form of graphics, we warn you. This could result from conversion of graphics or from the use of prototypes from a different system.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK TAGATOM MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.288 Tagname is illegal in PCView [ERROR 104]

```
Tagname TAGNAME (Module 1,01,02 Block 1001) is illegal in PCView (contains '/')
```

Tagnames with '/' cause problems because '/' is a path delimiting character for QNX. Apparently, WinTools does not notice this and will pass such names into .DR files.

This message applies only to PCView Version 5, according to people we have consulted. If you have the older PCView Version 4.3 system, it is not an issue. However, if you upgrade to the newer version, you could have problems. This is most likely to be the case on QNX systems on PCs.

```
--Scanning database file TD_021398_CHEROKEE_1.DBF  
Tagname A/BSELECT (Module 1,01,02 Block 786) is illegal in PCView (contains '/').
```



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC PCV ILLEGAL TAGNAME.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.289 Tested block on different page [CHECK 211]

```
Tested block on different page: TSTALM block Module 1,07,31 Block 21423 on  
10703I1 checks Module 1,07,31 Block 22255 on 1070321
```

If a block being tested by a TSTALM block is on a different page, it should be checked to make sure that the intended block is being tested.

These messages exist because they often point to an incorrect value of S1. If cloned logic was not completed, and the logic was cloned on another page, this message will occur along with a message saying the target is tested by multiple TSTALM blocks.

This message is cosmetic in that DBDOC makes TSTALM blocks easier to manage. However, the test is capable of finding errors that are not easily found any other way.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK TSTALM PAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.290 Tokenization unrecognized [INTERNAL 269]

```
Tokenization Unrecognized Data in ABC12345 at 1: <Unrecognized Token Data>" "
```

A token in a PGP/S+ file is unexpected/unrecognizable. The graphic may not work as expected.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.291 Too big for float [ERROR 10027]

```
Too big for float: Module 1,01,02 Block 12345 (FC23 S3) value 1.0
```

The parameter is bigger than the biggest legal floating point number.

The parameter is bigger than the biggest legal floating point number. Some constant has been noted as indicating a number greater than 9.2E+18. This usually means that an integer was put into a floating point field, making an illegal number.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.292 Too big for short: integer value [ERROR 10009]

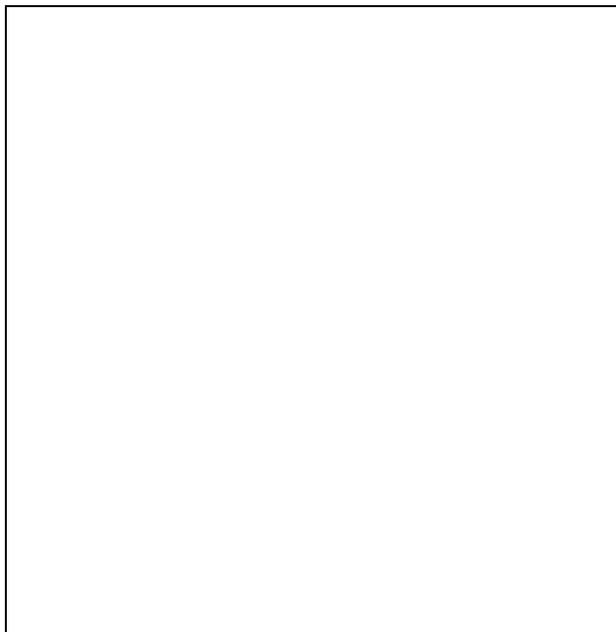
```
Too big for short: Module 1,01,02 Block 12345 (FC23 S3) integer value 65536  
(real value 1.0)
```

This message means that a CLD file has a value for a specification that is apparently illegal. We have found this as uncorrected damage from early versions of Composer after the verify with update feature was used.

Most of these messages occur when you switch to Composer version 2.1. They can be ignored for the most part.

```
--Parsing file E:\PROJECT\ANC_CO~2\EBY0884.CLD...  
Block 9399 S1 integer value 65534 (real value 0.000000) too big for short.
```

From the picture, you can see that somebody must have put 0.0 in S1 of block 9399 (a real number) instead of 0 (an integer). A real number does not have the same structure as an integer, so DBDOC tells you about it. We do not know if it works, but some errors of this type have found actual problems for clients.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.293 Too big for unsigned short [ERROR 10024]

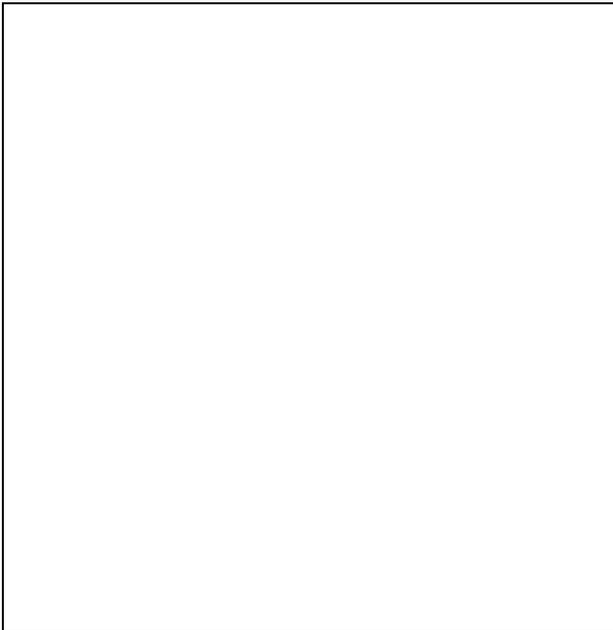
```
Too big for unsigned short: Module 1,01,02 Block 12345 (FC23 S3) integer value  
65536 (real value 1.0)
```

This message means that a CLD file has a value for a specification that is apparently illegal. We have found this as uncorrected damage from early versions of Composer after the verify with update feature was used.

Most of these messages occur when you switch to Composer version 2.1. They can be ignored for the most part.

```
--Parsing file E:\PROJECT\ANC_CO~2\EBY0884.CLD...  
Block 9399 S1 integer value 65534 (real value 0.000000) too big for short.
```

From the picture, you can see that somebody must have put 0.0 in S1 of block 9399 (a real number) instead of 0 (an integer). A real number does not have the same structure as an integer, so DBDOC tells you about it. We do not know if it works, but some errors of this type have found actual problems for clients.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.294 Too many fields [INTERNAL 223]

```
Fewer substitutions than fields: 0<1
```

DBDOC has encountered a situation where there are fewer substitutions than fields. This message indicates that the type of database processing that has been chosen is inappropriate in some way. Each type of special database processing depends on certain fields being found in the database. If they are not, the intended linking cannot work. Please check the database processing you have chosen. If it seems correct, send us the generated error.zip and a zipped copy of the database giving these messages. We regularly add to the things we support.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-BUILD DATABASE FIELD.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.295 Too many outputs [ERROR 137]

```
Too many outputs: expected 3 got 5 for Module 1,01,02 Block 1001 FC 34
```

This message means that a function code has been encountered that does not have the normal number of outputs. We have encountered this in the case of Ladder files.

The mismatch among versions of .LAD files probably means that files have been built that should not have been. DBDOC builds all the files of a particular extension in the folder, so you should move invalid ones into an ARCHIVE type subfolder. This will get them out of the DBDOC build, while keeping them available for study if needed.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TOO MANY OUTPUTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.296 Too many segment control blocks [ERROR 305]

```
Specification of Segment Control Block Module 197,23,03 Block 1234 exceeds maximum SEGCRM count of 8. Blocks 3141, 5926, 5358, 9793, 2384, 6264, 3383, and 2795 already specified.
```

The maximum number of Segment Control Blocks in a module is 8.

The *Function Code Application Manual* says:

Each module can support up to eight segment control blocks.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC ILLEGAL SPEC VALUE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.297 Too many tag databases for tagmap [BUILD 249]

```
Too many (72) tag databases for tagmap; only the first 30 will be displayed
```

If you see this error, it means you have selected too many databases.

Take a look at your Composer or WinTools database selection with *BuildPlus / Tools / Project Options / Database Selections*. You should only choose large database taglists. Often, there are transfer lists of a small number of tags that should not be chosen. Also, there can be archived (dated) lists recording the state of the tags at some point. These should not be selected either.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD TOO MANY TAG DATABASES.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.298 Too small for float [ERROR 10028]

```
Too small for float: Module 2,01,02 Block 100 (FC222 S3) value 2.47956E-039
```

This message says that the value of the indicated specification Sn is too small to be a legitimate floating point number.

It is probable that the INFI 90 system has no actual problem with this. The message is being investigated and might disappear.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.299 Too small for short [ERROR 10023]

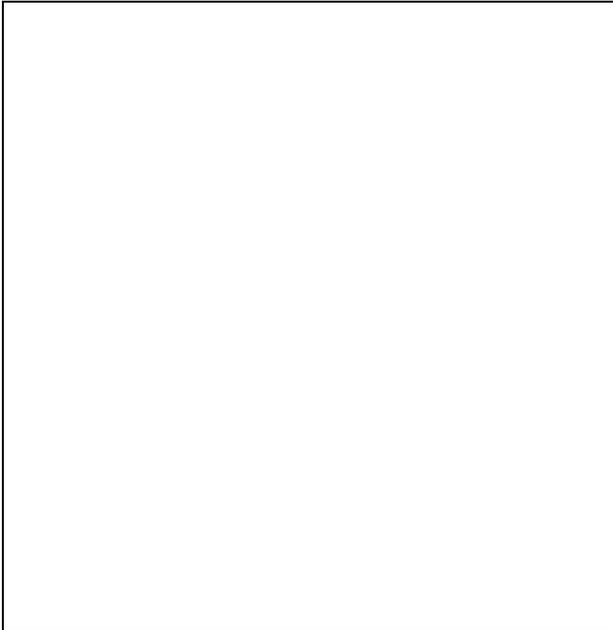
```
Too small for short: Module 1,01,02 Block 12345 (FC23 S3) integer value -32768  
(real value 1.0)
```

This message means that a CLD file has a value for a specification that is apparently illegal. We have found this as uncorrected damage from early versions of Composer after the verify with update feature was used.

Most of these messages occur when you switch to Composer version 2.1. They can be ignored for the most part.

```
--Parsing file E:\PROJECT\ANC_CO~2\EBY0884.CLD...  
Block 9399 S1 integer value 65534 (real value 0.000000) too big for short.
```

From the picture, you can see that somebody must have put 0.0 in S1 of block 9399 (a real number) instead of 0 (an integer). A real number does not have the same structure as an integer, so DBDOC tells you about it. We do not know if it works, but some errors of this type have found actual problems for clients.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.300 Too small for unsigned short [ERROR 10025]

```
Too small for unsigned short: Module 1,01,02 Block 12345 (FC23 S3) integer value  
-32768 (real value 1.0)
```

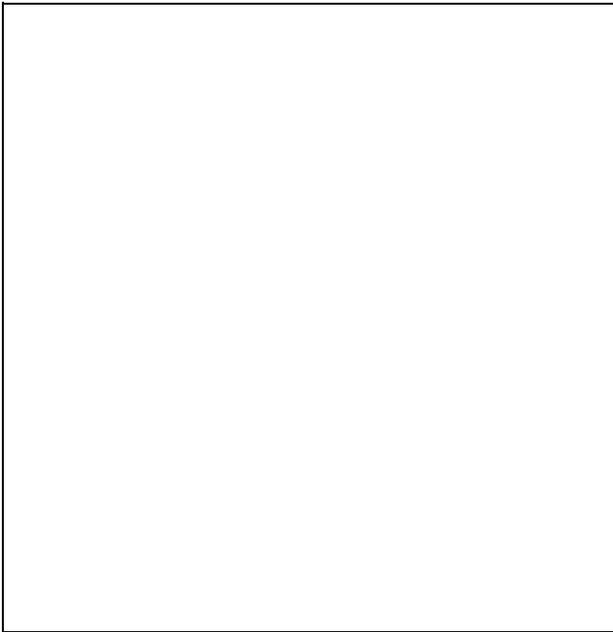
This message means that a CLD file has a value for a specification that is apparently illegal. We have found this as uncorrected damage from early versions of Composer after the verify with update feature was used.

Most of these messages occur when you switch to Composer version 2.1. They can be ignored for the most part.

```
--Parsing file E:\PROJECT\ANC_CO~2\EBY0884.CLD...  
Block 9399 S1 integer value 65534 (real value 0.000000) too big for short.
```

From the picture, you can see that somebody must have put 0.0 in S1 of block 9399 (a real number) instead of 0 (an integer). A real number does not have the same structure as an integer, so DBDOC tells you about

it. We do not know if it works, but some errors of this type have found actual problems for clients.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC COMPOSER SPECS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.301 Trend block is wired from constant [COSMETIC 258]

Trend block Module 2,10,31 Block 22544 (3VTTS001-PP) FC 66 is wired from constant block 5 (0.0)

The trend tag identified is wired from a constant block. It might be using processing time in the controller needlessly, as well as possibly being sent to the history system, using data bandwidth and often disk space.

A TREND: Analog Trend [FC66] block records the mean, minimum, maximum, sample, and sum for an input over a given time. This takes processing time, data bandwidth and disk space, to record the trends. When these trend blocks are attached to a constant, all data collected will also be that constant. It is a waste of resources to store this data.



In the above diagram, not only is a trend block wired to a constant (in this case 0.0), but three trend blocks are wired to the same constant.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC CONSTANT TREND TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.302 Trend block not used on any graphics or archived [CHECK 257]

```
Trend block Module 1,10,31 Block 22935 (CM3:3ABC) 0 not used on any graphics or archived
```

These trend blocks are fetched regularly from the controller. However, they are not in any trend. Thus, they waste both CPU time in the controller and bandwidth being brought to the console.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK UNUSED TREND BLOCK.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.303 Trend definition file not found [CHECK 196]

```
Trend definition Trend:0 not found (not configured) in FP_DYHisttrend
```

The PPB or Conductor NT graphic calls for another graphic which was not found. This is not really an error, but is a weakness for the operators using the graphic system. In Hyperview, go to "MISSING GRAPHICS / Index of Unconfigured Graphics" in the "System Information" Chapter to see where these graphics are

referenced.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-IGNORE INVALID TREND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.304 Trend does not have matching tag in any associated database [ERROR 107]

```
Trend TRENDTAG does not have matching tag in any associated database
```

This message means that during the scanning of the trend databases in a PCView SODG graphics system, a trend tag was found that was defined with a tag that did not match, or it does not have matching tag in any associated database.

In PCView, a trend tag is only meaningful if it is defined with the same block as its source tagname.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC PCV TREND MISMATCH.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.305 Trend index (tag) does not reference a trend in any associated trend database [ERROR 142]

```
Trend index 123 (tag Tagname) does not reference a trend in any associated trend database
```

This means that a trend index is used on a SODG graphic, but the trend index is not in the trend database. The trend graphic is not functioning properly – the trend graphic with the indicated trend index has no data.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNDEFINED TREND INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.306 Trend index does not reference a trend in any associated trend database [ERROR 143]

Trend index 123 does not reference a trend in any associated trend database

This means that a trend index is used on a SODG graphic, but the trend index is not in the trend database. The trend graphic is not functioning properly – the trend graphic with the indicated trend index has no data.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNDEFINED TREND INDEX.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.307 Trend referenced, but no tags configured in any database [COSMETIC 254]

Trend TREND:901 referenced, but no tags configured in any database in FP_DYtrendpb01 included by EBaUI_TRND01H_c

This message means that the Trend Group referenced has no tags in it.

This message shows a cosmetic issue that is probably not an error. It may be used to make the system tidier.

This error type was originally reported in error file **DBDOC-COSMETIC EMPTY TREND.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.308 Trend tag not found in any database [ERROR 163]

Trend tag TREND not found in any database

The graphic refers to a tag that does not exist in your trend database.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC HISTORY TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.309 Trend tag referenced by two tags [ERROR 161]

Trend tag TREND referenced by TAG1 and DESC1

This is a rarely seen message. Please contact us for assistance.

The history database refers to a tag that does not exist in your tag database. The history tag has not been defined, although it is shown. Note that trend definitions are also affected. Since this history tag has no meaning, there is a problem with any trend group that references it.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC HISTORY TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.310 Trend tag refers to missing or invalid entry [ERROR 162]

Trend tag TREND refers to missing or invalid entry in history database

Only Conductor NT systems can produce this message.

The tag refers to an entry that does not exist in your history database. The history tag has not been defined, although it is shown. Trend definitions are also affected. Since this history tag has no meaning, there is a problem with any trend group that references it.

Example:

```
Trendcfg tag CBTE-01 refers to missing or invalid entry in history database.
```

The Trend error is caused by the fact that there is no Historical tag CBTE-01, which is what the Historical Tag should have been called. The trend group is thus flawed because it is calling up an entry that does not exist.

In the case of the message in the example, it is pretty clear that the Historical Tag named "CBTE01" should be modified to be "CBTE-01" and the Tag Atom entry corrected.

In the example, the Trend error refers, in fact, to an invalid tag, not a missing one. The other form of the error is, of course, to name a trend tag that does not exist at all. To resolve that error, you either eliminate the trend tag entry in the Trend Configuration section, or you correct the spelling of the Trend Tag entry in the Trend group.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC HISTORY TAG.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

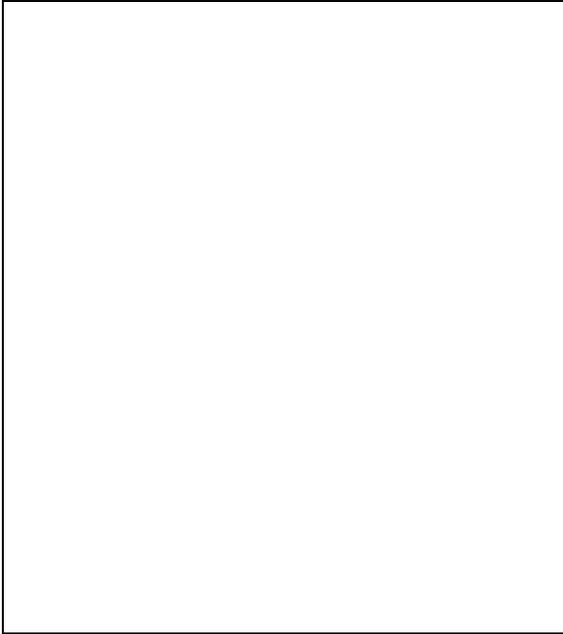
6.6.2.311 Trip on fault (defined I/O point bad) [ERROR 148]

`Trip on fault: Module 1,01,02 Block 1001 (FC 79) has S19 with value 100 (slave bad if defined I/O point bad)`

This message means that the value of the indicated specification will cause the module to trip if the slave has a fault, or if there is bad quality on any of the inputs.

In this case, a CISI/O: Control Interface Slave [FC79] value 100 will cause a trip if any defined I/O point is bad. Value 101 is used to mark the slave as bad if any defined I/O point is bad, but will not trip.

The "trip" specification controls if the module / controller should fail over to the redundant processor if a hardware fault is found. If the trip spec is set to "trip on fault", the module will stop working. If it is set to "continue operation" on fault, the module will smoothly transfer control to the redundant module. This is the normal mode of operation.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TRIP ON FAULT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.312 Trip on fault (will trip on failure) [ERROR 147]

```
Trip on fault: Module 1,01,02 Block 1001 (FC 79) has S4 zero (will trip on failure)
```

This message means that the value of the indicated specification will cause the module to trip if the slave has a fault.

The "trip" specification controls if the module / controller should fail over to the redundant processor if a hardware fault is found. If the trip spec is set to "trip on fault", the module will stop working. If it is set to "continue operation" on fault, the module will smoothly transfer control to the redundant module. This is the normal mode of operation.

Anytime an input / output function is found that is set to "trip on fault", it indicates that the system could be negatively impacted when it is not expected. You should make sure that there really is no redundant hardware present when you see this message.

AIS/FBS: Analog Input/Slave [FC132] has its own realities. In FC 132, the block is handled as a "chain" of three blocks, controlled by the specifications of the first block in the chain. Thus, any garbage value can be in S3 of a FC 132 block with no effect, unless that block is the first one in the chain (for FC 132 blocks, these messages will sometimes appear in the error filter file DBDOC-CHECK_AIS-FBS_TRIP_ON_FAULT.ERR).

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TRIP ON FAULT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

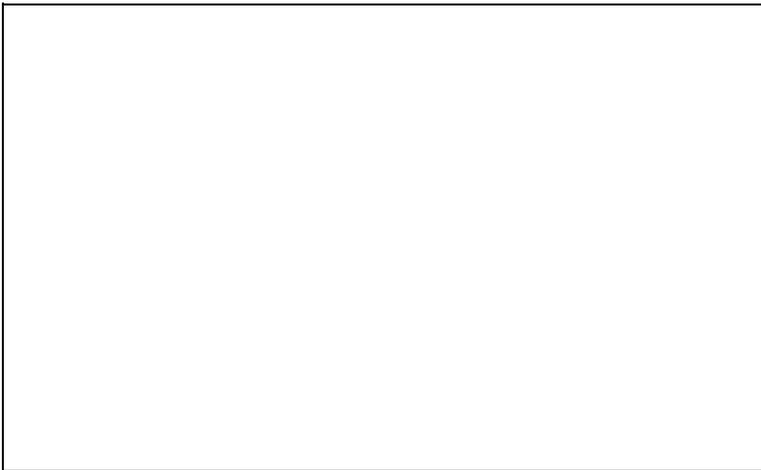
6.6.2.313 TSTALM checks subsequent block [CHECK 212]

```
TSTALM Module 3,11,06 Block 24080 checks subsequent Block 24084
```

This message tells you that a TSTALM block has a lower block number than the block whose status it is testing. This may often be ignored, because there are ways of making the block processing not follow block numbers.

These messages exist because in a redundant module switchover, the first scan of the module may have indeterminate or incorrect values for the blocks being tested, as it will not have been executed.

When the TSTALM block has a lower block number than the block it is testing, it executes first in the cycle, therefore it will test the value of the block, taken in the previous cycle.



The test block, (2855), is before the block being tested, (in this case 7011).

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK TSTALM SUBSEQUENT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.314 TSTALM does not test base block [CHECK 310]

TSTALM Module 1,01,02 Block 1010 does not test base block at Module 1,01,02 Block 1000

A TSTALM block does not test block N of the target FC, but instead some block N+x.

Newer versions of S+ flag an error if a TSTALM block points to other than block N of the target block. In that case, the module will not compile and cannot be loaded.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.315 TSTQ on import quality only [CHECK 070]

TSTQ Module 1,01,02 Block 1000 tests communications quality only on Module 1,01,02 Block 1001 (FC 42)

TSTQ block inputs must have quality. If they do not have quality, the test will not work. Examine logic to make sure that bad quality source values will be detected if that is intended.

Import blocks have bad quality because either:

- The source has bad quality, or
- The import process has failed, because:
 - ◆ Exception reports are not being generated or
 - ◆ Bus imports have failed.

In some systems, there is systematic checking that imported values are being received reliably. Use the information in this report to verify that bad quality source data will be taken into account in the logic.

Function codes that have quality or propagate the quality of their input are listed at Function codes with quality.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK TSTQ COMMUNICATION ONLY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.316 TSTQ tests block not propagating quality [ERROR 071]

```
TSTQ Module 1,01,02 Block 1000 tests Module 1,01,02 Block 1001 (FC 67), which is not propagating quality
```

This message tells you that a TSTQ: Test Quality [FC31] block is being used to test a signal that has no quality to propagate. The input will never be found to have bad quality and no action will happen that might be intended. This often shows a situation where an AO/L FC 30 block is being tested for quality, but the input signal has no quality to propagate.

Signals that have quality are represented in DBDOC as blue or purple lines on CAD and CLD diagrams. Function codes that have quality or propagate the quality of their input are listed at Function codes with quality.

This message sometimes appears where macros have been taken to implement logic, so it might not be an error. However, we often find such oversights as testing the quality of an Analog Transfer function block instead of its input, or the quality of a digital logic block rather than the inputs or outputs to the block.

```
--Generating Effective Quality report...
```

```
TSTQ Module 1,02,10 Block 6030 tests Module 1,02,10 Block 5615 (FC 67), which is not propagating qual
```

It is very important to note that this error will only cause a problem when a signal that is counted upon goes bad quality and the logic fails to detect that problem. It was missed in checkout and commissioning, or in subsequent work because it cannot be tested validly.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TSTQ.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.317 TSTQ tests block with no quality [ERROR 063]

```
TSTQ Module 1,01,02 Block 1001 tests Module 1,01,02 Block 1002 (FC 3 block N+1), which does not have quality
```

The TSTQ block inputs must have quality. If they do not, the test will not work. T: Analog Transfer [FC9] blocks and SQRT: Square Root [FC7] blocks are ones that are often tested in error.

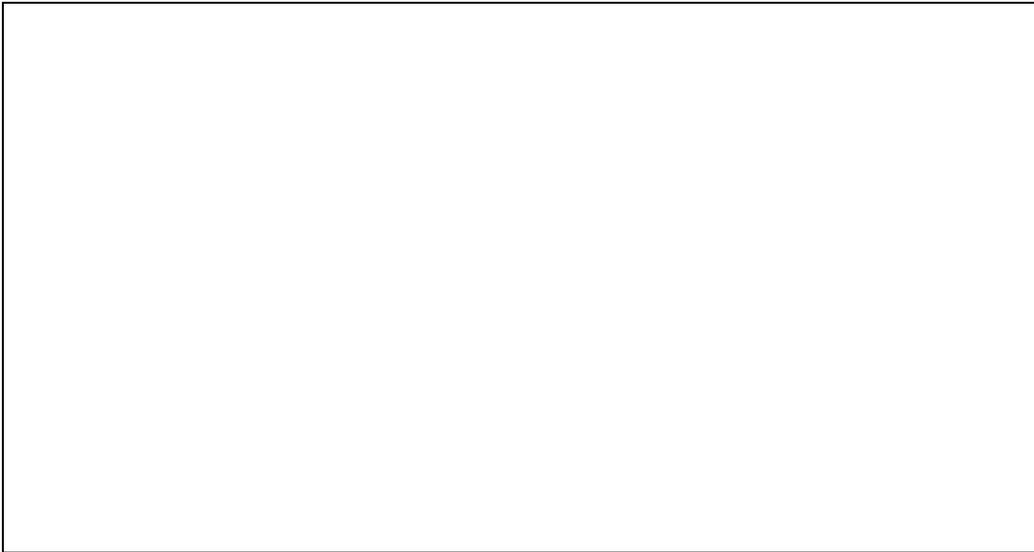
This message tells you that a TSTQ: Test Quality [FC31] is being used to test a signal that does not have the Quality attribute, or the signal being tested has no quality to propagate. The input will never be found to have bad quality and no action will happen that might be intended.

Signals that have quality are represented in DBDOC as blue or purple lines on CAD and CLD diagrams. Function codes that have quality or propagate the quality of their input are listed at Function codes with quality.

This message sometimes appears where macros have been taken to implement logic, so it might not be an error. However, we often find such oversights as testing the quality of an Analog Transfer function block instead of its input, or the quality of a digital logic block rather than the inputs or outputs to the block.

```
--Post Processing drawing T CCW Pumps Auto Trip  
TSTQ Module 1,02,10 Block 6031 tests Module 1,02,10 Block 5663 (FC 38), which does not have quality
```

It is very important to note that this error will only cause a problem when a signal that is counted upon goes bad quality and the logic fails to detect that problem. It was missed in checkout and commissioning, or in subsequent work because it cannot be tested validly.



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC TSTQ.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.318 TSTQ tests executive block [CHECK 156]

```
TSTQ Module 1,01,02 Block 1001 tests quality on executive block Module 1,01,02  
Block 3, which does not have quality
```

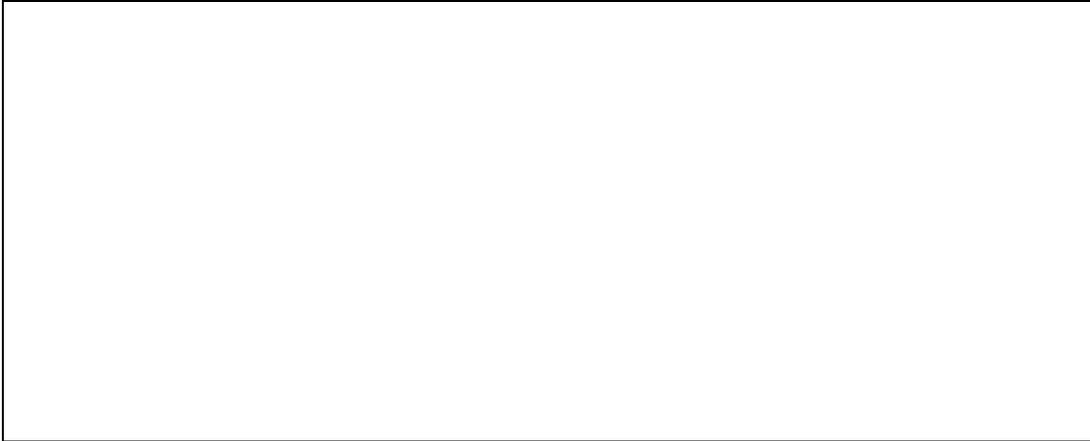
The TSTQ block inputs must have quality. If they do not, the test is useless. However, testing quality of executive constant blocks appears to be a common way of disabling unneeded logic.

Testing constants is generally visible and often intentional, acting as a simple place holder in multipurpose logic.

Function codes that have quality or propagate the quality of their input are listed at Function codes with quality.

```
TSTQ Module 20,09,31 Block 28552 tests quality on executive block 5,
```

```
which does not have quality [156]
TSTQ Module 1,06,31 Block 21464 tests quality on Module 1,06,31 Block 21333 (FC 52),
which does not have quality [156]
TSTQ Module 1,06,31 Block 21464 tests Module 1,06,31 Block 21333 (FC 50),
which does not have quality [156]
```



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK TSTQ.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.319 Type code cannot be rendered [INTERNAL 278]

```
Type Code DB_INFO in FILE1234.G cannot be rendered
```

An unhandled variable type has been detected. Any hotspot links or live values related to a variable with this type will not be rendered.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.320 Unable to find bitmap [BUILD 100]

```
Unable to find bitmap missing.bmp: using default size (200x200)
```

This message means DBDOC could not find a bitmap when the graphic was being compiled. Our DBDOC picture of the graphic may not be valid if you have missing bitmaps.

We suggest you find these missing bitmaps and then rerun the wizard to include the symbols into the system.

If you end up with numerous missing bitmaps, it is probable that your build is not configured properly.

This message usually shows an error in the build setup. Contact us for assistance configuring your build.

This error type was originally reported in error file **DBDOC-BUILD MISSING SYMBOL.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.321 Unable to parse node [INTERNAL 027]

```
Unable to parse node from file sample.txt
```

DBDOC was unable to parse the node.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.322 Unconnected wire end [ERROR 339]

```
Unconnected wire end at (4896, 4608)
```

This messages indicates a wire is defined with a missing end.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.323 Unexpected function code [INTERNAL 017]

```
Unexpected function code 25
```

DBDOC encountered an unexpected function code.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.324 Unhandled calculation [ERROR 326]

```
Unhandled calculation 12345 at 2300,1150
```

This calculation is not possible at build time.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC-INTERNAL MESSAGE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.325 Unhandled complex source configuration [INTERNAL 341]

```
Unhandled complex source configuration: 5 sources, 2 uses
```

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.326 Unhandled data after record [INTERNAL 234]

```
Unhandled data after 0 record: 0 bytes
```

A record has been found in a file that we do not know how to handle. This usually occurs when an obscure and rarely used function is encountered, or when a new version of the graphics system introduces new functionality.

Please send us the file and any related documentation you can find.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNHANDLED ENTITY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.327 Unhandled escape type [INTERNAL 236]

Unhandled Escape type: 0File A, File B

A record has been found in a file that we do not know how to handle. This usually occurs when an obscure and rarely used function is encountered, or when a new version of the graphics system introduces new functionality.

Please send us the file and any related documentation you can find.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNHANDLED ENTITY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.328 Unhandled line [INTERNAL 237]

Failed to handle line 12345

DBDOC encountered a file error.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNDOCUMENTED.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.329 Unhandled LMM spec value [INTERNAL 235]

Unhandled LMM spec value 0

A record has been found in a file that we do not know how to handle. This usually occurs when an obscure and rarely used function is encountered.

Please send us the file and any related documentation you can find.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNHANDLED ENTITY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.330 Unhandled pe header data [INTERNAL 233]

Unhandled pe header data: type 0 at offset 0

A record has been found in a file that we do not know how to handle. This usually occurs when an obscure and rarely used function is encountered, or when a new version of the graphics system introduces new functionality.

Please send us the file and any related documentation you can find.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNHANDLED ENTITY.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.331 Unknown AC 800M block type [INTERNAL 338]

Unknown AC 800M block type with color a66846 at 0,0

This message indicates an AC 800M block type was detected that is not currently handled by us.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.332 Unknown access code [INTERNAL 273]

Tokenization Unknown Access Code 0x1 in FILE1234.G at 0xABCD

An access code was encountered that is unknown to GMCL.

This error can trigger Dynamic Definition Undefined (277) errors in the same file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.333 Unknown atom [CHECK 047]

```
Unknown atom C_FOO for tag TAGNAME
```

This message means that a tag atom was used in Conductor NT or Process Portal B that we do not know about. We regularly add new tag atoms as we run into them.

As of DBDOC 11.2 Feature Update, there are new messages for tagatoms starting with "_" and ones with specious ")" character dangling.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK UNKNOWN ATOM.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.334 Unknown Bin file touchpoint type [ERROR 332]

```
Unknown Bin File xmlFile.xml Touchpoint Type TypeCodeName
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.335 Unknown Component Number [INTERNAL 270]

```
Tokenization Invalid Component Number 1 in ABC12345 at 1
```

A component number was encountered that is unknown to GMCL or invalid. The associated graphical element (polygon, circle, etc...) will not be rendered correctly.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.336 Unknown dynamic bar direction [INTERNAL 232]

```
Unknown dynamic bar direction -1 -- defaulting to left to rightFP_DYbargraph
```

Our understanding of dynamic bar graph display parameters in Conductor NT / Process Portal B graphics is incomplete.

If you get this message, please send us your .M1 files that generated this message with an explanation of what is special about the bar graph.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL DYNAMIC BAR.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.337 Unknown Exception [INTERNAL 053]

```
Unknown exception in file Broken.DWG -- skipping
```

This message identifies a file that we cannot process. Please send us a copy of it if it has no apparent problems in your system.

If the file DBDOC-CHECK_FILE_OPEN_FAILURE is generated, sometimes they may both be resolved if the project is rebuilt.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SKIPPING.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.338 Unknown logic state type [INTERNAL 245]

```
Unknown logic state type 0
```

We do not understand this logic state. Please check the file with the appropriate graphics tools and send the file to us if it appears to be valid.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNKNOWN LOGIC STATE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.339 Unknown mimic file touchpoint type [ERROR 330]

```
Unknown Mimic File xmlFile.xml Touchpoint Type ABC
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.340 Unknown picture type in ctx [INTERNAL 218]

```
Unknown picture type 2159955266 in VB.Image ctx at offset 0x000AC3DAUnknown  
Location
```

This message means that a call was found for a type of image that DBDOC does not yet support.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNKNOWN PICTURE TYPE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.341 Unknown record type [INTERNAL 240]

```
Unknown record type 0 at offset 0
```

The file contains records we do not support at the present time. Please send us a copy of the symbol file or graphic and any other information to help us add the support needed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNSUPPORTED RECORDS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.342 Unknown shape (missing library) [ERROR 042]

```
Unknown shape MISSING ignored (missing library SAMPLE)
```

This message says that a CLD/CAD sheet requires a shape from a user library that was not found.

Rebuild your DBDOC project with the library specified in Wizard to correct this problem.

This report gives you a checklist of shapes that are lost.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNKNOWN SHAPE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.343 Unknown shape (no user library) [ERROR 041]

```
Unknown shape MISSING ignored (no user LBR specified)
```

This message means the CLD/CAD sheet did not specify a user library, but the CLD/CAD needs a shape from a library.

To resolve this message, either fix the CLD/CAD sheet or exclude it from the DBDOC build.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNKNOWN SHAPE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.344 Unknown shape (not in library) [ERROR 043]

Unknown shape MISSING ignored (not in library SAMPLE.LBR)

This means that a shape was required by a CLD/CAD file, but was not found in the library accessed by the build.

This can result either from:

- a lost shape
- a local copy of the user library being needed. Rerunning the Wizard and deselecting the user library specification will take care of this.

In Composer systems, this message can mean that multiple user macro libraries that had evolved under WinCAD were not consolidated when the project was imported into Composer. The information in the CLD files might be misleading or incorrect.

This report gives you a checklist of shapes that are lost.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC UNKNOWN SHAPE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.345 Unknown subtype for block ed 50 [INTERNAL 242]

Unknown subtype 0 for Block 000 ed 50 -- defaulting to left-to-right dynamic bar

The file contains records we do not support at the present time.

Please send us a copy of the symbol file or graphic and any other information to help us add the support needed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNSUPPORTED RECORDS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.346 Unknown subtype on line [INTERNAL 241]

Unknown subtype on line 0: ei 0,0

The file contains records we do not support at the present time. Please send us a copy of the symbol file or graphic and any other information to help us add the support needed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNSUPPORTED RECORDS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.347 Unknown tag type [INTERNAL 259]

Unknown tag type RTU_SPARE

The reported tag has a tag type that we do not yet support.

Please let us know what function code it describes.

If DBDOC cannot support the type of tag in your system, please contact GMCL to have the problem looked into.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNKNOWN TAG TYPE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.348 Unknown type code [INTERNAL 276]

Tokenization Unknown Type Code 0x1 in FILE1234.G at 0xABCD

In parsing a file, a type code was encountered that is unknown to GMCL.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.349 Unkonwn mimic file type code [ERROR 331]

Unknown Mimic File xmlFile.xml Type Code TypeCodeName

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.350 Unparsable XML Line [ERROR 329]

XML file xmlFile.xml could not parsed line </Name/>

This message indicates an internal failure to parse an XML file, please report this to GMCL.

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.351 Unrecognized binary data [INTERNAL 274]

Tokenization Unrecognized Binary Data in ABC12345 at 1

Unexpected data was encountered while parsing a binary file.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL SPLUS-PGP.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.352 Unreferenced graphic file [INFO 214]

Unreferenced graphic: PERG_AREA14H.M1

This graphic is not linked from any other graphic in the chapter. It will not be called up by normal vectoring between graphics.

This shows that the graphic is present, just unconnected to other graphics in the chapter.

This message reports system information that is otherwise difficult or impossible to find.

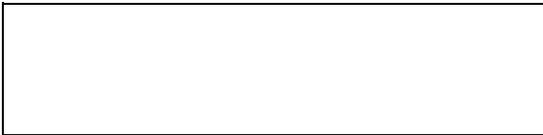
This error type was originally reported in error file **DBDOC-INFO UNREFERENCED GRAPHIC.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.353 Unreferenced graphic name [INFO 215]

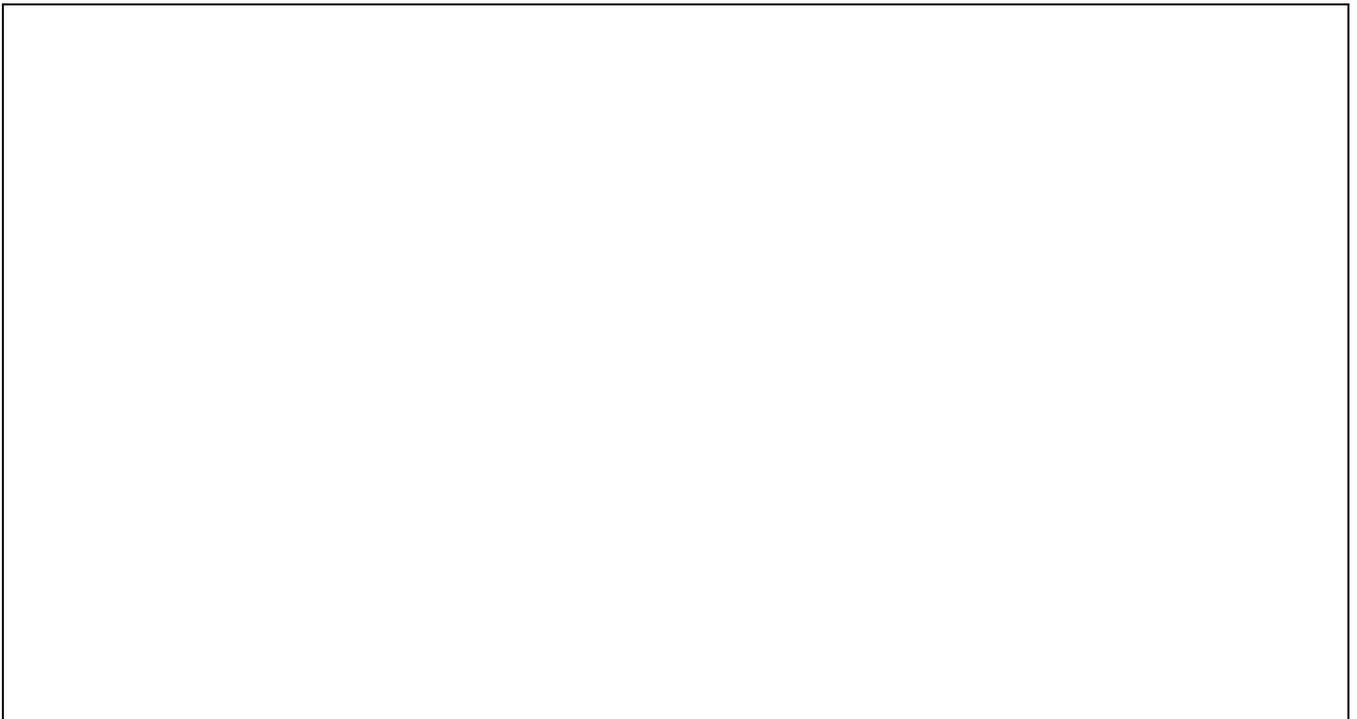
Unreferenced graphic: M-PLANTA.DIS - M-GENERAL1

The graphic is not linked to by any other graphic in the system. Unless its name is entered by hand, it cannot be called up.

When a graphic is called upon from the error browser, it fails to open to anything.



However, when the graphic is searched for, it opens without difficulty.



This shows that the graphic is present, just unconnected to anything.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO UNREFERENCED GRAPHIC.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.354 Unresolved connection [ERROR 344]

```
Unresolved connection to 11:_16M2080RUN.Out3
```

This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.355 Unsupported dynamic symbol subtype [INTERNAL 239]

```
Unsupported Dynamic Symbol subtype 0
```

The file contains records we do not support at the present time.

Please send us a copy of the symbol file or graphic and any other information to help us add the support needed.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNSUPPORTED RECORDS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.356 Unsupported FC used in CFC [10045]

```
CFC MY_CFC contains FC91, which is not supported in CFCs.
```

This CFC contains a function code that is not supported by CFCs.

This error type was originally reported in error file '. ***Most errors can now be viewed in the Hyperview Error Browser.***

6.6.2.357 Unsupported function [INTERNAL 238]

Unsupported function: Unknown Function

Data has been found in a file that we do not know how to handle. This usually occurs when an obscure and rarely used function is encountered, or when a new version of the graphics system introduces new functionality.

Send us the file and any related documentation.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNSUPPORTED FUNCTION.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.358 Unsupported graphic primitive [INTERNAL 217]

Unsupported graphic primitive ABBMSFlexGrid_TL.ABBOPCMSFG

This message means that a call was found for a graphic primitive that DBDOC does not yet support.

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL GRAPHIC PRIMITIVE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.359 Unsupported non-literal tag name [INTERNAL 343]

Unsupported non-literal tag name: ABC.XYZ.Name

This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file '. **Most errors can now be viewed in the Hyperview Error Browser.**

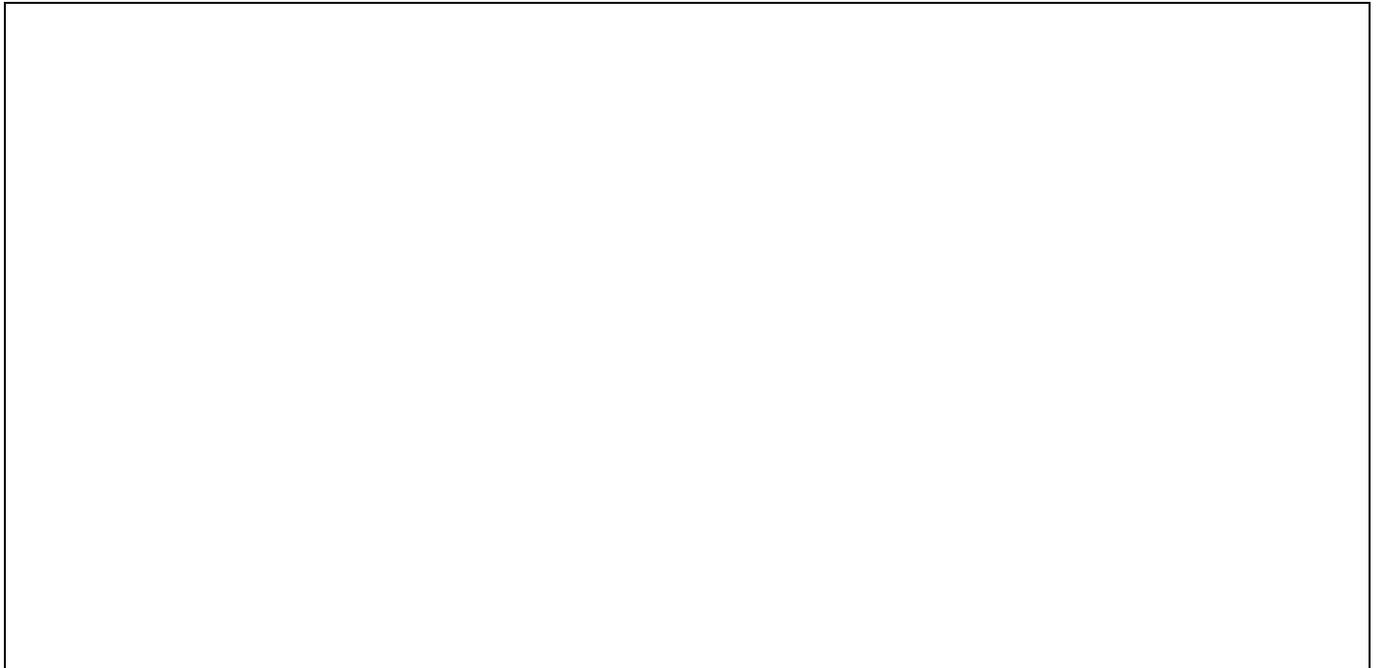
6.6.2.360 Unused extra inputs [INTERNAL 136]

```
Unused extra inputs for Module 1,01,02 Block 1001 FC 10: found 5, used 2
```

This message means that a function code has been encountered that does not have the normal number of inputs. We have encountered this in the case of Ladder files where there is a mismatch of .LAD file versions.

Example:

```
--Scanning LAD file 30603.LAD  
Unused extra inputs for Module 3,06,03 Block 203 FC 151: found 6, used 4.
```



This message shows an issue that the DBDOC build process did not handle. It can be reported to G. Michaels Consulting Ltd. in case it can be fixed.

This error type was originally reported in error file **DBDOC-INTERNAL UNUSED EXTRA INPUTS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.361 User macro registered multiple times [INFO 10039]

```
User macro MACRO registered multiple times
```

This message is designed to improve the development of DBDOC by asking you to collaborate with us. We would like to better understand duplicated macros and shapes, so if you can explain how the messages for your system got there, and anything meaningful, please let us know.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO MULTIPLE MACROS.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.362 Vertical text on drawing [INFO 050]

```
Vertical text NASM02-0 on drawing DRAWG25A.CAD
```

The diagnostic command "log_vertical_text" was used in the build to find text written vertically so that it could be examined.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO VERTICAL TEXT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.363 Wired gain spec is adapted [INFO 120]

```
Wired gain spec is adapted: S2 of Module 1,01,02 Block 1001 of 3 FC 1 adapted by  
Module 1,01,02 Block 1234
```

This message indicates that a wired gain spec is adapted. Verify that the intent of the logic is to have the gain controlled by an Adapt block.



This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO GAIN SPEC ADAPTED.ERR**. Most errors

can now be viewed in the Hyperview Error Browser.

6.6.2.364 X coordinate adapted [INFO 029]

X-coordinate adapted - S2 (1.00) of Module 1,01,02 Block 1001

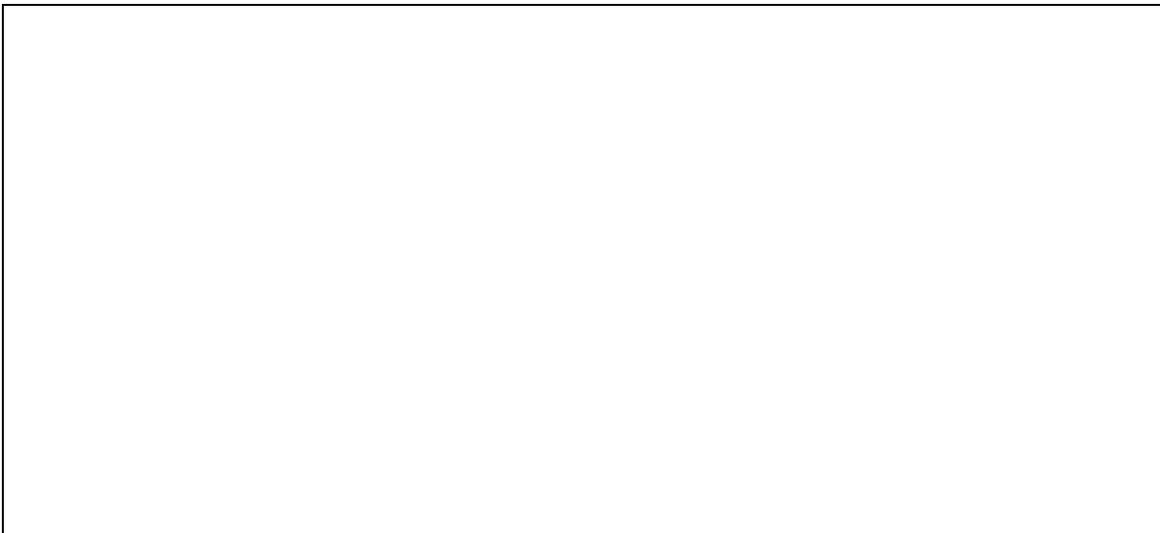
You may want to check that there are base or design values for the parameters of F(X): Function Generator [FC1] blocks that are adapted. If there are, the graph will be meaningful. Also, the order of processing (adapt block before function block) will not be as important.

This is not usually a significant error. We report them for these reasons:

- They are very hard to find and can be mysterious in operation.
- By making them visible to you, you can see if the actual implementation makes sense by looking at live data
- Since DBDOC will graph the function if the X and Y coordinates make sense, this file allows you to go to the work of finding good "design" values of the specifications so that the DBDOC presentation is useful to a user
- The intent of the adapted function generator should be documented, probably right there on the CLD or CAD sheet
- There might be an error in that a block number is wrong or was reused, with unintended consequences

Example:

```
--Scanning drawing 24305U8: As Shipped  
X-coordinate adapted - S2 (-20.00) of Module 2,43,05 Block 2125
```



You can see that the default value of S2 will affect the way the function is documented.

We suggest that you should figure out default values for all adapted coordinates. That way, DBDOC can show you what the design function generator shape is supposed to be. Also, live data will then show you

deviation from the design shape.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO ADAPTED COORDINATE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.365 X coordinates out of order [ERROR 033]

```
X coordinates out of order for F(x) Module 1,01,02 Block 1001
```

This message means that your x coordinates are not in ascending order. This may lead to strange graphs.

You can see the x coordinates on the block by viewing the specification of the block. View specs by pressing the "s" key, or from the menu item: View - Specs.

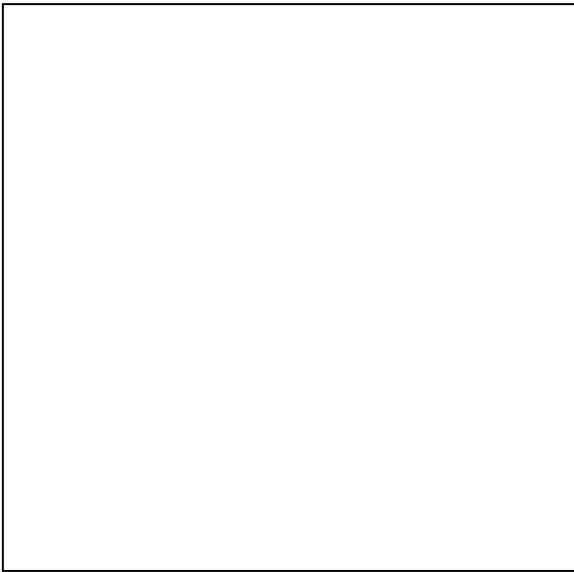
This message means that coordinates may be out of order for certain function codes. The x coordinates S2, S4, S6 etc. should have values in ascending order. The function generator implementation in the "controller" (the logic in a module) is very simple-minded. Essentially, the algorithm, given successive (x,y) pairs (x1,y1), (x2,y2), (x3,y3), (x4,y4), (x5,y5), (x6,y6) is as follows:

- If input is less than x1, output is y1
- If input is less than x(i), output is the linearized value using the line (x(i-1),y(i-1)) , (x(i),y(i)), which does not give the result people expect if the x coordinates are not in ascending order.
- If input is greater than x6, output is y6.

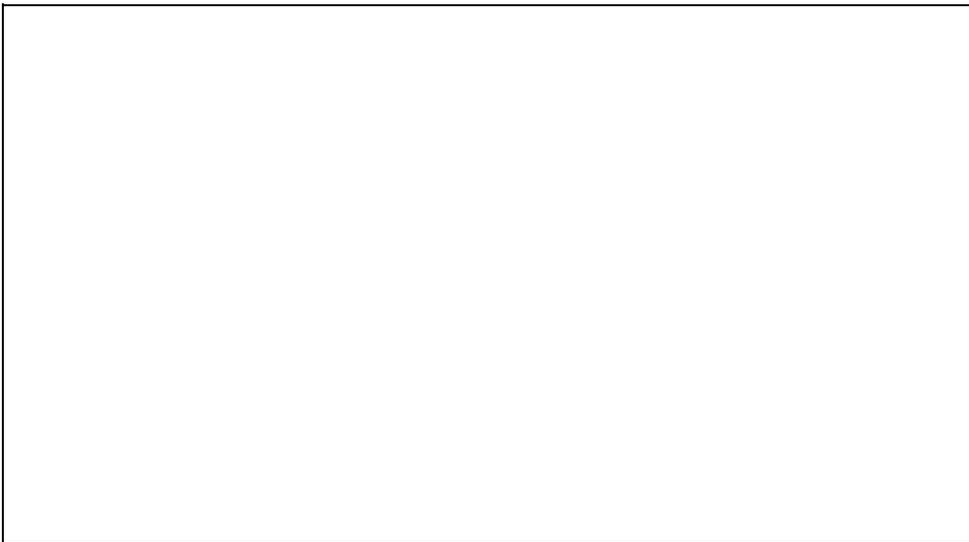
For example:

```
X coordinates out of order for F(x) Module 12,52,05 Block 3503.  
( S2, S3)          0.0          100.0  
( S4, S5)          10.0         660.0  
( S6, S7)          20.0         660.0  
( S8, S9)          300.0         500.0  
(S10,S11)          40.0          200.0  
(S12,S13)          50.0          100.0
```

The F(x) looks like this:



Here's another example:



This message is likely to show an error in the system configuration. It should be reviewed. The issue raised may be in unused logic or graphics, or have other realities that make it a non-issue.

This error type was originally reported in error file **DBDOC X COORDINATES OUT OF ORDER.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.366 XOR output forced to 0 [CHECK 281]

XOR output forced to 0 by duplicated input Module 1,01,02 Block 1001 fed twice into FC 101 Module 1,01,02 Block 1003

This block always outputs zero, because both its inputs are from the same source. It may not be clear that the logic has been disabled.

The message is intended to encourage documentation of forced logic, or removal of unused logic.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK FORCED OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.367 XOR output forced to 1 [CHECK 280]

```
XOR output forced to 1 in FC 101 Module 1,01,02 Block 1001
```

This block always outputs one, because its inputs are constant one and constant zero. It may not be clear that the logic has been disabled.

The message is intended to encourage documentation of forced logic, or removal of unused logic.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK FORCED OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.368 XOR output forced to complement input [CHECK 283]

```
XOR output forced to track NOT input Module 1,01,02 Block 1001 -- one input  
constant one to FC 101 Module 1,01,02 Block 1003
```

This block is acting like a NOT block, because one of its inputs is constant one. It may not be clear that the logic has been disabled.

The message is intended to encourage documentation of forced logic, or removal of unused logic.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK FORCED OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.369 XOR output forced to track input [CHECK 282]

XOR output forced to track input Module 1,01,02 Block 1001 -- one input constant zero to FC 101 Module 1,01,02 Block 1003

This block's output is always the same as one of its inputs, because the other input is constant zero. It may not be clear that the logic has been disabled.

The message is intended to encourage documentation of forced logic, or removal of unused logic.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK FORCED OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.370 XOR output forced to track input [CHECK 279]

XOR output forced to track input Module 1,01,02 Block 1001 -- only one input to FC 101 Module 1,01,02 Block 1003

This block's output is always the same as its input, because it only has one input. It may not be clear that the logic has been disabled.

The message is intended to encourage documentation of forced logic, or removal of unused logic.

This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK FORCED OUTPUT.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.371 Y coordinate adapted [INFO 030]

Y-coordinate adapted - S3 (1.00) of Module 1,01,02 Block 1001

You may want to check that there are base or design values for the parameters of F(X): Function Generator [FC1] blocks that are adapted. If there are, the graph will be meaningful. Also, the order of processing (adapt block before function block) will not be as important.

This is not usually a significant error. We report them for these reasons:

- They are very hard to find and can be mysterious in operation.
- By making them visible to you, you can see if the actual implementation makes sense by looking at live data
- Since DBDOC will graph the function if the X and Y coordinates make sense, this file allows you to go to the work of finding good "design" values of the specifications so that the DBDOC presentation is useful to a user
- The intent of the adapted function generator should be documented, probably right there on the CLD or CAD sheet
- There might be an error in that a block number is wrong or was reused, with unintended consequences

Example:

```
--Scanning drawing 24305U8: As Shipped  
Y-coordinate adapted - S3 (10.00) of Module 2,43,05 Block 2125
```



You can see that the default value of S3 will affect the way the function is documented.

We suggest that you should figure out default values for all adapted coordinates. That way, DBDOC can show you what the design function generator shape is supposed to be. Also, live data will then show you deviation from the design shape.

This message reports system information that is otherwise difficult or impossible to find.

This error type was originally reported in error file **DBDOC-INFO ADAPTED COORDINATE.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.6.2.372 Zero gain on input block [CHECK 262]

Zero gain value for S4 of Module Block 8143 FC 15 means that Block 8024 is being ignored

The indicated input block is being ignored because the gain applied to it is 0.0. It does not appear as though the block is doing nothing unless specs are viewed as well. This often indicates a value available for tuning that is currently unused, especially if the input block is a constant.



This message shows a potential error in the system configuration, although most such errors prove to be harmless. It should be reviewed and hidden, unless it shows an issue that should be addressed.

This error type was originally reported in error file **DBDOC-CHECK BLOCK WITH ZERO GAIN.ERR**. Most errors can now be viewed in the Hyperview Error Browser.

6.7 Other error checking techniques

Aside from error reporting, there are some techniques that can be used for system verification:

- Finding graphics links made by tag index instead of tag name
- Using WinMerge to compare build reports
- Using Config Parser
- Occasionally DBDOC_SUMMARY.ERR contains messages like

```
sqlite_step_exception in "E:\Project\EXPORTS\master.db" with 8,160,10,1620:  
Step 11: database disk image is malformed
```

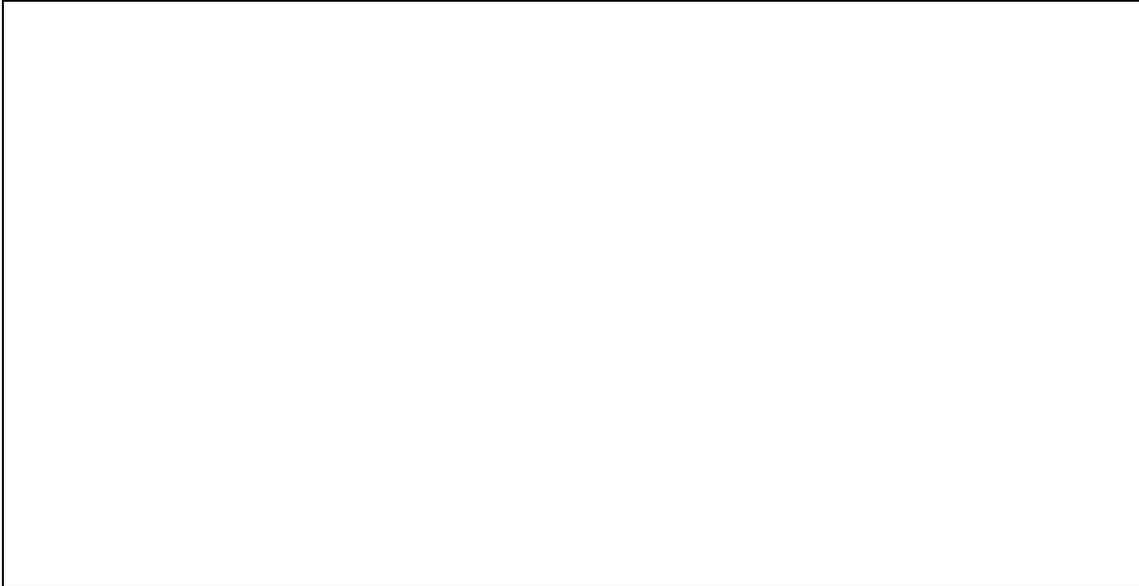
or

```
exportlookup.reset() returns 11 for E:\project\EXPORTS\master.db.
```

These messages are temporary, caused by changes in the structure of the master.db file, and after rebuilding the project, they are expected to disappear. For more information contact GMCL.

6.7.1 Finding graphics links made by tag index instead of tag name

In a SODG graphics system, search for the text "blank tagname" (without the quotes). If you find any, you know that your system has graphics links being made by tag index instead of tag name. This is not stable in the long term.



6.7.2 Using WinMerge to compare build reports

WinMerge is an open-source tool that allows you to specify two files or two directories and do a byte-for-byte file comparison. The result of the comparison is displayed in a window, showing you any changes that may exist between the two files or directories. This is a useful tool to compare differences in your error files between builds.

If you have WinMerge installed on your computer, BuildPlus will enable this option for you. You are given the choice to either Compare Files or Compare Directories. In either case, a dialog will appear where you can either type in or browse for the appropriate files/directories for comparison.

Note: Earlier versions of BuildPlus use WinDiff instead of WinMerge.

6.7.2.1 Compare Files

On the **Tools** menu, select **Compare with WinMerge** and click **Compare Files**.

(The default selections are the DBDOC_SUMMARY.ERR files of the two most recent builds.)

For each file you wish to compare, click **Browse**, navigate to the file, and click **OK**.

Once both file paths have been entered, click **OK**. This will launch WinMerge to show differences between the files.

6.7.2.2 Compare Directories

On the **Tools** menu, select **Compare with WinMerge** and click **Compare Directories**.

(The default selections are the error file folders for the two most recent successful builds.)

For each directory you wish to compare, click **Browse**, navigate to the folder, and click **OK**.

Once both directory paths have been entered, click **OK**. This will launch WinMerge and show which files are different within this folder. Double-clicking on the file name will show the differences within individual files.

7 Using CIULink

CIULink is a program that allows the user to view the data from the CIUMon interface program in different applications, such as Microsoft® Excel or Microsoft® Access, or can be used on its own as a viewer itself. By using the DDE (Dynamic Data Exchange) function in either Microsoft® application, you can request the information from CIUMon and view it in a format that you desire.

7.1 Running CIULink

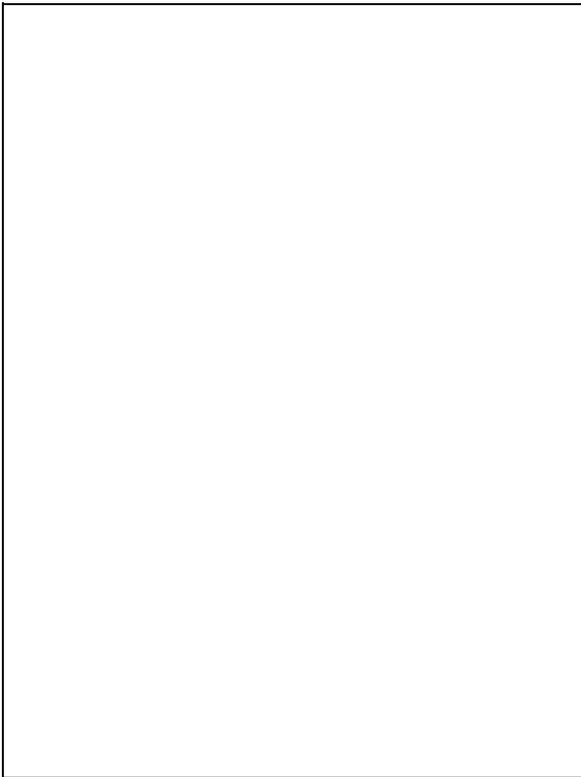
The following will outline the steps involved to ensure that CIULink will run correctly with your chosen application.

To start CIULink, on the Windows **Start** menu, open the **GMCL DBDOC** folder and click **Utility - CIULink**.

Alternately, you can find CIULink in the folder where DBDOC is installed. By default, this will be one of the following:

- C:\Program Files\GMCL\DBDOC\PROGRAMS (default for build kit on 32-bit operating systems)
- C:\Program Files (x86)\GMCL\DBDOC\PROGRAMS (default for build kit on 64-bit operating systems)
- C:\DBDOC\Hyperview\Programs (default for mini-Hyperview kit)

After you open CIULink, on the **File** menu, click **Set IP Address**. The following window will open:



Check to see that the "Loop Setup List" displays the correct IP Address for the computer running CIUMon. If it is not, then enter all the correct information in the appropriate fields and click **Insert**; the changes will be reflected in the "Loop Setup List". If you are connecting to an SPM01 or SPM02, you will have to enter in the SPM's specific PCU; if your CIU is of any other type, you should leave the PCU field blank (for example, in our setup Loop 1).

The Port Number is the port that is specified in CIUMon's configuration settings. The Local CIULink Port is the port on the CIULink computer that CIULink will bind to in order to send its requests out and listen for responses. It is best to leave this option blank and let CIULink check for the first available port within the specified range. For the majority of cases, you may use the default port range of 5000 to 5100. However, if you need to communicate across a firewall, only specific ports may be left open for communication, so you can modify the range to suit your needs.

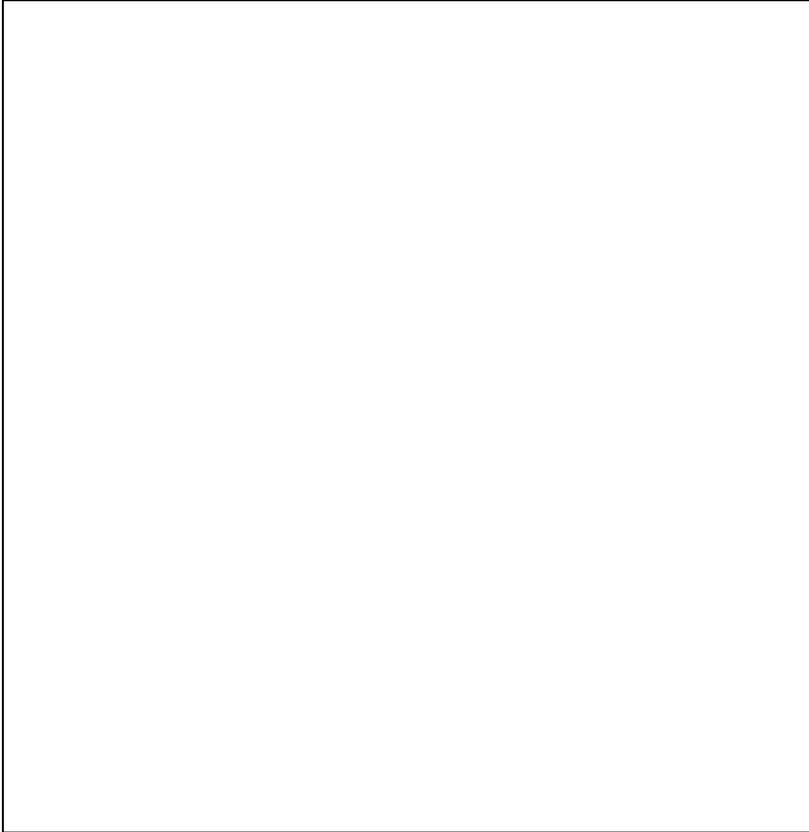
If you have more than one CIUMon running, continue entering all valid configurations and click **Insert** to add them to the Setup List. Once all additions and modifications have been completed, click **OK** to finish.

With both CIUMon and CIULink running, you can now begin running the application you wish to use to view the data.

7.2 Running CIULink as a Viewer

7.2.1 Configuring CIULink

CIULink on its own can be used as a viewer to observe outputs for various points in the system. If you have not run CIULink before, proceed through the following steps to ensure proper performance. After opening up the program and setting up the IP Address as explained above, the next step is to tell CIULink from which database files to get the correct module information. By selecting **Tag Monitoring** from the menu and choosing **Enter Database File Names**, the following window will appear:

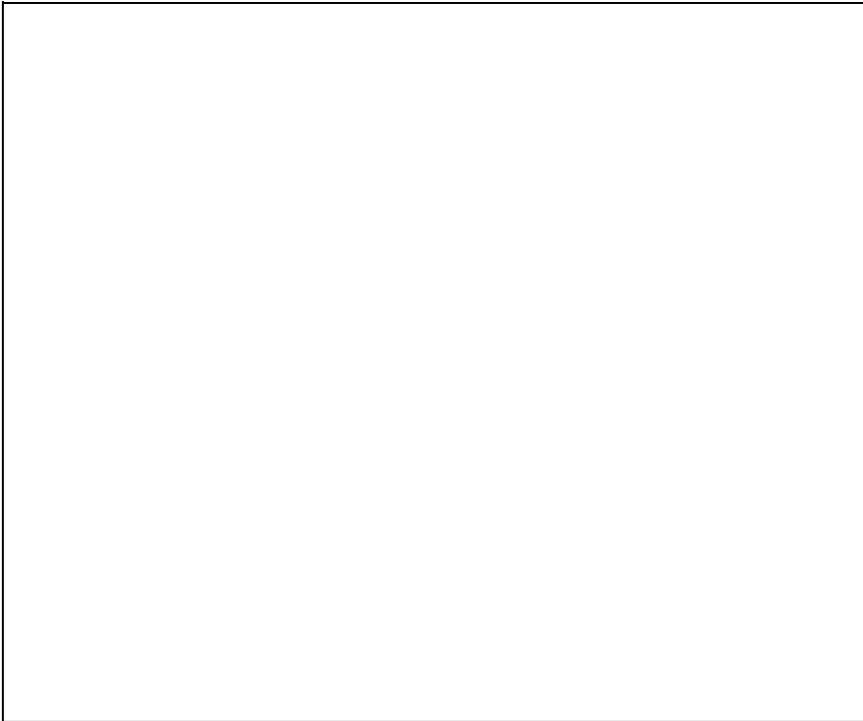


You can browse for the correct database files for which you are working on; when you have found and selected the correct file, click the **Add** button and the full path for the file name will appear in the **Current Database File Selection** area of the window. Add all the appropriate database files for your system to the list to ensure that all points will be covered. If you have made an incorrect choice, just click and highlight the incorrect file in the **Current Database File Selection** area and choose **Delete**; this will remove the database from the CIULink monitoring. When you are satisfied with all your choices, choose **Close**.

Note: Databases selected have to be in .dbf or .csv format. If the original database is not in this format, then select the databases extracted during the build process ? these can be found in the TAG subfolder of the project directory.

7.2.2 Adding Points for Monitoring

The next step is to tell CIULink which points you wish to monitor. By choosing **Tag Monitoring** from the menu and selecting **Tag Name Lookup**, the following window will open:

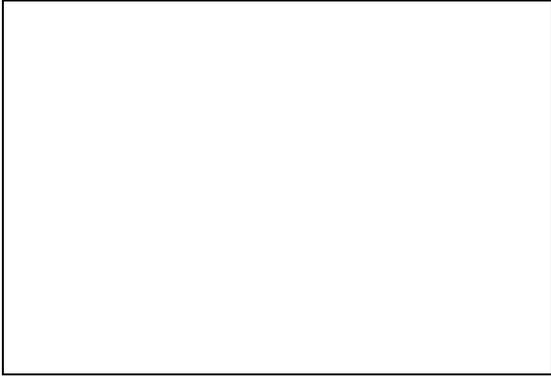


- Select **By Block** to enter the point you wish to monitor in *Loop, PCU, Module, Block* form (eg: Module 1,01,02 Block 4140), together with a description of the point so that you can easily reference it later. When you press **Lookup**, CIULink will check if the point exists within one of the databases you selected in the previous step. If it does, then it will be added to the **Results** box with its full description and the database file in which it was found; otherwise, the description that you type will be used.
- Select **By Tag Name** to enter a tag name from one of the databases you selected earlier. When you press **Lookup**, CIULink will look through the databases for the tag name you typed. If CIULink can find your tag, then it will be added to the **Results** box with its full description and the database file in which it was found.

Continue adding all of the points you would like to monitor. Should you wish to remove a point from the monitoring list, select the point in the **Results** box and click **Delete Entry**. When you are finished, click **Close** to continue.

7.2.3 Set Update Interval

Choose the interval at which to monitor the points by choosing **Set Update Interval** from the **Tag Monitoring** option on the menu. Another window will open with a section where you are able to set how frequently you wish for points to be monitored. It can be entered in values of hours, minutes, seconds or a combination of all three.



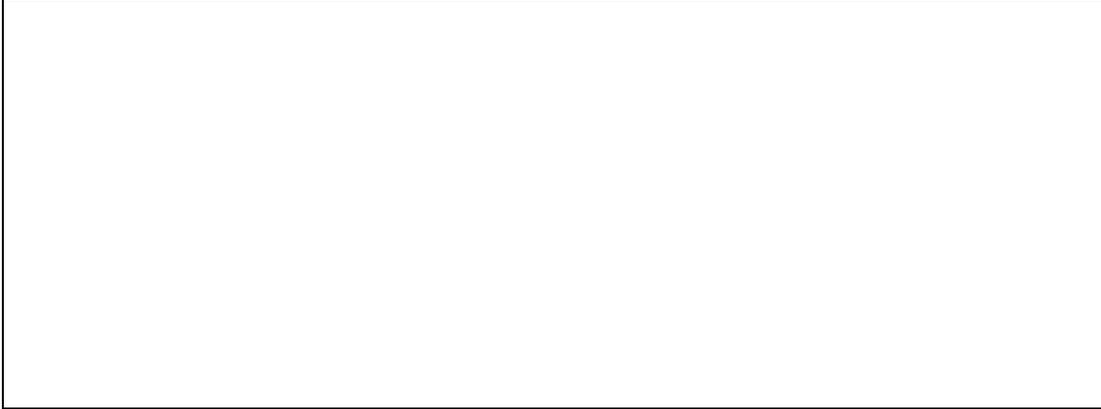
7.2.4 Selecting Viewing Options

You are able to choose the way in which the data on your selected points is displayed on the screen. Choose **View** from the menu and select either **Debug View** or **Export View**; a checkmark will appear beside your chosen view.



7.2.4.1 Debug View

In this particular view, data is displayed as a series of requests that CIULink puts out to CIUMON and the responses it receives back. The outgoing messages are shown in **blue** and incoming responses are shown in green.



In this example, we are requesting information for Module 1,01,01 Block 1591 and Module 1,01,01 Block 4140 and have subsequently received three responses back from CIUMON with outputs from those points in our system.

7.2.4.2 Export View

In this view, data is shown in a tabular format and its saved log is more compatible for import into other applications, such as Microsoft® Excel. The log that is created from this view (discussed below) uses commas as delimiters to distinguish between the different columns.



Here, the responses are shown for Blocks 1591 and 4140. In this particular case, we are requesting data every 1 second; the timestamps reflect the response time of the first and last data points received. If a response has not been detected by CIULink by the time the next response is to be sent, **MISSING** will appear in the respective data position. **Note:** Additional data points may be added while monitoring; however, deletions are not allowed until monitoring has stopped.

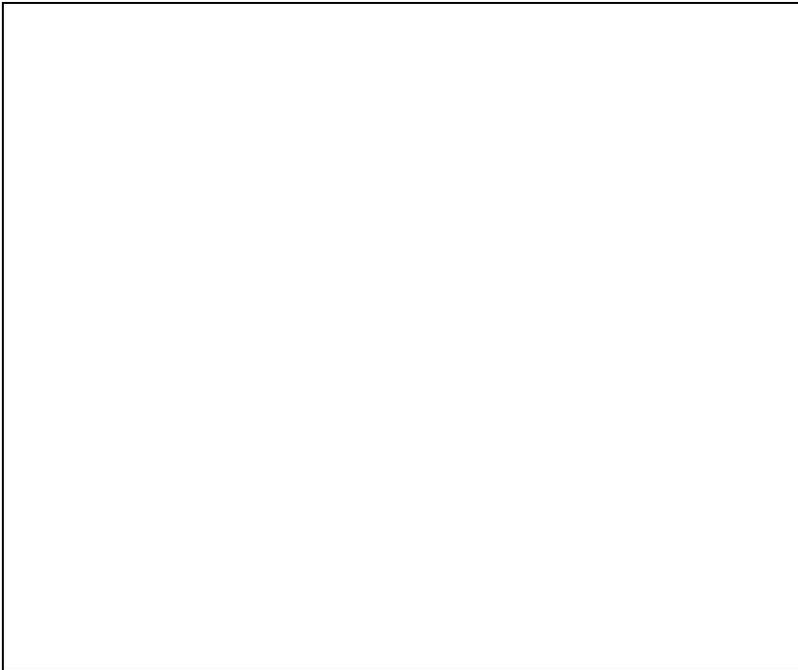
7.2.5 Monitoring data points

To start monitoring, select **Monitoring** from **Tag Monitoring** on the menu bar. A check mark will appear beside the option when CIULink is in monitoring mode. To stop monitoring, again select **Monitoring** from the menu; the check mark will disappear and monitoring will cease.

7.3 Logging with CIULink

In some cases, you may wish to log the responses from the network for the data points you are monitoring for future reference or manipulation in some other application. You can create a log of past responses or log responses as they are received by CIULink.

By selecting **Logging** then **Log Data** from the menu you are presented with the following choices:

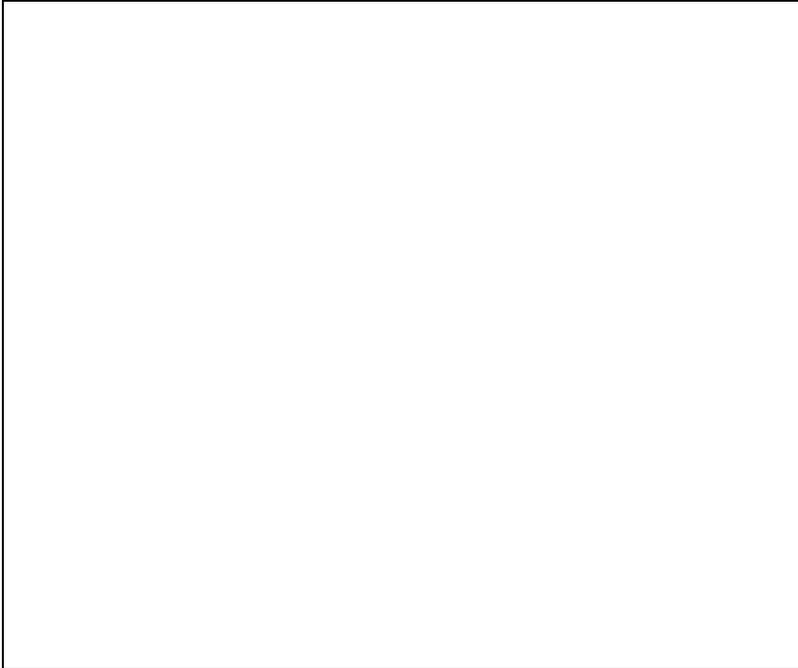


7.3.1 Log New Data as It Arrives

This option allows you to log data continually as responses come in until you turn logging off. The **Log Data** selection in the **Logging** option on the menu will appear with a check mark beside it to indicate that continuous logging is being performed. To turn this off, just select this option again from the menu; the check mark will disappear and writing to the chosen file will stop.

7.3.2 Log Data Already Collected

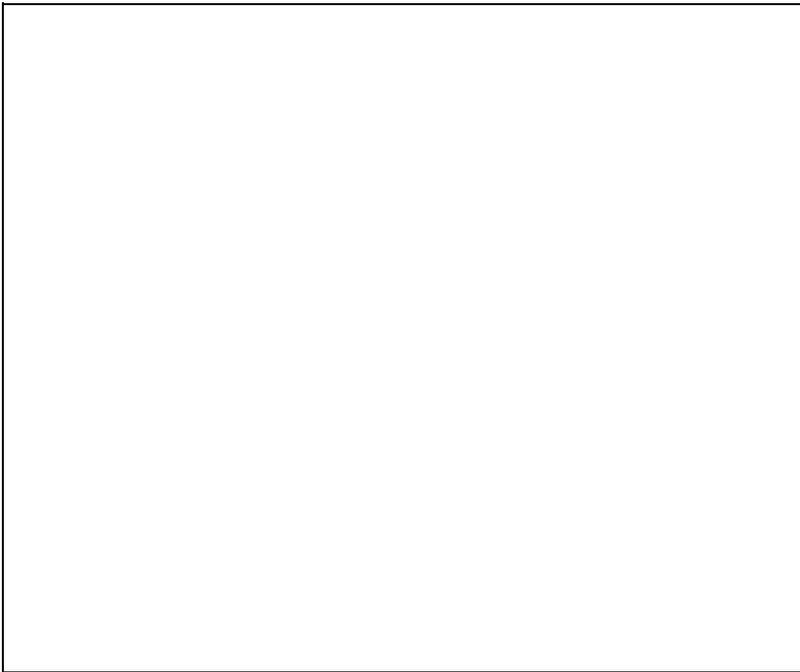
Choosing this option allows you to create a log of responses CIULink has received. You can either choose **All lines** to log all lines that have been collected so far or choose to enter a custom value for the most recent number of lines.



This will show you the present number of lines available for printing and allow you to enter the number of lines you wish to write to your log; in this case, there have been 1391 responses received by CIULink since monitoring has begun. (Please note that the buffer has a maximum of 10000 lines; any monitoring beyond that will result in the earliest data being lost for logging and display.) Enter the number of lines which you wish to write to the log and click on **OK**.

7.3.3 Choosing where to save logged data

You are required to choose a path in which to save your logged data. If you know the path you wish to save data to you can type it in manually.



Hitting the **Browse** button will let you navigate to a directory where you wish to save your data to. An example of the Browse dialog box is shown in the image below.



Once you have chosen a directory to log your data to it will be saved to a file in the directory specified.

If you are in **Export View** your data will be saved to a CSV (Comma Separated Values) file. This data displays very nicely in Microsoft® Excel. The default name for your log file is as follows:

CIULink_<Taglist Name Being Monitored>_<Date if option is chosen>.csv

For example,

CIULink_Bichem Tag Set_2-26-2008.CSV

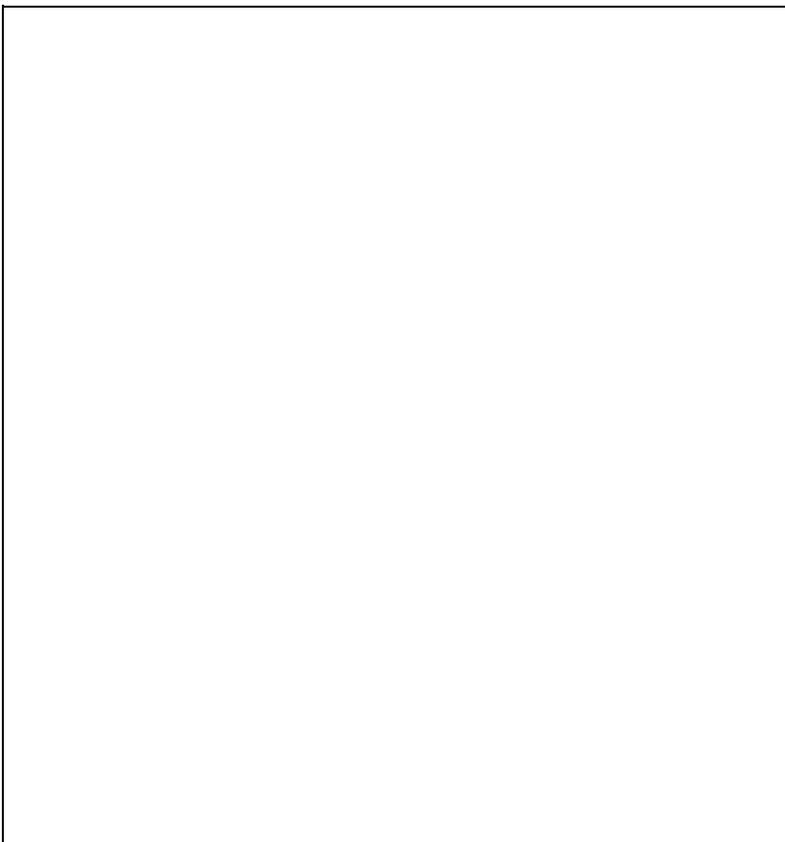
To have the date appended to your filename, click the **Append the date to the log's filename**.

The **Reset** button sets the path in the textbox back to the default and also selects the **Append the date to the log's filename** checkbox.

There is an additional option at the bottom of the dialog box, **Delete all existing data from the specified log file**. Checking this option will clear all values present in the log file selected.

7.3.4 Logging Options

Clicking the **Logging Options** button will allow you to have more control over your logging. An example of the dialog box is shown below:



At the top of the **Logging Options** dialog the directory and filename of the file being logged to are shown.

Log File Size is the next section shown on the screen. Entering a number in the box will set a maximum size for your log file. Your values will stop logging to this file once it reaches the maximum size you have specified. If you would not like to set a maximum file size for logging you can click the **Check here for unlimited logging**.

The **Export View Logging Options** are applicable to Export View only. They let select or deselect data to be included in the log file.

7.3.5 Reading CIULink logs

To view the logs created in CIULink: on the **Logging** menu, click **Launch Windows Explorer**.

It is easiest to view these logs in a spreadsheet program such as Microsoft Excel.

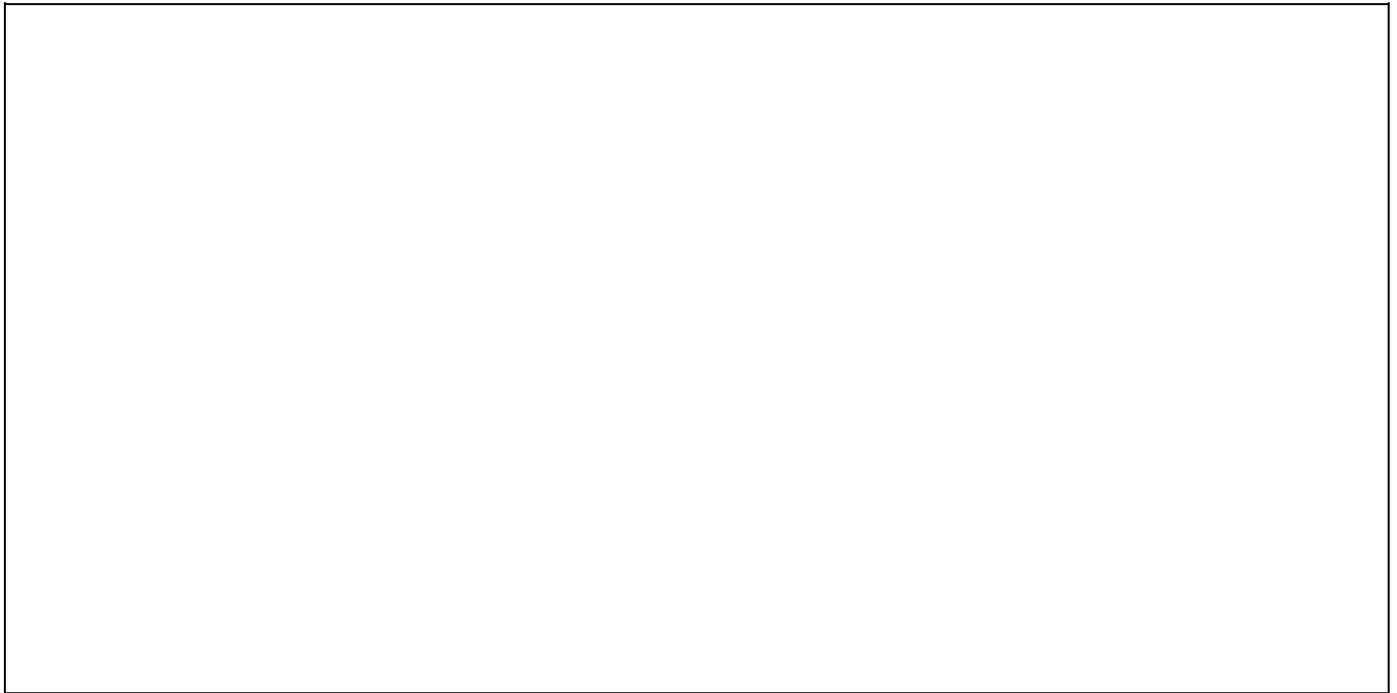
7.3.6 Understanding the format of CIULink logs

The data in CIULink logs is organized into columns for each of the blocks that you are monitoring. The first three rows contain headings to indicate what data you can find in each column.

The data in these first three rows is:

- The first line shows which blocks you will get data for in each column.
- The second line shows the tag names that correspond to these blocks.
- The third line indicates which columns contain the date and time

If you open the logs in a text editor, the columns are separated by commas. You can look at the first three lines of the log to figure out which column contains the data for each block, and then count the number of commas in the subsequent lines to find the data for that block. However, it is much easier to view the logs in a spreadsheet program such as Microsoft Excel, which will organize the data into columns automatically.



7.3.7 Dates and times in CIULink logs

The first two columns indicate the date and time when CIULink requested the data that appears in the log. If you need more precise timing information, you can use logging options to indicate the time when the data actually arrived in CIULink -- typically, this will be about a tenth of a second later. You can display the time when CIULink received the first data point in each row, the last data point, or each data point individually.

With this information, you can be certain that the data was current sometime between when CIULink sent out the request, and the time when CIULink received the data. It takes a few dozen milliseconds for a request for data to travel from CIULink to your CIU, and a similar time for the data to travel from the CIU back to CIULink, so it is not possible to be certain *exactly* when the CIU sent out the data. But knowing when CIULink sent out the request and when it received the data provides a short window that puts an upper and lower bound on this time.

7.3.8 Displaying dates and times in a spreadsheet

By default Microsoft Excel might not display the time columns correctly. To correct this:

1. Click the column heading to select the time column. (For example, if the Time of Request is displayed in column "B", click the "B" heading to select the column.)
2. On the **Format** menu, click **Cells**.
3. In the **Category** box, click "Time". Choose a time format from the **Type** box.
 - ◆ If you selected **Include Date in field** in CIULink's logging options, scroll down in the **Type** box to choose a format that displays both the date and the time.

7.4 Loading, Saving and Deleting Tag Lists

You may have instances in which you wish to monitor different sets of points at different times. CIULink allows you save different lists of points so that you do not have delete and enter your different points each time you want to switch between data points. You can also export your list of points so that you can view them on another computer, or in Hyperview's Watch Window. All of the tag list options are located under the **File** menu.

Specify the points you want to monitor in CIULink with the **Tag Name Lookup** option under **Tag Monitoring**.

Save your points to different lists and load the appropriate one for viewing when desired. After you have entered in a set of points using the **Tag Name Lookup** dialog, save them in a list by selecting **Save Tag List** from under the **File** menu.

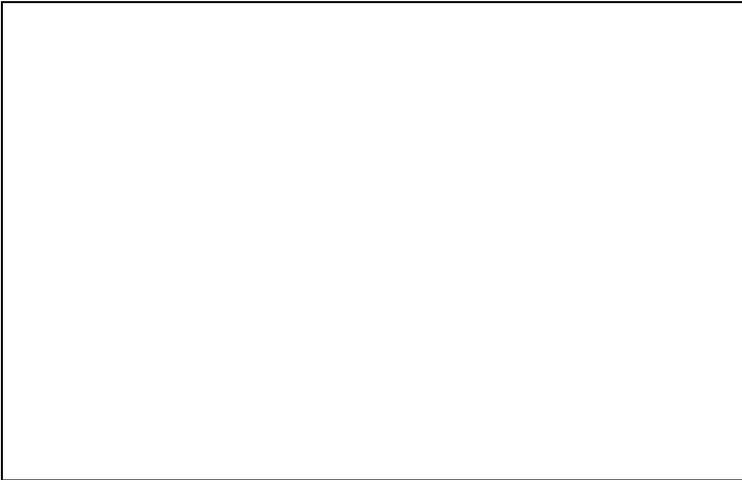


Shows the name of the currently loaded list at the top as well as highlighting it in the list. Choose an existing list name to save your current set of points to or type in a new name in the edit box provided.

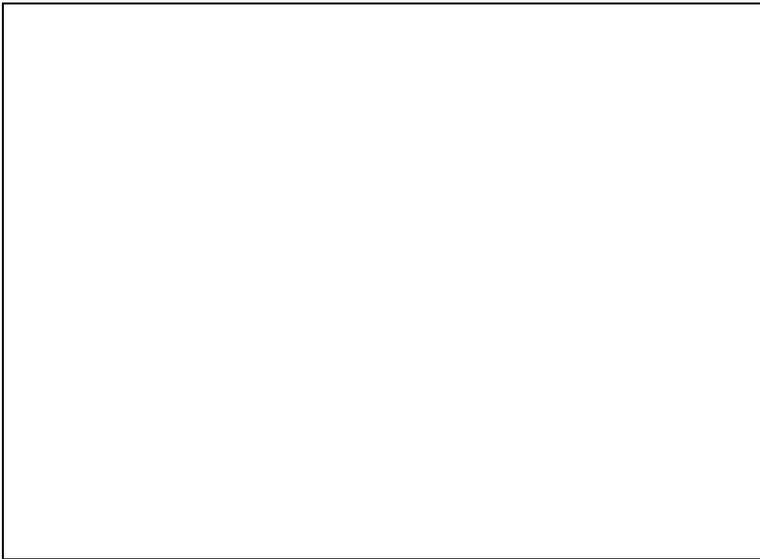
Switch between sets of points by selecting **File | Load Tag List**. This option will only be enabled if you are not currently monitoring in CIULink. Choose the appropriate tag list from the list that appears in the dialog. There is a special option in the list named (Blank Taglist) - this will let you start a fresh blank tag list in CIULink.

Delete obsolete lists of tags with the **Delete Tag List** menu option. Again, this option will only be enabled if monitoring is turned off.

Choose the list that you want to delete and click the **Delete** button. The actual deletion of the chosen list(s) does not occur until you press the **OK** button.



On all tag list dialogs is the **Review List** button that gives you the chance to see what points have been included in a particular list. Highlight the name of a tag list in a given dialog box and click the button. Another window will open up showing the points contained within the chosen list:



7.4.1 Export Tag List

You can save any of your tag lists to a file so that you can take them to another computer or import them in Hyperview's Watch Window. There are two ways to export your tags:

1. If you want to export the tags that you are currently viewing, select the **Export Current Tags** option from under the **File** menu.
2. Launch either the **Load Tag List** or **Save Tag List** dialog and select the tag list that you want to export from the choices that appear. Click the **Export Taglist** button.

In both instances, CIULink will launch the standard Windows **Save As** dialog to let you specify the location and filename to which to save the tag information.

7.4.2 Import Tag List

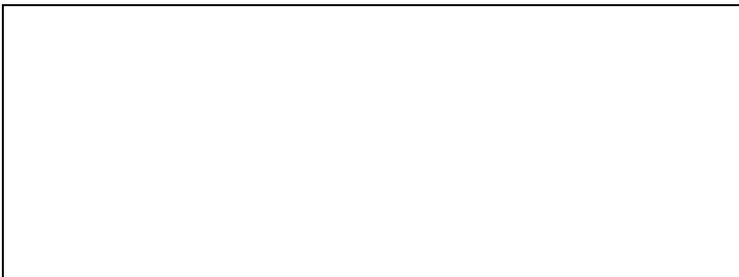
To import a tag list on a different computer, start up CIULink and select the **Import Tag List** option from under the **File** menu. The following dialog will appear:



Learn about importing a tag list into Hyperview's Watch Window.

7.4.3 Significant Digits

Specify the number of significant digits that you want in the display in Export View. Select **Set Significant Digits for Export View** from the **Tag Monitoring** menu and enter the appropriate number of digits in the dialog box that appears. You can enter 0 (zero) to remove the significant digit specification.



For example, if we enter 3 for our significant digit specification, then in our Export View the value "1329" will appear as "1.33E+003".

Begin monitoring by selecting Monitoring from the **Tag Monitoring** menu. A checkmark will appear beside the option if monitoring is on.

Check here to have no limit on significant digits: Selecting this option will not limit the amount of digits collected by CIULink.

7.5 Running from Microsoft® Excel

7.5.1 Creating a new workbook

In Excel, click in the cell that will be displaying the data and initiate the DDE function. For example, to observe the output from Module 1,01,02 Block 4185, enter the following formula into the cell:

```
=CIULink|Output!'Module 1,01,02 Block 4185'
```

If both CIUMON and CIULink are running, Excel will begin displaying the data requested. It is important to note that the syntax of the DDE function must be entered exactly as shown above in order for the link to work.

Note: If you are using LibreOffice Calc, you can use the following formula to observe the output from Module 1,01,02 Block 4185:

```
=DDE("CIULink","Output","Module 1,01,02 Block 4185")
```

7.5.2 Export lists from CIULink and import them into Excel

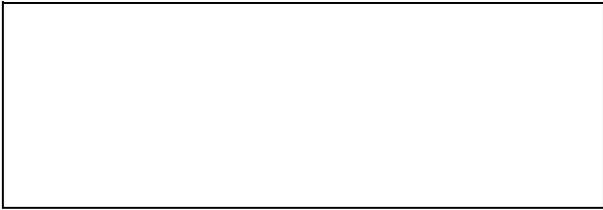
CIULink can automatically create a file that you can open in Excel to monitor your data.

To create a file to monitor the points that you are currently looking at in CIULink, from the **File** menu, select **Save Current Tags to Excel DDE File** and choose a location for your file. When you open that file in Excel, you will see data automatically appear for your points.

If you would like to create a file for one of your saved tag lists, instead of the tags you are currently looking at, simply select either **Load Tag List...** or **Save Tag List...** from the **File** menu, highlight the tag list that you want to look at in Excel, and press **Excel DDE Export**.

7.5.3 Running from a workbook with existing DDE function

If you have previously created an Excel workbook with the correct DDE function call(s) in it, upon re-opening the workbook the following window will appear:



Again, with both CIUMON and CIULink already running, clicking on "Yes" will begin the updating of the data requested.

7.6 Running from Microsoft® Access

One way to view the data in Access is to create a form with the text control that has a DDE function as its control source. For example, to observe the output from Module 1,01,02 Block 4185, enter the following formula into the **Control Source** line of the properties for the text box:

```
=DDE("CIULink","Output","Module 1,01,02 Block 4185")
```

When the form is opened, the data requested will be updated in the text box, provided both CIUMON and CIULink are running. For the link to work, make sure that the DDE function call is entered in the same syntax as shown above.

8 DBDOC Installation and Setup

8.1 Overview of installing, updating, and administering DBDOC

Welcome to DBDOC!

- If you are installing DBDOC for the first time, see **Installing DBDOC**.
- If you are updating an existing DBDOC installation, see **Updating your DBDOC installation**.
- For information about the DBDOC dongle, see **Working with the DBDOC security dongle**.
- If you would like to set up DBDOC for multiple users, see **Sharing Hyperview and project files over a network**.

Please contact us with any questions or installation problems and we will be happy to advise you over the phone.

8.2 Installing DBDOC

Welcome to DBDOC!

Use these instructions if you are installing DBDOC for the first time. If you are updating an existing DBDOC installation, see **Updating your DBDOC installation**.

If this is your first time using DBDOC, please contact GMCL. DBDOC is a powerful tool with many options and we strongly encourage you to let us guide you through configuring DBDOC for your system.

Contact us with any questions or installation problems and we will be happy to advise you over the phone.

The DBDOC system consists of a number of different applications which are all installed together when you do a DBDOC installation.

About basic DBDOC system installation

The major components of the DBDOC system are:

1. **BuildPlus**, the DBDOC interactive build system that guides you through the build process.
2. **Hyperview**, the DBDOC browser, used to view the system snapshot.
3. **CIUMon**, the server which fetches live system data from your CIU, Serial Port Module, or RoviSys Turbo and delivers it to **Hyperview** or **CIULink**.
4. **CIULink**, a simple tool for collecting data on groups of blocks in your system (not to be confused with the **Watch Window** which has similar functionality, but also trends the data. **Watch Window** is part

of Hyperview).

Normally the build system is installed in one location accessible to the DBDOC administrator. The Hyperview browser is installed in a central location accessible via a shared filesystem to multiple Hyperview users. One or more copies of the CIUMon live data server may be installed on various EWS machines that have CIU, serial port module, or RoviSys Turbo connections, and are accessible over the network to Hyperview users.

1. For instructions on installing the DBDOC build system, see Installing the DBDOC build system
2. For instructions on installing CIUMon, see Installing the CIUMon online data server
3. For instructions on how to share Hyperview with multiple users, see Sharing Hyperview and project files over a network
4. For instructions on installing Hyperview on a stand-alone computer, see Installing Hyperview on an individual computer
5. For instructions on installing support for 800xA graphics, see Installing the 800xA DLLs for Hyperview support of 800xA graphics.
6. For instructions on installing the DBDOC demo, see Installing the DBDOC Demo

8.2.1 Installing the DBDOC build system

Use the DBDOC build system to create a snapshot of your system that you can view in Hyperview, the DBDOC browser. To install the DBDOC build system

1. Choose a DBDOC build computer. This computer should run a supported operating system. For information about how increasing your computer's speed and memory will affect build times, see DBDOC build system hardware performance requirements.
 - Note:** The last fully supported update for Windows 2000 was version 10.4. If you need to run DBDOC on Windows 2000, please contact us for support.
2. If your computer has a previous DBDOC or CIUMon installation, see Updating your DBDOC installation.
3. Log on with an administrator account.
4. Insert the DBDOC DVD and run the installation wizard.
 - Note:** You can also find the Full DBDOC Install kit DBDOC_kit.exe in the system32\DIST folder of the DBDOC Installation DVD. You can also contact us to download the Full DBDOC Install kit, or if you have any questions.
5. Enter the password on the back of your DBDOC DVD case to begin the installation.
 - Note:** You will need to restart your computer to complete the installation.

Note: This procedure performs a full DBDOC install. You do not need to install the Hyperview browser or CIUMon online data server separately in order to use these programs on the DBDOC build computer.

If anything goes wrong with the installation, check for install.log in your user's Documents folder in a "dbdoc" subfolder, or in the root of the installation path, and contact GMCL with those logs and a report of what appeared to go wrong.

You will need administrator privileges to install the DBDOC build system this way. You do not need administrator privileges to run the DBDOC build system, but you do need either a USB security dongle or a software key.

- For information about the DBDOC dongle, see *Working with the DBDOC security dongle*.
- For information about setting up BuildPlus to create a snapshot of your system, see *BuildPlus*.

8.2.2 Installing the CIUMon online data server

8.3 Installation of CIUMon

Use the CIUMon online data server to connect DBDOC to your CIU and view live data. To install the CIUMon online data server

1. Choose a CIUMon computer. This computer should run a supported operating system, and should connect to a CIU. You can use RoviSys Turbo to share a CIU between DBDOC and another program like Composer.
2. Log on with an administrator account.
3. Run the CIUMon installer. Via the DVD Autorun wizard, simply choose "CIUMon only" on the Component selection page. You can also contact us to download this kit, or if you have previously installed DBDOC via the DVD you can find the CIUMon kits in the **DBDOC\PROGRAMS\CIUMon** subfolder of the DBDOC installation on your BuildPlus computer.

Note: If you use the included virtual serial ports, you may need to restart your computer to complete the installation.

Note: For Windows 2000, you must instead use **CIUMON_W2K_Kit.exe**; and for Windows NT 4.0, use **CIUMON_NT4_Kit.exe**. Please contact GMCL for installation help.

8.4 Privileges and Setup

You will need administrator privileges to use the CIUMon online data server.

For information about setting up CIUMon to connect to your CIU and view live data, see *Setting up CIUMon*.

8.5 Installing more than one CIUMon

It is also not uncommon to run more than one CIUMon data server on different machines. For example, one CIUMon may be connected to the CIU on a particular loop, and another to a Serial Port Module, or a CIU on a different loop. A CIUMon server might also be installed on an intermediate LAN in order to allow data to be transferred from the DCS LAN to the business LAN in a NERC compliant manner.

8.5.1 Sharing Hyperview and the M14 over a network

We recommend setting up Hyperview and the most recent project file in a shared location so that they can be used by anyone with access to your network. Usually remote users would have a shortcut to the shared Hyperview application.

- Sharing Hyperview from a central network location means that users have only a shortcut on their own computers that does not change when you update the Hyperview browser (for example when GMCL releases a new version of Hyperview). Even if you update the DBDOC version, you do not need to change their shortcut.
- Sharing the project file from a central network location means that users will always automatically see the most recent or appropriate version in their browsers. Most installations rebuild the project file regularly, and it is convenient for users to automatically see the most recent build.

8.6 Choose a location for the shared folder

In order to share Hyperview and the project file with users, you first need to choose a suitable shared location accessible to DBDOC users. DBDOC users will run this central copy of Hyperview using their desktop shortcuts.

It is much better for users to access Hyperview from a folder on their own LAN than from across a slow bridge or Wide Area Network (WAN). It is therefore best to choose a shared folder location on the same LAN as your target Hyperview users so they can run Hyperview via a high speed connection. In this context, 10 Mbps is slow, 100 Mbps fast enough and 1000 Mbps is extremely fast.

Note: If there are users who need to use Hyperview without having access to the shared network folder, they can install Hyperview directly on their computer. See the article **Installing Hyperview on an individual computer** for more information.

8.7 The Hyperview installer

The Hyperview installer contains the files you need to run Hyperview over a network. If you do not have a DVD, or you want to make sure you are using the latest version, contact us to receive access to download the kit from our website.

8.7.1 Option 1: Install Hyperview directly to a network share

If you have write access to a viable existing shared folder, you can run the installer to install Hyperview directly to that folder. This is the easiest method because the shortcuts created by the installer will already be valid and require no manual intervention.

8.7.2 Option 2: Install Hyperview locally first, then create a network share

If you are unable to install Hyperview directly to a network share, you may install it on your own computer, then make that folder shared across your network. Alternatively, you can run the installer locally and then copy the entire folder it creates onto another computer to share. In either case, we recommend contacting GMCL for assistance in setting a centralized Hyperview up in this manner, as there are many subtle problems this may cause.

8.8 Configure BuildPlus to automatically copy project files to the shared location

The last step in setting up sharing DBDOC involves BuildPlus. Every time you build a new project file, you will want that most recent version to be the one shared across the network. To ensure this, you have to configure BuildPlus to copy your project file to this shared directory after each build.

On the **Tools** menu, click **Project Options** and navigate to the **Hyperview/.Dbdoc Options** tab.



Copy the final .DBDOC file to the following locations

At the completion of a successful build, BuildPlus can automatically transfer a copy of the newly built project file to locations of your choice. Click **Browse** to add each destination directory. To remove a destination,

select it from the list, then click **Delete**.

The default shortcut created by the Hyperview installer automatically loads the latest project file placed into the `Data` directory. So for example, if installed Hyperview to the network path `\\Eliot\Applications\Hyperview`, you would want to set the copying location to `\\Eliot\Applications\Hyperview\Data`.

8.8.1 Installing Hyperview on an individual computer

If you need to use Hyperview when you are not connected to the network (for example on a laptop offsite), you need to install the Hyperview mini-kit on your own PC or laptop. You do **not** need administrator privileges to install the Hyperview kit.

1. Run the Hyperview installer.
 - ◆ Via the DVD Autorun wizard, simply choose "Hyperview only" on the Component selection page.
 - ◆ You can also download the latest version of DBDOC from our downloads page.
2. Copy the project files into the `Data` folder and keep them up to date.

To maintain your Hyperview installation:

1. Copy the current project files from time to time.
2. Get updated version of the programs folder as needed.
3. Please contact us if you have any questions or issues.

8.8.2 Installing the 800xA DLLs for Hyperview support of 800xA graphics

You will be prompted to install the 800xA DLL files during installation. **Only users with 800xA graphics require these files.**

It is a password protected installer - contact us and we will send you the password.

If you did not choose to install the 800xA DLLs on installation, you can install them at a later time with the following steps:

1. Browse to the 800xA DLLs folder in your installation directory (By default: *C:\Program Files\GMCL\DBDOC\PROGRAMS\800xA DLLS*). Alternatively, you can download the installer from our downloads page.
2. Double click 800xA_DLL_Kit.exe to start the installer.
3. Enter the password and click **Yes**.

The installer will guide you through the rest of the installation.

You can find more information on including 800xA graphics in your build on the Building in 800xA graphics help page.

8.8.3 Installing the DBDOC Demo

The DBDOC Demo allows you to get a basic view of how Hyperview works, without having to build your system. To ensure that you have the most up-to-date installation of the DBDOC Demo, do the following:

- If you had a previous installation, go to **Programs and Features** in the **Control Panel** and uninstall the old DBDOC demo.
- To ensure that all of the old demo files have been removed, you can delete the installation directory (*C:\Program Files\GMCL\Demo* by default) and any files it contains.
- If you have downloaded the demo kit from our website, open *DBDOC_DEMO.exe* and follow the Installation Wizard. If you are installing from a DVD that we have provided for you, then click **Install DBDOC Demo** on the DBDOC Installation DVD window to start the installation procedure. Leave the DBDOC Installation DVD window open until the installation has completed.

8.9 Operating system requirements and installation notes

This documentation will provide you with details related to installing DBDOC at your site.

- Getting installation help
- Installation notes
- The DBDOC kits and their supported operating systems
- DBDOC build system hardware performance requirements

- Working with the DBDOC security dongle
- What happens when I install over a previous version of DBDOC?

8.9.1 DBDOC build system hardware performance requirements

There are no hard and fast system performance requirements for running the DBDOC Build System, but more memory, faster storage, and more single-threaded CPU speed will speed up your builds.

RAM

- More is better, 6 GB recommended for large systems.
- Doubling RAM can reduce build time by as much as a factor of five, due to the significant slowdown that can result if you run out of RAM during a build.

CPU Speed

- Increasing the CPU clockspeed has a linear effect.
- The build system is mostly single-threaded, so additional CPU cores will unfortunately not help very much. They will only speed up builds if the build process was otherwise being interrupted by other processes on the system (especially Windows itself), which is to say, you should still have at least 2 CPU cores, optimally 4. This is inevitable with any CPUs from the past decade, but worth noting when allocating resources for a virtual machine.

Storage Speed

- Compared to HDD storage, SSD storage can speed up DBDOC builds greatly, at a level varying from noticeable (say, 25%) to dramatic (400% in some cases) depending on the type of data being built for your system. This primarily applies to the buildroot; while faster speeds on the source of data being built into the project will have some effect, it's in random write performance that builds can most get bottlenecked.

8.9.2 Working with the DBDOC security dongle

You will need a DBDOC USB Dongle in order to build your system (or, if this is not possible, a software-only license can be provided at additional cost).

- Windows automatically installs the necessary drivers.
- BuildPlus will tell you whether or not the dongle is present on the port.

Note: The dongle is only required for builds. If your build machine does not have a USB port available to it (for example if it is a virtual machine whose setup precludes this), see our documentation on USB over networks, or contact GMCL for a software-based license.

Also see Installing the DBDOC build system and Licensing DBDOC.

8.9.3 What happens when I install over a previous version of DBDOC?

The installation process will check to see if a previous version of DBDOC has been installed on the machine. If there is one, it will:

- Preserve all settings (CIUMon, services, Hyperview, BuildPlus projects)
- Create an archive folder of all the program files in a sibling directory named DBDOC_archive.
- Uninstall the previous version of DBDOC.
- Delete any remnants that may be left after the uninstall.
- Create a log file DBDOC_install.log, located in the Windows directory.

See **Updating your DBDOC installation** for complete instructions as to how to update your DBDOC installation.

8.10 Updating your DBDOC installation

These instructions will guide you in installing the latest version of DBDOC over a previous version.

We recommend that you update the different components of DBDOC in the following order. This will ensure that no data is lost and will minimize live data downtime during the update.

To update DBDOC:

8.10.1 Step 1: Rebuild your project before updating

It is important to rebuild your project immediately before updating. This lets you keep the management of change descriptions generated by DBDOC accurate across the update.

This new project file will also be useful as a backup if you have any problems with the update.

8.10.1.1 Reasons for rebuilding first

- Confirms that the current build system works.
- Provides a valid Management of Change Report.
- Ensures you have a working current DBDOC project file.
- Provides a diagnostic set of data in case of a problem.

8.10.1.2 How do you rebuild?

1. Open BuildPlus. On the toolbar menu, click the **Build** button.
2. If there are any errors, stop and contact GMCL.

8.10.1.3 What to do after a successful build

1. In BuildPlus, on the **Tools** menu, click **Launch Windows Explorer**. A Windows Explorer window will open.
2. In the Windows Explorer window, open the `ERROR_FILES` folder.
3. Rename the file `ERROR.7z` to `ERROR-old.7z`. This will differentiate your old error files from the new ones.

8.10.2 Step 2: Update CIUMon

Note: If CIUMon is running on a computer that should not be rebooted, please contact GMCL.

Updating CIUMon first ensures that live data is available throughout the rest of the update process. Note that you will need administrator privileges to update CIUMon.

8.10.2.1 Stop CIUMon

If CIUMon is running on the desktop, exit it now. If it is running as a service it will be stopped and restarted automatically when you install the new version.

8.10.2.2 Install the new version of CIUMon in the same location

The new version of CIUMon will be installed in the same location as the old version. The update method now depends on where CIUMon is installed.

Note: If you find the CIUMon is installed in **BOTH** of the locations below on a single computer, please

contact GMCL before continuing with the update. Two CIUMon installations on a single computer may cause problems.

Note: If you are not sure how CIUMon was installed, you may try either of the following update methods. If you try the wrong method, you will get an error message indicating that you need to use the other method instead.

8.10.2.2.1 A: If CIUMon is installed in C:\Program Files\GMCL\DBDOC\Programs

In this case, CIUMon was installed with the Build System, and you should now update the Build System and CIUMon together. Follow the instructions in **Step 3: Updating the Build System**, then come back and verify that the new version of CIUMon works.

8.10.2.2.2 B: If CIUMon is installed in C:\DBDOC\CIUMon\Programs

In this case, CIUMon was installed with the CIUMon kit, and you need to update CIUMon by itself.

1. Insert the DBDOC DVD. In Windows Explorer, open My Computer. Right-click the DVD, then click **Explore**. Double-click **DBDOC Kits**.
2. Execute the correct CIUMon kit for your system:
 1. For Windows NT 4, run `CIUMon_NT4_Kit.exe`.
 2. For Windows 2000, run `CIUMon_W2K_Kit.exe`.
 3. For newer versions of Windows, run `CIUMon_Kit.exe`.
3. The installer will open. Make sure to install in the same folder as your existing CIUMon installation.

8.10.2.3 Verify that the new version works

Run CIUMon manually to make sure that it will run and connect to the CIU correctly. Once it is connected, the messages in the CIUMon window will end with a green "OK".

If you run multiple CIUMon processes, verifying that each of them works is critical. Note that only the first one you run will know which configuration to use. You will have to select the correct configuration for the other CIUMon processes when you test them.

If you usually run your CIUMon processes as services, quit CIUMon once you have verified that it works so that you can restart the services.

8.10.2.4 Restart CIUMon services

In the Windows Services Manager, find each CIUMon service and start it.

Open Task Manager (press **Ctrl+Shift+Esc**). Check that the correct number of CIUMon processes are running in the "processes" display.

Now make sure that Hyperview users can access live data.

8.10.3 Step 3: Update the Build System

Update BuildPlus on the computer that you use to do builds. Note that you will need administrator privileges for this update.

8.10.3.1 Install the new version

1. Insert the DBDOC DVD in the build computer.
2. On the DVD splash screen, click **Next**.
3. When prompted, enter the password from the back of the DBDOC DVD case.

You can also download the installers from our website.

We recommend updating Hyperview before you rebuild your project. This will ensure that the version of Hyperview you are using will be able to take advantage of any changes or improvements we have made.

8.10.4 Step 4: Updating Hyperview

Updating Hyperview next ensures that users will be able to open the new project file when you build it. There are two different procedures for updating Hyperview, depending on whether it is possible to schedule a time when no users are currently using it. It is slightly more involved to update Hyperview while it is in use, so use Option 1 if you can be certain nobody currently has Hyperview open.

8.10.4.1 Option 1: Updating Hyperview when nobody is using it

Use this method if you are the only Hyperview user, or you can be sure that no other Hyperview users have it running.

8.10.4.1.1 Find or schedule a time for an update

Unless you use **Option 2** (below), the old version of Hyperview cannot be replaced while people are using it. All users will need to stop using Hyperview briefly during the update. Select a time when few users will be using Hyperview, and notify Hyperview users about the coming update. If you have multiple Hyperview installations (for instance, on your secure LAN and on your business LAN), schedule an update time for each installation.

8.10.4.1.2 Check the current install location

The new version of Hyperview must be installed as the same location as the old version, so that Hyperview users' shortcuts will continue to work. Make sure to check the locations for all your Hyperview installations.

For each shared Hyperview installation:

1. On a computer that uses that installation, find the shortcut that is used to access Hyperview.
2. Right-click the shortcut and then click **Properties**.
3. Click the **Shortcut** tab.

4. Look at the "Start In" data. It will tell you the location of the Hyperview installation.

8.10.4.1.3 Install the new version in the same location

The new version of Hyperview must be copied to the same location as the old version, so that Hyperview users' shortcuts will continue to work.

For each shared Hyperview installation:

1. On any computer that can access the shared folder where Hyperview is installed, insert the DBDOC DVD. Exit the DBDOC Installation DVD window. In Windows Explorer, open My Computer. Right-click the DVD, then click **Explore**. Double-click **DBDOC Kits**.
2. Double-click Hyperview_Kit.exe and follow the instructions on screen to install the kit.
3. In the shared folder where the old version of Hyperview was installed, rename the `Programs` folder to `Programs_old` and `Documents` to `Documents_old`.
4. Copy the contents of `C:\DBDOC\Hyperview` to the shared folder.

8.10.4.1.4 Make sure that the new version works

For each shared Hyperview installation:

On a computer that uses that installation, double-click the Hyperview shortcut. Hyperview should open your project.

1. In Hyperview, browse to a CAD or CLD sheet, then press **M**. A single live data value should appear for each hotspot on the sheet.
 Note: These values will turn red after a short time, to let you know they are not being continuously updated.
2. On the **Help** menu, click **Contents**. The DBDOC help window should open.

If there are any problems with this step, please contact GMCL.

8.10.4.2 Option 2: Updating Hyperview when users may be using it

Use this method if it is difficult to ensure that there are no open copies of Hyperview running anywhere from the shared installation location. It is a little bit more involved than **Option 1**.

8.10.4.2.1 Check the current install location

The new version of Hyperview must be installed as the same location as the old version, so that Hyperview users' shortcuts will continue to work. Make sure to check the locations for all your Hyperview installations.

For each shared Hyperview installation:

1. On a computer that uses that installation, find the shortcut that is used to access Hyperview.
2. Right-click the shortcut and then click **Properties**.

3. Click the **Shortcut** tab.
4. Look at the "Start In" data. It will tell you the location of the Hyperview installation.

8.10.4.2.2 Install the new version in the same location

The new version of Hyperview must be copied to the same location as the old version, so that Hyperview users' shortcuts will continue to work.

For each shared Hyperview installation:

1. On any computer that can access the shared folder where Hyperview is installed, insert the DBDOC DVD. Exit the DBDOC Installation DVD window. In Windows Explorer, open My Computer. Right-click the DVD, then click **Explore**. Double-click **DBDOC Kits**.
2. Double-click `Hyperview_Kit.exe` and follow the instructions on screen to install the kit.
3. Rename `C:\DBDOC\Hyperview\Programs` to `C:\DBDOC\HYPERVIEW\Programs_new`
4. Copy `C:\DBDOC\HYPERVIEW\Programs_new` to the enclosing folder of the shared `Programs` folder. The enclosing folder will now contain `Programs` (with the old `hyperview.exe`), and `Programs_new` (with the new `hyperview.exe`).
5. Rename the (running) `hyperview.exe` in the shared `Programs` folder to `hyperview_old.exe` (there should be no problem doing this).
6. Also rename `XpdfRasterizer.dll` in the shared `Programs` folder to `XpdfRasterizer_old.dll`.
7. Copy the files in `Programs_new` to `Programs`. Confirm replacing anything with the same name. There should be no complaints of the "file is open" variety. The new version of `hyperview.exe` will now be in the shared `Programs` folder.

When a user restarts the running Hyperview from their usual shortcut, the new version of Hyperview will be used.

8.10.4.2.3 Make sure that the new version works

For each shared Hyperview installation:

On a computer that uses that installation, double-click the Hyperview shortcut. Hyperview should open your project.

1. In Hyperview, browse to a CAD or CLD sheet, then press **M**. A single live data value should appear for each hotspot on the sheet.
 Note: These values will turn red after a short time, to let you know they are not being continuously updated.
2. On the **Help** menu, click **Contents**. The DBDOC help window should open.

If there are any problems with this step, please contact GMCL.

8.10.5 Step 5: Rebuild your project

Rebuild your project, and put the new project file file in the shared folder.

8.10.5.1 Rebuild in BuildPlus

Open BuildPlus. On the **Build** menu, click **Build**.

Note: After updating from DBDOC version 10.3.1 or earlier, BuildPlus will convert your projects to our new node format. This may take several minutes. This conversion will occur only once.

If there are any errors, stop and contact GMCL.

8.10.5.2 Make sure that Hyperview users can access the new project file

1. On a computer that uses Hyperview over the network, open Hyperview. Hyperview should open your project.
2. On the **Help** menu, click **Current M14 File Revision**. Check the **M14 Compile Started** date.
3. If Hyperview did not open the new project file automatically, check the shortcut target:
 1. Right-click the shortcut and then click **Properties**.
 2. Click the **Shortcut** tab.
 3. Check that the **Target** is something like "`[Hyperview install location]\hyperview.exe`" "`[shared project files location]*.dbdoc`".
The wildcard (*) tells Hyperview to open the newest project file.

8.10.5.3 Make sure that Hyperview users can view live data

1. On a computer that uses Hyperview over the network, open Hyperview. Hyperview should open your project.
2. In Hyperview, browse to a CAD or CLD sheet, then press **M**. A single live data value should appear for each hotspot on the sheet.

Note: These values will turn red after a short time, to let you know they are not being continuously updated.

8.11 Licensing DBDOC

This section includes information on the number of documents that can be built, and what to do when licensing expires. BuildPlus requires licensing. It is fully licensed when your DBDOC USB dongle is installed

on the machine running BuildPlus, or a software-based license is provided (available for DBDOC 11.2 or higher).



To update your USB dongle, or to request and apply a software-based license:

1. Use *BuildPlus Help DBDOC License* to see your current license allocation and generate a .C2V file
 2. Contact GMCL with your new license requirements (different document count, different expiry date, etc.) and the .C2V file
 3. GMCL will send a .V2C file
 4. Select the .V2C file in the license dialog and *Apply V2C*
 5. The license on the dongle, or via software-only, should now be updated
- There is an unlimited duration license on purchase.
 - Users will typically need one license for each computer running BuildPlus.
 - Running Hyperview to view your project file does not require a license.
 - Running CIULink to collect data from your system via CIUMon does not require a license, nor does CIUMon itself.

- If you are requesting a software license, ensure you removed any DBDOC hardware license dongles **before** running BuildPlus and generating the .C2V file.
- To migrate a software license from one system to another, contact GMCL.

8.11.1 How many documents can I build?

The practical limit is in the range of 30,000 documents, but depends on many factors.

To see how many documents you are licensed to build:

1. Start BuildPlus.
2. In the **Help** menu, click **DBDOC License**.
3. The number of documents you are licensed to build is stated in the first line.

The following document types are counted toward your build:

- CAD files / Sheets in CLD files
- Process Portal B / Operate IT / Conductor NT models (but not submodels)
- SODG Graphics (but not symbols)
- Ladder (each counts for 50 documents)
- AutoCAD, MicroStation and embedded PDF pages (each counts for 1/3 of a document, rounded up)
- Batch files
- 800xA AFW graphics
- S+ and PGP graphics (but not submodels/symbols)
- 800xA PG2 graphics extracted via the Assyst tools
- AC 800M table topics

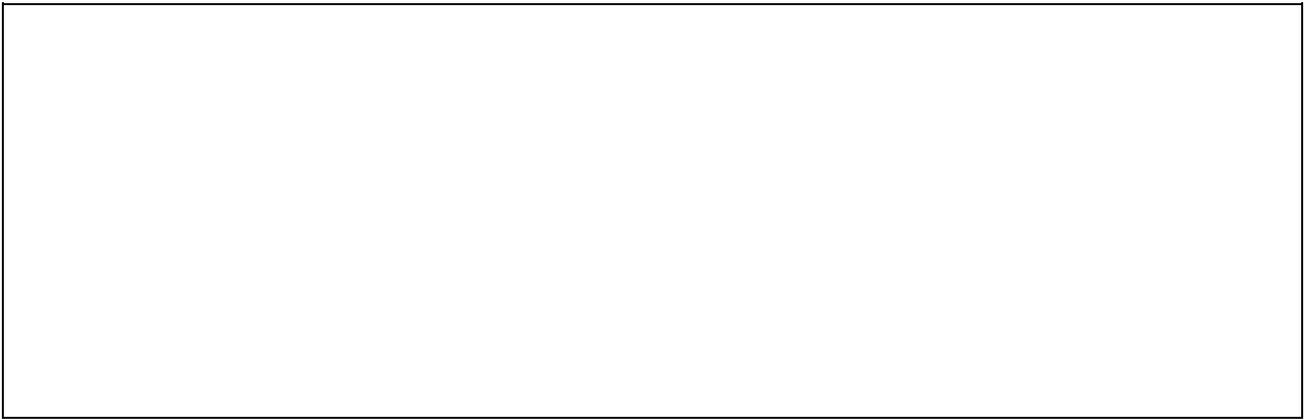
The following document types are *not* counted toward your build:

- Linked PDF & TIFF files
- Text files
- AC 800M diagram pages

The build process counts the number of documents. The number of graphics included in the document count is not allowed to exceed the number of configuration documents. The hyperlink compilation is terminated if the licensed document count is exceeded. It is your option to either reduce your build set or opt for a larger license. For time limited licenses, a warning is displayed if there are 30 days or less remaining in the license.

8.11.2 When your license expires

When your license expires, the following dialog boxes will appear when you start BuildPlus:



You will still have limited functionality of BuildPlus.

In BuildPlus you will still be able to access the following:

- View .dbdoc/.M14 Document
- Rename .dbdoc/.M14 Document
- Clean intermediate files
- Create Zip File of Error/Debug Files
- Launch Windows Explorer

CIUMon will be fully functional.