Operating Procedures

Section 2

Introduction

This section describes how to start, stop and operate the ABB Automation InfoPlus.21 interface. The Windows NT user account needs no special requirements to run the InfoPlus.21 interface server software. However, sufficient account privileges are required to set up TCP/IP host services for the server.

Interface Startup

The ***cimio_bailey_start.bat*** file, located in the ***%CIMIOROOT%\io\Abb*** directory, will start the DLGP (device logical gateway process) and DIOP (device I/O process). Any command line parameters given to the startup of the DLGP or DIOP should be entered into this file.

Command line parameters for the DLGP are shown in Table 2-1. The command line parameters for the DIOP are shown in Table 2-2.

Table 2-1. DLGP Command Line Parameters

Parameter	Description	Default
[**-11** *logFileName1*]	Name of DLGP log file 1.	bailey_dlgp1.log
[**-12** *logFileName2*]	Name of DLGP log file 2.	bailey_dlgp2.log
[**-d** *dbgLevel(0-6)*]	Debug level (0=OFF to 6=DEBUG).	2
[**-diopn** *diopNodeName*]	DIOP node name.	Current node
[**-diops** *diopServiceName*]	DIOP service name.	CIMIOBAILEYDIOP
[**-dlgps** *dlgpServiceName*]	DLGP service name.	CIMIOBAILEYDLGP
[**-dt** *shutdownTimeout*]	DIOP shutdown message time-out (1/10th seconds).	300
[**-ls** *logFileSize*]	Maximum size of each log file.	100,000
[**-rt** *receiveTimeout*]	Receive message time-out (1/10th seconds).	50
[**-st** *sendTimeout*]	Send message time-out (1/10th seconds).	100

Table 2-2. DIOP Command Line Parameters

Parameter	Description	Default
[**-ct** *clientTimeout*]	Client time-out (minutes).	0 (infinite)
[**-d** *dbgLevel(0-6)*]	Debugging logging level (0=OFF to 6=DEBUG).	2
[**-diops** *diopServiceName*]	DIOP service name.	CIMIOBAILEYDIOP
[**-dlgpn** *dlgpNodeName*]	DLGP node name.	current node
[**-dlgps** *dlgpServiceName*]	DLGP service name.	CIMIOBAILEYDLGP
[**-dqt** *dataquietTimeout*]	(Data quiet time-out) Forces any point that has not exception reported in the specified number of minutes to report the value to the client with the current server system time. Requires the IO_FREQUENCY field in IoUnsol transfer records to be greater than zero (e.g., +0000:00:20.0). Must be used in conjunction with the -ebpe DIOP command line parameter.	None
[**-dtime** *time-syncWait*]	Wait for control system time-sync for a specified time (sec.) greater than zero.	Wait forever
[**-ebpe**][1,2]	Performance enhancement improves handling of unsolicited XR processing. Because of the higher demands on the CPU in this mode, the parameter **/high** should be used instead of **/realtime** in a start command. Refer to the example following this table.	Standard operation (The system installs with this enhancement turned on.)
[**-ici** *logicalICI*]	ABB Automation logical ICI to connect to.	1
[**-l1** *logFileName1*]	Name of DIOP log file 1.	bailey_diop1.log
[**-l2** *logFileName2*]	Name of DIOP log file 2.	bailey_diop2.log
[**-ls** *logFileSize*]	Maximum size of each log file.	1,000,000
[**-rd** *iciReconnectDelay*]	Delay in minutes between reconnect attempts.	5
[**-rt** *receiveTimeout*]	Receive message time-out (1/10th seconds).	5
[**-st** *sendTimeout*]	Send message time-out (1/10th seconds).	10
[**-stime** *synchronize time*	Synchronize CIM-IO server to DCS time	none
[**-t** *dcsTimeOffset*]	Time offset (in minutes) of control system from server computer.	0

Table 2-2. DIOP Command Line Parameters *(continued)*

Parameter	Description	Default
[**-ut**][1]	Turns on unsolicited time-out checks if present.	No time-out checking
[**-xr** *maxExceptionReadCount*]	Maximum times exceptions will be read from the computer interface when processing exceptions.	15

NOTES:

1. These two parameters are mutually exclusive. Only one of the two can be used.
2. Cannot be used with GET transfers.

A startup batch file on Windows NT could look like the following example:

```
start /B /MIN /realtime cimio_bailey_diop.exe -dqt 240 -d 2
-ebpe

start /B /MIN /high cimio_bailey_dlgp.exe
```

NOTE: Because of the higher demands on the CPU in the performance enhancement mode, the parameter **/high** should be used instead of **/realtime** in a start command, as shown in the example.

Multiple computer interfaces can be accommodated by running multiple instances of the server under different TCP/IP service names. Service names will need to be added to the services file, as well as ***cimio_logical_devices.def*** for each new DIOP and DLGP. The startup of these servers might look as follows:

```
start /B /MIN /realtime cimio_bailey_diop.exe -ici 1 -diops
CIOBLYDIOP1 -dlgps CIOBLYDLGP1

start /B /MIN /high cimio_bailey_dlgp.exe -dlgps
CIOBLYDLGP1 -diops CIOBLYDIOP1

start /B /MIN /realtime cimio_bailey_diop.exe -ici 2 -diops
CIOBLYDIOP2 -dlgps CIOBLYDLGP2

start /B /MIN /high cimio_bailey_dlgp.exe -dlgps
CIOBLYDLGP2 -diops CIOBLYDIOP2
```

Interface Shutdown

The interface can be shut down by sending a shutdown notification to the InfoPlus.21 interface server. This can be sent with the client program to the DLGP to terminate and clean up the server. A shutdown notification can also be sent with a provided program (***cimio_bailey_shutdown.exe***), where the logical device name (e.g., iobailey) is passed as a command line parameter. A shutdown batch file (***cimio_bailey_shutdown***) is provided on the server in the ***%CIMIOROOT%\io\Abb*** directory to manually stop the DIOP and the DLGP, as well as to remove the semAPI device driver.

Interface GUI

A Web-based interface GUI is included for remote manual control of the interface startup and shutdown operations, including command line parameter modification. Access it via a Web browser using the URL:

http:// *ip21serverName* **/ip21_inter_release/ip21_control_gui.htm**

The GUI can optionally be installed on the interface platform.

NOTE: Windows NT option pack four must be installed prior to GUI installation.

Client Operation

The CIM-IO client task has to be running in order to exchange data between the database and the device. The client task for the interface is TSK_MBLY. This task can be started automatically when the database is started by having it referenced in the ***setcim.run*** file. This is discussed in ***CIM-IO Client Installation and Configuration*** in the ABB Automation ***InfoPlus.21 Interface Introduction and Installation*** instruction.

The alternative method is to use the SETCIM utility RUNTSK to start the TSK_MBLY client task. The utility will prompt the user for the program name (CIMIO_C_CLIENT) and the name of the task (TSK_MBLY).

Example: *Enter program name:* **cimio_c_client**

Enter task name <TSK_CIMI>: **TSK_MBLY**

Enter output file name <setcim:[group500]cimio_c_client.out>: **TSK_MBLY.OUT**

Enter error file name<setcim:[group500]cimio_c_client.err>: **TSK_MBLY.ERR**

Enter any other qualifier(s) </MAXIMUM_WORKING_SET=500>:
ENTER

Tag Name Addressing

The tag name specification will contain all addressing information for the control system value to be accessed by the InfoPlus.21 interface. The InfoPlus.21 interface server uses the point types described in Appendix A. The following two subsections explain the input and control output tag name addressing structure.

Input Tag Name Addressing

The reference will be specified using the following format:

loop.pcu.module.block.point_type[*.field_num*]

where:

loop	Loop number.
pcu	Process control unit. (This is also known as the node.)
module	Module number.
block	Block number.
point_type	semAPI point type number.
field_num	Field number. If the field number is used, a single attribute value will be returned for that point.

Three legal cases can occur when addressing tags with field numbers and requesting structures of single values:

- When no field number is given, and CIM-IO requests a CIMIO_P_STRUCT. In this case, the structure defined in Appendix B for that point type is returned.
- When no field number is given, and CIM-IO does not request a CIMIO_P_STRUCT. In this case, the default attribute value for that point type is returned. Default attribute values for each point type are specified in Appendix B.

- When a field number is given, and CIM-IO does not request a CIMIO_P_STRUCT, the value of the attribute corresponding to the field number is returned. If CIM-IO requests a CIMIO_P_STRUCT, the field number is ignored, and the structure defined for that point type is returned. Lists of attributes for each point type that can be retrieved by using the field number are given in Appendix B.

For single value reads, the optional field numbers and default fields are shown in the tables in Appendix B.

Control Output Tag Addressing and Legal Values

The reference will be specified using the following format:

loop.pcu.module.block.point_type[*.field*]

where:

loop	Loop number.
pcu	Process control unit. (This is also known as the node.)
module	Module number.
block	Block number.
point_type	semAPI point type number.
field	Optional field number. If the number is used, a single attribute value will be returned for that point.

It is important to note that a write to a Harmony block requires a specific number of parameters. Since CIM-I0 tags can contain information for a single parameter, the server may need to collect information from a number of tags in order to collect all of the parameters before the write is executed.

All of the tags containing parameter data for a write to a single block must be specified in the same PUT transfer record. The order of tags specified for a particular block is not important, but it is important that the tags be contiguous. The server scans the PUT record and executes the write when the last tag pertaining to a block is encountered (i.e., when the tag block number changes) or when the end of the record is encountered. If no tag is specified for a particular parameter, a default value is used.

Three legal cases can occur when addressing tags with field numbers and sending structures of single values:

- When no field number is given, and CIM-IO sends a CIMIO_P_STRUCT. In this case, the structure defined in Appendix C for that point provides the write parameters.
- When no field number is given, and CIM-IO does not send a CIMIO_P_STRUCT. In this case, the default attribute value for that point type is sent. Default attribute values for each point are specified in Appendix C.
- When a field number is given and CIM-IO does not send a CIMIO_P_STRUCT, the value of the attribute corresponding to the field number is sent. If CIM-IO sends a CIMIO_P_STRUCT, the field number is ignored and the structure defined for that point type is sent. Lists of attributes for each point type that can be sent by using the field number are given in Appendix C.

Refer to the CIM-IO manuals for instructions on how to implement PUT records for single value transfers.

For single value reads, the optional field numbers and default fields are shown in the tables in Appendix C.

Data Types

The InfoPlus.21 interface server is capable of returning either single values or structures to the client task.

Table 2-3 describes the CIM-IO standard data types (columns) and the semAPI data types, or device data types (rows).

Table 2-3. Data Type Conversion

semAPI	CIM-IO				
	Double	Real	Short Integer	Long Integer	ASCII
Real	X	X	X	X	X
Integer	X	X	X	X	X
ASCII	—	—	—	—	X

NOTE: An X in the cell indicates that the interface supports the conversion between the two data types.