DNP3 INTEGRATION KIT FOR PAC CONTROL USER'S GUIDE

Form 1773-140917—September 2014

OPTO 22
Automation made simple.

43044 Business Park Drive • Temecula • CA 92590-3614
Phone: 800-321-OPTO (6786) or 951-695-3000
Fax: 800-832-OPTO (6786) or 951-695-2712
www.opto22.com

Product Support Services
800-TEK-OPTO (835-6786) or 951-695-3080
Fax: 951-695-3017
Email: support@opto22.com
Web: support.opto22.com

DNP3 Integration Kit for PAC Control User's Guide
Form 1773-140917—September 2014

The information in this manual has been checked carefully and is believed to be accurate; however, Opto 22 assumes no responsibility for possible inaccuracies or omissions. Specifications are subject to change without notice.

Opto 22 warrants all of its products to be free from defects in material or workmanship for 30 months from the manufacturing date code. This warranty is limited to the original cost of the unit only and does not cover installation, labor, or any other contingent costs. Opto 22 I/O modules and solid-state relays with date codes of 1/96 or newer are guaranteed for life. This lifetime warranty excludes reed relay, SNAP serial communication modules, SNAP PID modules, and modules that contain mechanical contacts or switches. Opto 22 does not warrant any product, components, or parts not manufactured by Opto 22; for these items, the warranty from the original manufacturer applies. These products include, but are not limited to, OptoTerminal-G70, OptoTerminal-G75, and Sony Ericsson GT-48; see the product data sheet for specific warranty information. Refer to Opto 22 form number 1042 for complete warranty information.

Opto 22
Automation Made Simple.

Table of Contents

Introduction

The DNP3 Integration Kit for PAC Control (Part #PAC-INT-DNP3) allows Opto 22 SNAP PAC controllers, using PAC Control, to connect via an Ethernet network and serial port and communicate using DNP3, the Distributed Network Protocol.

The DNP3 Outstation can connect to 1 – 4 DNP3 masters using Ethernet and Serial ports.

The DNP3 Master can connect to 1 – 10 DNP3 outstations using Ethernet and Serial ports.

The Integration Kit contains:

- An example DNP3 strategy containing the DNP3 charts that are imported into a strategy to enable an Opto 22 controller to communicate as a DNP3 Level 2 Master or Outstation. See page 3.
- A set of PAC Control subroutines that are added to a strategy to add DNP Events. See page 25.
- A PAC Control subroutine that is used by the DNP3_Master chart.
- An Error Handler chart to manage I/O and log errors.

The DNP3 strategy transmits message strings as specified in the DNP3 Specification Version 2.02 15 December 2007. See http://www.dnp.org/

This manual assumes that you understand fully how to use PAC Control, DNP3, and the DNP3 device to be used.

Functions Supported

The following DNP function codes are supported by the PAC Control strategy:

Function Code	Name
0x00	Confirm
0x01	Read
0x02	Write
0x03	Select
0x04	Operate
0x05	Direct Operate
0x06	Direct Operate NR
0x07	Immed Freeze
0x08	Immed Freeze NR
0x09	Freeze Clear
0x10	Freeze Clear NR
0x13	Cold Restart
0x20	Enable Unsolicited
0x21	Disable Unsolicited
0x23	Delay Measure

Groups Supported

The following groups are supported:

Group	Name
1	Binary Input
2	Binary Input Event
10	Binary Output
12	CROB
20	Counter
21	Frozen Counter
22	Counter Event
30	Analog Input
32	Analog Input Event
40	Analog Output Status
41	Analog Output
50	Time And Date
51	Time And Date CTO
52	Time Delay
60	Class Object
80	Internal Indicators

What is Required

You will need:

- An Opto 22 SNAP PAC controller with firmware R8.1a or later
- A PC running PAC Control Professional 8.1a or later software and the DNP3 Integration Kit for PAC Control
- (Optional) A SNAP PAC Learning Center (to run the example strategy)

Installing the Integration Kit

To install the integration kit on your computer, extract the zip file to your drive, usually C:\PACDNP3Level2.

Using the Example Strategy

Running the Example Strategy

The example strategy uses the SNAP-PAC-S1 Controller and the SNAP PAC Learning Center's I/O to demonstrate how to use the DNP3 Master and DNP3 Outstation protocol.

To run the example strategy:

1. Set up the Learning Center so that you can access it from your PC. For more information, see form 1638, *SNAP PAC Learning Center User's Guide*.
2. Start PAC Control Pro.
3. Navigate to the integration kit's folder, PACDNP3Level2.
4. Open the strategy file, PAC DNP3 Level2.idb.
5. Open the R1 I/O unit and change the IP address to your Learning Center IP address.
6. Add your control engine and set it as the Active Engine.
7. In the Powerup chart, comment out the DNP3_Master or DNP3_Outstation_Protocol chart if only one is needed.
8. Download and start the example strategy.

Using an R-series Controller

If you want to use a SNAP-PAC-R1 or SNAP-PAC-R2 to run the DNP3 charts, it may not have the same performance as a PAC S-series controller. This depends on how busy the controller is with other charts and tasks you have configured.

User data is entered in block 167 of the chart named DNP3_Outstation_Protocol. Line 8 (poPAC_DNP_Controller_SNAP_PAC_S1_or_S2 = &SNAP_PAC_S1;) loads the controller running the DNP3 charts into a pointer variable. This pointer variable (poPAC_DNP_Controller_SNAP_PAC_S1_or_S2) will accept either a SNAP-PAC-S1 or SNAP-PAC-S2 control engine.

In order to use a SNAP-PAC-R1 or SNAP-PAC-R2 to run the DNP3 Integration Kit, do these additional steps:

1. Make one of the two following changes in line 8 of block 167, depending on the type of control engine you are using:

   Use poPAC_DNP_Controller_SNAP_PAC_R1 for the SNAP-PAC-R1 in place of poPAC_DNP_Controller_SNAP_PAC_S1_or_S2

   or

   Use poPAC_DNP_Controller_SNAP_PAC_R2 for the SNAP-PAC-R2 in place of poPAC_DNP_Controller_SNAP_PAC_S1_or_S2

   Use only one of the pointers.

2. Replace "SNAP_PAC_S1" on line 8 with your control engine name.

   Make sure to keep the preceding "&" symbol. For example, if your control engine name is PANEL5, &SNAP_PAC_S1 would become &PANEL5.

Importing the DNP Charts

Before importing the charts, include the DMP3 subroutines.

For DNP3 Master, import the DNP3_Master and DNP3_Master_UnSol charts.

For DNP3 Outstation, import the DNP3_Outstation_Protocol, DNP3_Outstation_Auto_Events, and DNP3_Outstation_CROB charts into your strategy. First export each chart in the example strategy as a PAC Control chart export file (.cxf file), and then import it into your strategy. For more information, see Chapter 8 of form 1700, the *PAC Control User's Guide*.

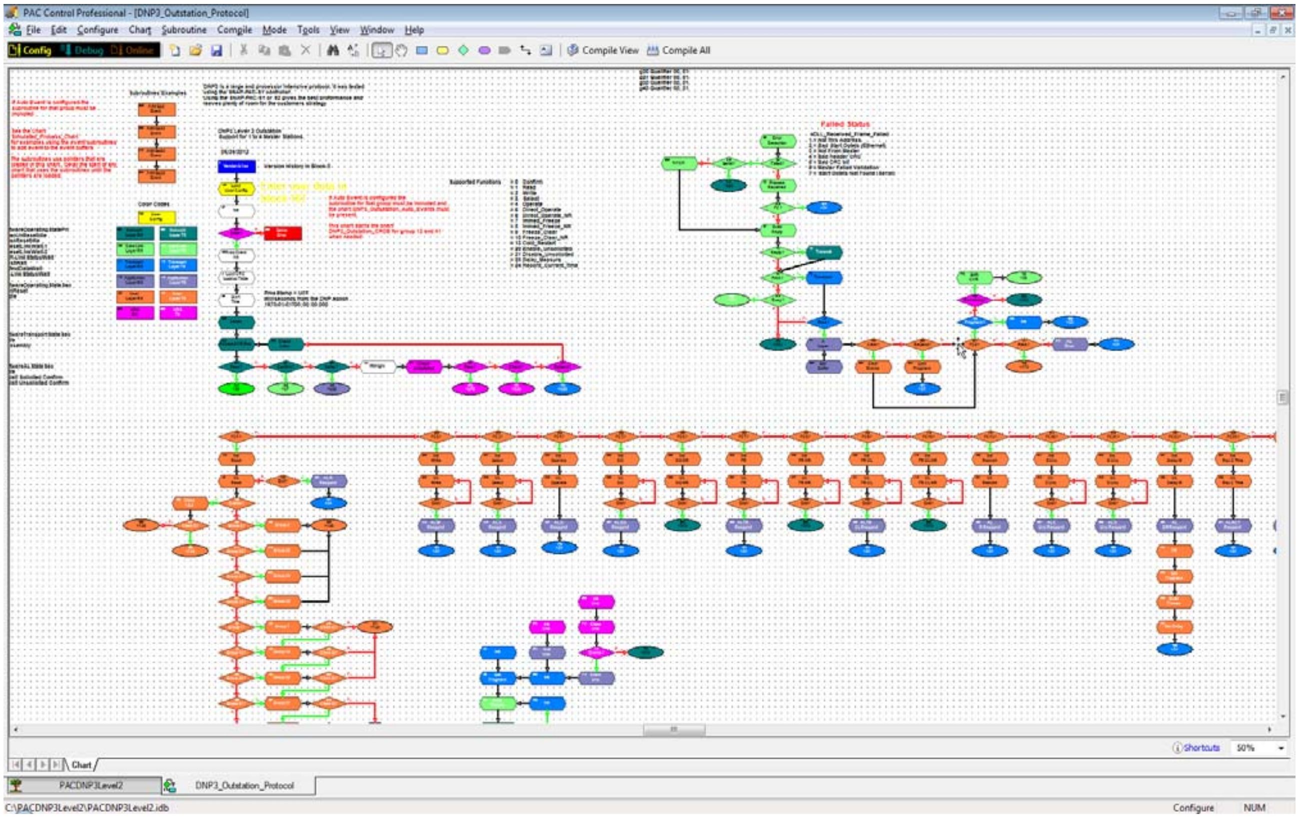
Running the DNP Charts

Start the DNP3_Outstation_Protocol chart or the DNP3_Master chart in the Powerup chart of your strategy. The other charts are started by the DNP3 charts as needed.

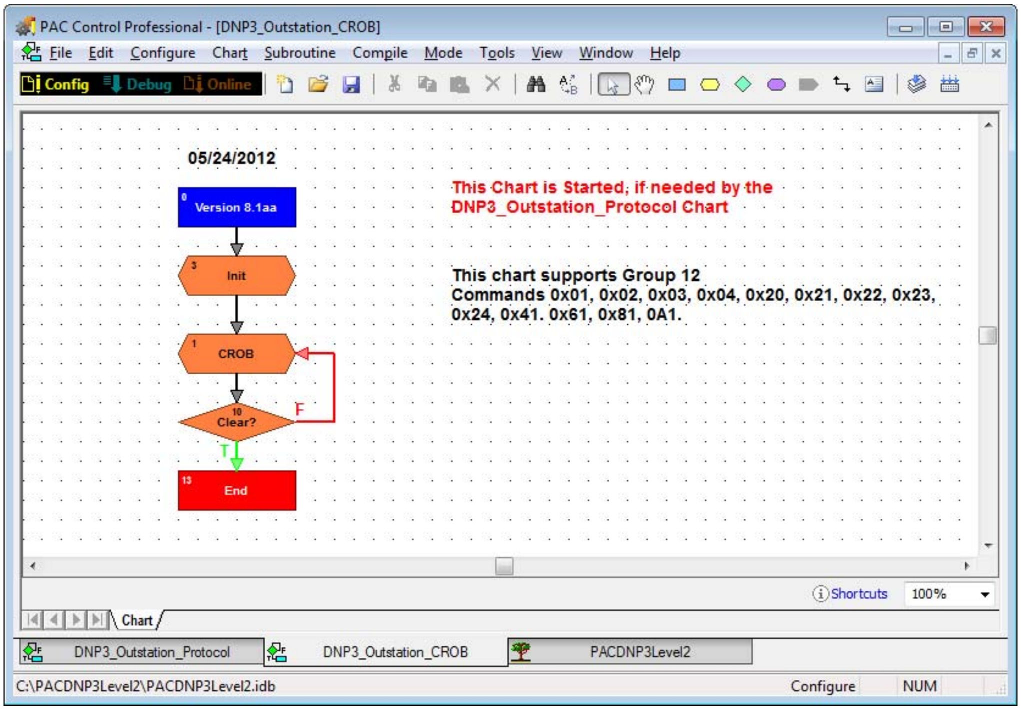
The Outstation Charts

When imported into a strategy, the DNP3_Outstation_Protocol, DNP3_Outstation_Auto_Events, and DNP3_Outstation_CROB charts enable an Opto 22 controller to communicate as a DNP3 Level 2 Outstation.

DNP3_Outstation_Protocol Chart


PAC Control Professional - [DNP3_Outstation_Protocol]
File Edit Configure Chart Subroutine Compile Mode Tools View Window Help
Config Debug Online
Compile View
Compile All
Shortcuts
50%
Chart
PACDNP3Level2
DNP3_Outstation_Protocol
C:\PACDNP3Level2\PACDNP3Level2.idb
Configure
NUM

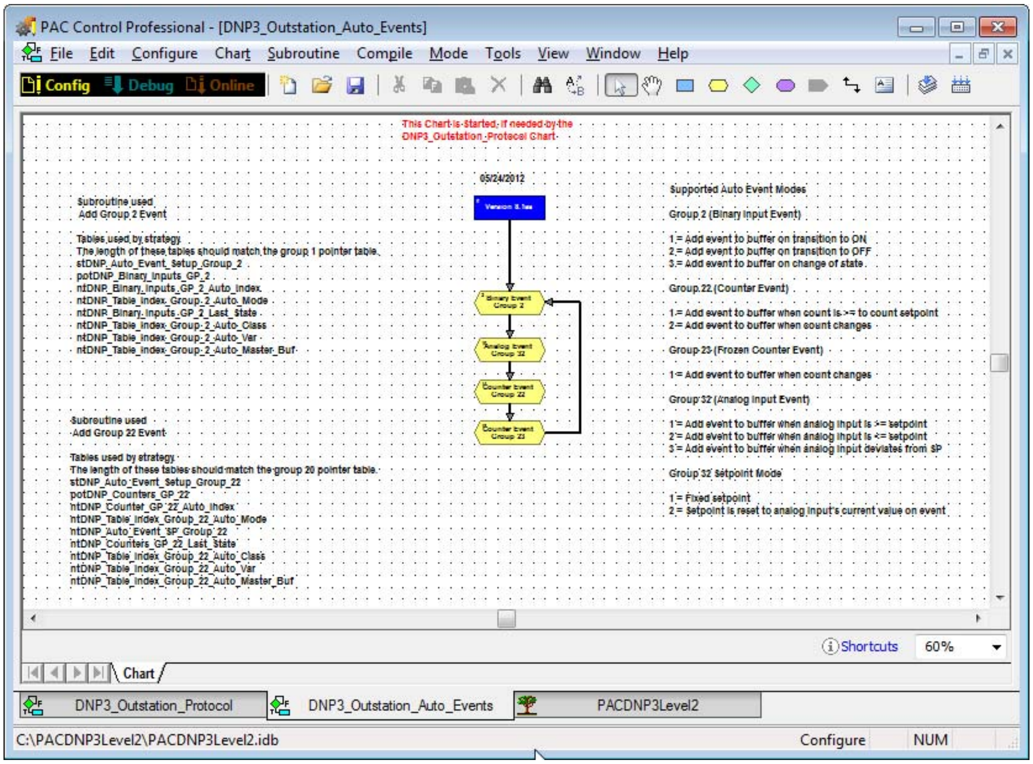
DNP3_Outstation_CROB_ Chart


PAC Control Professional - [DNP3_Outstation_CROB]
File Edit Configure Chart Subroutine Compile Mode Tools View Window Help
Config Debug Online
05/24/2012
Version 8.1aa
Init
CROB
Clear?
F
T
End
Shortcuts
100%
Chart
DNP3_Outstation_Protocol
DNP3_Outstation_CROB
PACDNP3Level2
C:\PACDNP3Level2\PACDNP3Level2.idb
Configure
NUM

This Chart is Started, if needed by the DNP3_Outstation_Protocol Chart

This chart supports Group 12
Commands 0x01, 0x02, 0x03, 0x04, 0x20, 0x21, 0x22, 0x23, 0x24, 0x41. 0x61, 0x81, 0A1.

DNP3_Outstation_Auto_Events Chart



This Chart is Started, if needed by the DNP3_Outstation_Protocol Chart

Subroutine used
Add Group 2 Event

Tables used by strategy
The length of these tables should match the group 1 pointer table.
stDNP_Auto_Event_Setup_Group_2
potDNP_Binary_Inputs_GP_2
ntDNP_Binary_Inputs_GP_2_Auto_Index
ntDNP_Table_Index_Group_2_Auto_Mode
ntDNP_Binary_Inputs_GP_2_Last_State
ntDNP_Table_Index_Group_2_Auto_Class
ntDNP_Table_Index_Group_2_Auto_Var
ntDNP_Table_Index_Group_2_Auto_Master_Buf

Subroutine used
Add Group 22 Event

Tables used by strategy
The length of these tables should match the group 20 pointer table.
stDNP_Auto_Event_Setup_Group_22
potDNP_Counters_GP_22
ntDNP_Counter_GP_22_Auto_Index
ntDNP_Table_Index_Group_22_Auto_Mode
ntDNP_Auto_Event_SP_Group_22
ntDNP_Counters_GP_22_Last_State
ntDNP_Table_Index_Group_22_Auto_Class
ntDNP_Table_Index_Group_22_Auto_Var
ntDNP_Table_Index_Group_22_Auto_Master_Buf

Supported Auto Event Modes

Group 2 (Binary Input Event)

1 = Add event to buffer on transition to ON
2 = Add event to buffer on transition to OFF
3 = Add event to buffer on change of state

Group 22 (Counter Event)

1 = Add event to buffer when count is >= to count setpoint
2 = Add event to buffer when count changes

Group 23 (Frozen Counter Event)

1 = Add event to buffer when count changes

Group 32 (Analog Input Event)

1 = Add event to buffer when analog input is >= setpoint
2 = Add event to buffer when analog input is <= setpoint
3 = Add event to buffer when analog input deviates from SP

Group 32 Setpoint Mode

1 = Fixed setpoint
2 = Setpoint is reset to analog input's current value on event

Fragment Send and Transmit Tables

The size of the fragments sent are controlled by the length of the fragment send tables (ntAL_Fragment_Send_Buffer_0 through ntAL_Fragment_Send_Buffer_5). The current configuration uses six fragment send tables with the length set to 2048. The DNP data should fit within the six tables. If more space is needed, either add more tables or increase the length of the six tables.

The following tables work together. If you change the length of one, change the others to match.

- ntAL_Fragment_Send_Buffer_0
- ntAL_Fragment_Send_Buffer_1
- ntAL_Fragment_Send_Buffer_2
- ntAL_Fragment_Send_Buffer_3
- ntAL_Fragment_Send_Buffer_4
- ntAL_Fragment_Send_Buffer_5
- ntAL_Last_Transmit_Fragment_0
- ntAL_Last_Transmit_Fragment_1
- ntAL_Last_Transmit_Fragment_2
- ntAL_Last_Transmit_Fragment_3
- ntUns_Last_Transmit_Fragment_0
- ntUns_Last_Transmit_Fragment_1
- ntUns_Last_Transmit_Fragment_2
- ntUns_Last_Transmit_Fragment_3

Fragment Received and Read Table

The size of the fragments received are controlled by the length of the fragment receive table (ntAL_Fragment_Received_Buffer_1). The current configuration has the length set to 2048.

The following tables work together. If you change the length of one, change the others to match.

- ntAL_Fragment_Received_Buffer_1
- ntAL_Hold_Last_Read_Request_0
- ntAL_Hold_Last_Read_Request_1
- ntAL_Hold_Last_Read_Request_2
- ntAL_Hold_Last_Read_Request_3
- ntAL_Last_Received_Fragment_0
- ntAL_Last_Received_Fragment_1
- ntAL_Last_Received_Fragment_2
- ntAL_Last_Received_Fragment_3
- ntUL_Last_CROB_0
- ntUL_Last_CROB_1

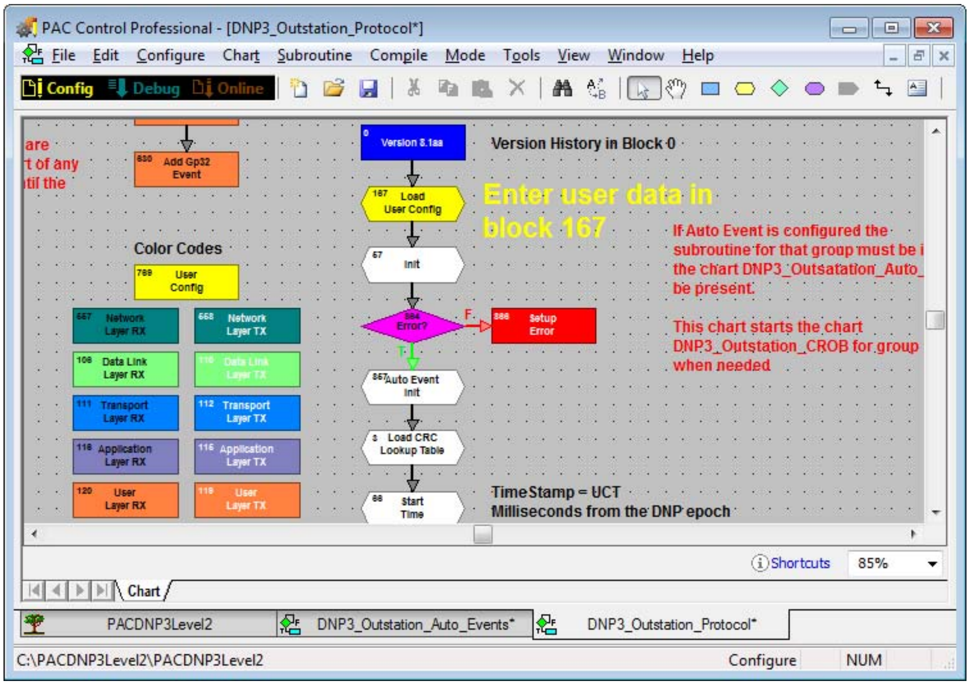
- ntUL_Last_CROB_2
- ntUL_Last_CROB_3

The size of the Master Station Event buffers are controlled by the length of the Event buffer tables (pstEvent_Buffer_0 - 3). If more or less are needed, just change the length of the tables. The tables are set to a length of 300.

The strategy will automatically adjust to the new length.

Configuring Outstation User Data

User data is entered in block 167 (Load User Data).



If Auto Event is configured the subroutine for that group must be i the chart DNP3_Outsatation_Auto_ be present.

This chart starts the chart DNP3_Outstation_CROB for group when needed

TimeStamp = UCT
Milliseconds from the DNP epoch

The configurable user data is as follows:

- Load pointer variable with controller running DNP3 Protocol
- Set DNP address of Outstation
- Enable/Disable Self Address feature
- Set IP port of each Master Station (used by Unsolicited Response if Master Station has not opened a session)
- Set Listen port for each Master Station
- Set Connection Type for each Master Station
- Set DNP address for each Master Station (used for validation)
- Set Number of Retries of responses sent of Master Station
- Enable/Disable Unsolicited Response for each Master Station
- Set Number of retries for Unsolicited responses for each Master Station

- Set Number of Events before sending to each Master Station
- Set Delay before sending Events to each Master Station
- Set Transport Segment Size
- Set CROB default times
- Set CROB wait time after Select to receive Operate function
- Set Keep-alive time
- Set Confirm timeout times
- Set Disable time after no confirm to Unsolicited Response
- Load pointer tables with I/O and virtual points
- Set default variation 0 for each group

Configuring Auto Events

User data is entered in block 167 (Load User Data). The reference indexes are to the pointer tables with the group point. The subroutine for each group used must be included

Event Single Buffer Mode

nDNP_Event_Single_Buffer_Mode

In the auto event setup you can assign each event to an event buffer for each master configured.

With nDNP_Event_Single_Buffer_Mode set false (0), if an event is placed in a master buffer only that master can read and remove it from its buffer.

For example, if you have 2 masters configured and all events are placed in both buffers (",0011"), when master 0 polls the events from buffer 0 it has no effect on the events in master buffer 1. These will need to be removed by master 1.

If master 1 is a backup for master 0 and is to be inactive until master 0 fails the event buffer for master 1 will overflow because it is not being polled.

By setting (nDNP_Event_Single_Buffer_Mode) to 1 or true all events are placed in buffer 0. The setting in each auto event group for the buffer is ignored.

Now if you have 2 masters configured, the events will be removed if either master polls for the events.

Supported Auto Event Modes

Group 2 (Binary Input Event)

1 = Add event to buffer on transition to ON

2 = Add event to buffer on transition to OFF

3 = Add event to buffer on change of state

Group 22 (Counter Event)

1 = Add event to buffer when count is >= to count setpoint

2 = Add event to buffer when count changes

Group 23 (Frozen Counter Event)

1 = Add event to buffer when count changes

Group 32 (Analog Input Event)

1 = Add event to buffer when analog input is >= setpoint

2 = Add event to buffer when analog input is <= setpoint

3 = Add event to buffer when analog input deviates from SP +- deadband

Group 32 Setpoint Mode

1 = Fixed setpoint

2 = Setpoint is reset to analog input's current value on event

Group 2 Auto Event Setup

String Table Syntax = "Index,Mode,Class,Variation,Master Buffer"

//Index of point in pointer table

//Mode 1=Event when ON 2=Event when OFF 3=Event when state changes

//Class 1, 2 or 3

//Variation 1, 2 or 3

//Master Buffer 0001 = Master buffer for master at index 0, 1000 = Master buffer for master at index 3, 0011 = Master buffer for master at index 0 and 1

Inputs can be selected in any order in table stDNP_Auto_Event_Setup_Group_2

No data in string table will disable auto events for binary Inputs

Examples:

- stDNP_Auto_Event_Setup_Group_2[0] = "2,3,1,2,0001";
- stDNP_Auto_Event_Setup_Group_2[1] = "1,1,2,3,0001";
- stDNP_Auto_Event_Setup_Group_2[2] = "4,2,1,1,0001";
- stDNP_Auto_Event_Setup_Group_2[3] = "6,3,1,2,0001";
- stDNP_Auto_Event_Setup_Group_2[4] = "0,3,1,2,0001";
- stDNP_Auto_Event_Setup_Group_2[5] = "3,1,1,2,0001";

Subroutine Used

Add Group 2 Event

Tables Used by Strategy

The length of these tables should match the group 1 pointer table.

- stDNP_Auto_Event_Setup_Group_2
- potDNP_Binary_Inputs_GP_2
- ntDNP_Binary_Inputs_GP_2_Auto_Index
- ntDNP_Table_Index_Group_2_Auto_Mode
- ntDNP_Binary_Inputs_GP_2_Last_State
- ntDNP_Table_Index_Group_2_Auto_Class
- ntDNP_Table_Index_Group_2_Auto_Var
- ntDNP_Table_Index_Group_2_Auto_Master_Buf

Group 22 Auto Event Setup

String Table Syntax = "Index,Mode,Class,Variation,Master Buffer"

//Index of point in pointer table

//Mode 1=Count >= sp 2=Event when Count changes

//Class 1, 2 or 3

//Variation 1, 2, 5 or 6

//Master Buffer 0001 = Master buffer for master at index 0, 1000 = Master buffer for master at index 3, 0011 = Master buffer for master at index 0 and 1

Inputs can be selected in any order in table stDNP_Auto_Event_Setup_Group_22

No data in string table will disable auto events for counters

Examples:

- stDNP_Auto_Event_Setup_Group_22[0] = "2,1,1,2,0001";
- stDNP_Auto_Event_Setup_Group_22[1] = "0,2,2,5,0001";
- stDNP_Auto_Event_Setup_Group_22[2] = "4,2,1,1,0001";

//Setpoint for Auto Events if Mode is (1). No Used for Mode 2

ntDNP_Auto_Event_SP_Group_22[0] = 10;

ntDNP_Auto_Event_SP_Group_22[1] = 0;

ntDNP_Auto_Event_SP_Group_22[2] = 0;

Subroutine Used

Add Group 22 Event

Tables Used by Strategy.

The length of these tables should match the group 20 pointer table.

- stDNP_Auto_Event_Setup_Group_22

- potDNP_Counters_GP_22
- ntDNP_Counter_GP_22_Auto_Index
- ntDNP_Table_Index_Group_22_Auto_Mode
- ntDNP_Auto_Event_SP_Group_22
- ntDNP_Counters_GP_22_Last_State
- ntDNP_Table_Index_Group_22_Auto_Class
- ntDNP_Table_Index_Group_22_Auto_Var
- ntDNP_Table_Index_Group_22_Auto_Master_Buf

Group 23 Auto Event Setup

String Table Syntax "Index,Mode,Class,Variation,Master Buffer"

//Index of point in pointer table

//Mode 1=Event when Count changes

//Class 1, 2 or 3

//Variation 1, 2, 5 or 6

//Master Buffer 0001 = Master buffer for master at index 0, 1000 = Master buffer for master at index 3, 0011 = Master buffer for master at index 0 and 1

Inputs can be selected in any order in table stDNP_Auto_Event_Setup_Group_23

No data in string table will disable auto events for frozen counters

Examples:

- stDNP_Auto_Event_Setup_Group_23[0] = "1,1,1,2,0001";
- stDNP_Auto_Event_Setup_Group_23[1] = "3,1,1,2,0001";

Subroutine Used

Add Group 23 Event

Tables Used by Strategy

The length of these tables should match the group 20 pointer table.

- stDNP_Auto_Event_Setup_Group_23
- ntDNP_Binary_Inputs_GP_23_Auto_Index
- ntDNP_Table_Index_Group_23_Auto_Mode
- ntDNP_Auto_Event_SP_Group_23
- ntDNP_Table_Index_Group_23_Auto_Class
- ntDNP_Table_Index_Group_23_Auto_Var
- ntDNP_Table_Index_Group_23_Auto_Master_Buf

Group 32 Auto Event Setup

String Table Syntax "Index,Mode,SPM,Class,Variation,Master Buffer"

//Index of point in pointer table

//Mode 1=Event when >= SP 2=Event when <= SP 3=Event when Deviation from SP

//SP Mode 1=Fixed Setpoint 2=Setpoint is reset to value at last event

//Class 1, 2 or 3

//Variation 1, 2, 3, 4, 5 or 7

//Master Buffer 0001 = Master buffer for master at index 0, 1000 = Master buffer for master at index 3, 0011 = Master buffer for master at index 0 and 1

Inputs can be selected in any order in table stDNP_Auto_Event_Setup_Group_32

No data in string table will disable auto events for Analog Inputs

Examples:

- stDNP_Auto_Event_Setup_Group_32[0] = "0,3,2,1,2,0001";
- stDNP_Auto_Event_Setup_Group_32[1] = "2,2,1,2,3,0001";
- stDNP_Auto_Event_Setup_Group_32[2] = "3,1,1,1,1,0001";
- stDNP_Auto_Event_Setup_Group_32[3] = "6,3,2,1,2,0001";

Setpoint for Auto Events if Setpoint Mode is fixed (1). If Setpoint Mode is Reset (2) the value of the point at startup is the setpoint.

- ftDNP_Auto_Event_SP_Group_32[0] = 0.00;
- ftDNP_Auto_Event_SP_Group_32[1] = 120.00;
- ftDNP_Auto_Event_SP_Group_32[2] = 5.00;
- ftDNP_Auto_Event_SP_Group_32[3] = 0.00;

Deadband for Mode 3. +- from setpoint

- ftDNP_Auto_Event_Deadband_Group_32[0] = 3.0;
- ftDNP_Auto_Event_Deadband_Group_32[1] = 0.0;
- ftDNP_Auto_Event_Deadband_Group_32[2] = 0.0;
- ftDNP_Auto_Event_Deadband_Group_32[3] = 10.0;

Subroutine Used

Add Group 32 Event

Tables Used by Strategy

The length of these tables should match the group 30 pointer table.

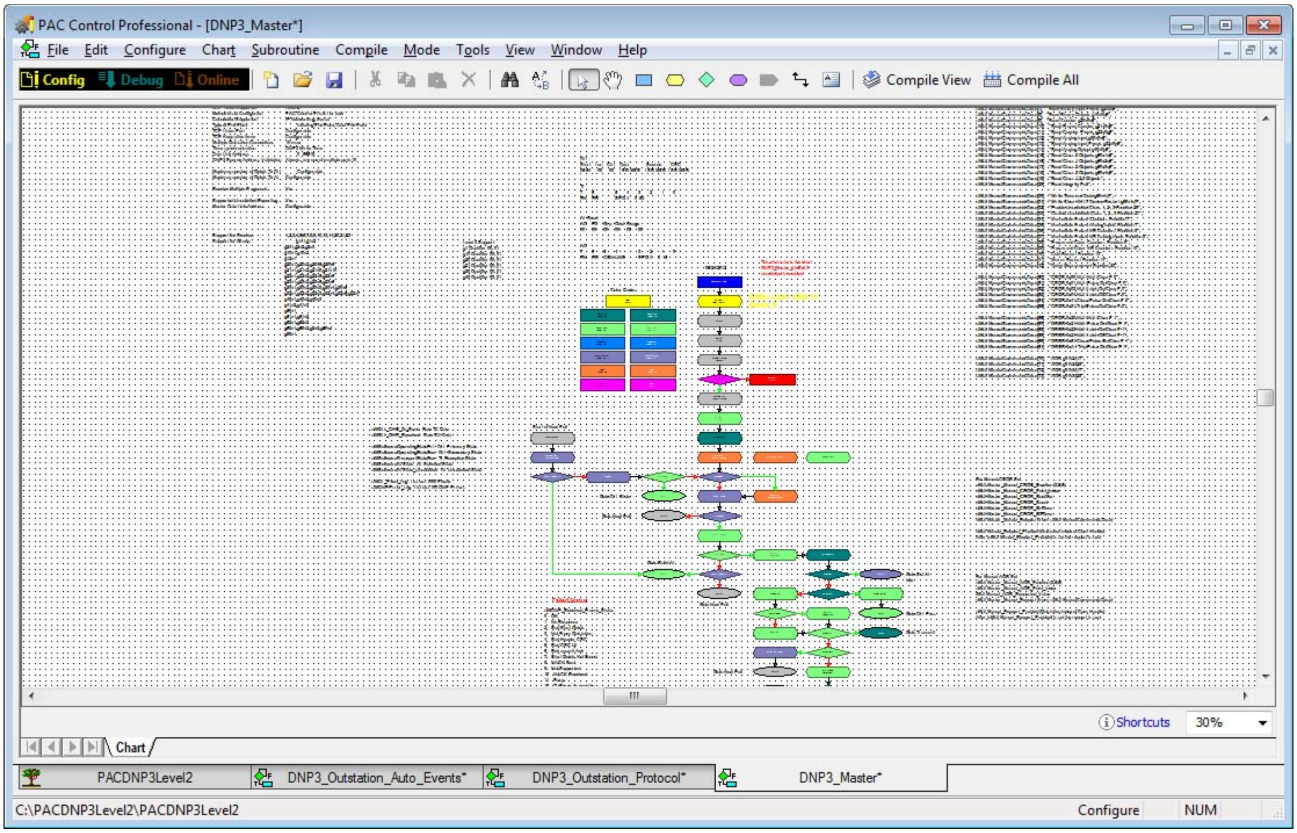
- stDNP_Auto_Event_Setup_Group_32
- potDNP_Analog_Inputs_Gp_32
- ntDNP_Counter_GP_32_Auto_Index
- ntDNP_Table_Index_Group_32_Auto_Mode
- ftDNP_Auto_Event_SP_Group_32

- ftDNP_Auto_Event_Deadband_Group_32
- ntDNP_Analog_Inputs_GP_32_Last_State
- ntDNP_Table_Index_Group_32_Auto_Class
- ntDNP_Table_Index_Group_32_Auto_Var
- ntDNP_Table_Index_Group_32_Auto_Master_Buf

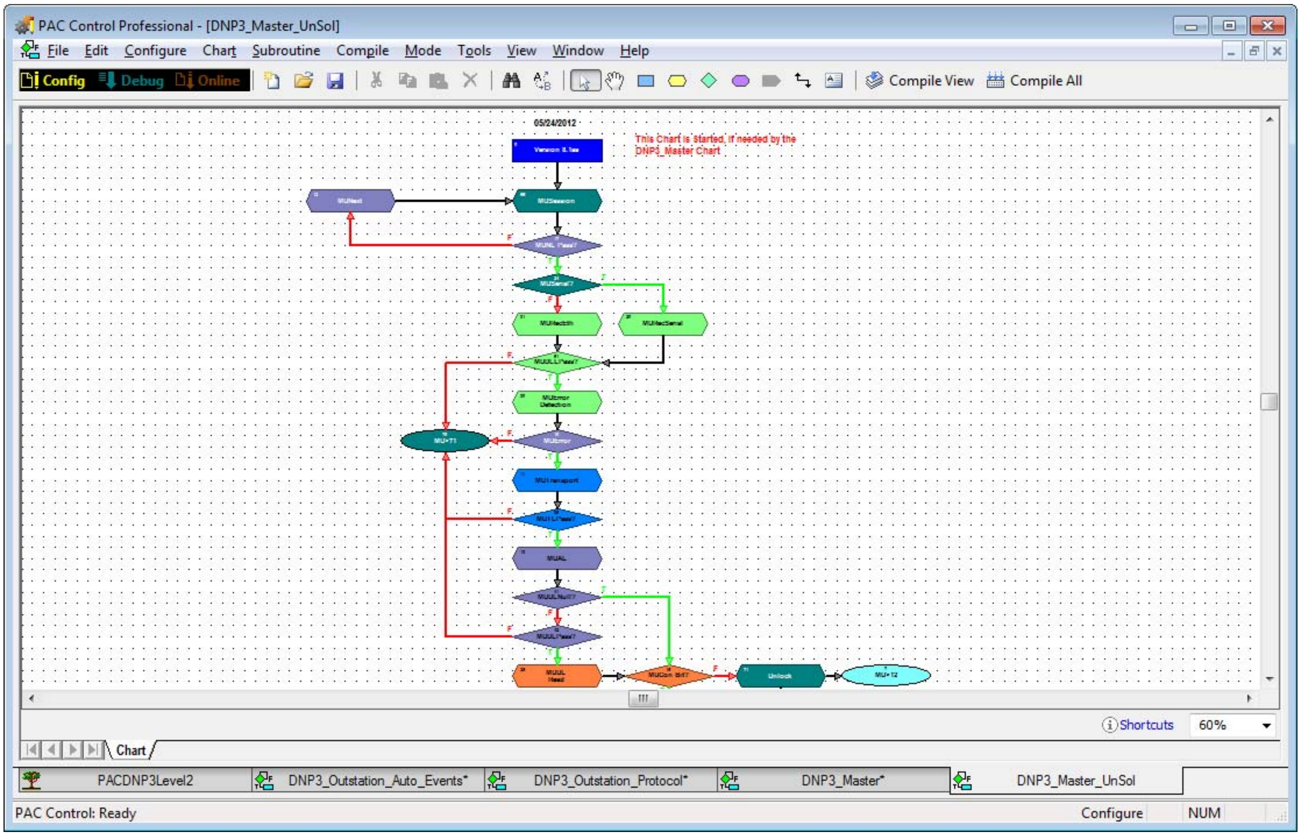
The Master Charts

When imported into a strategy, the DNP3_Master and DNP3_Master_UnSol charts enable an Opto 22 controller to communicate as a DNP3 Level 2 Master.

DNP3_Master Chart



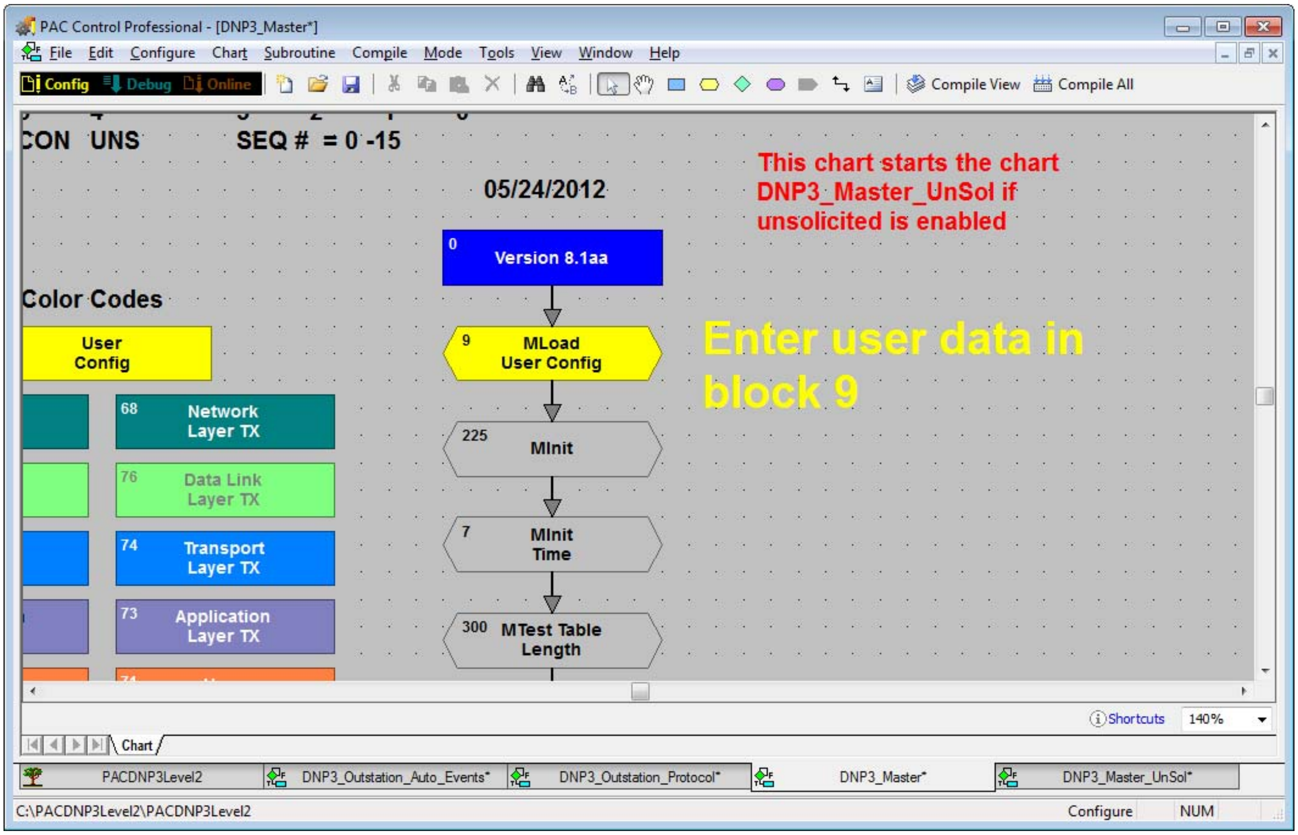
DNP3_Master_UnSol Chart



This Chart is started, if needed by the DNP3_Master Chart

Configuring Master User Data

User data is entered in block 9 (MLoad User Data).



This chart starts the chart DNP3_Master_UnSol if unsolicited is enabled

Enter user data in block 9

- Set DNP address of Master
- Set Port Mode (Serial or Ethernet)
- Set Com Handles
- Set Connection Type for each Outstation
- Set DNP address for each Outstation
- Enable communications to each Outstation.
- Enable/Disable Integrity Poll on startup
- Enable/Disable Retrieve Events on IIN1.1,IIN1.2,IIN1.3
- Enable unsolicited reports
- Set Wait time for response
- Set wait time for Serial data
- Set wait time for complete unsolicited report
- Set keep alive time. If there is no communication in time limit it will Request Link Status
- Set retry count
- Set disable unsolicited reports on IIN1.7

- Set enable unsolicited reports after IIN1.7
- Set Pointer table to store data
- Set defaults for manual CROB and AOB
- Set auto polling

DNP3 Master Auto Polling Setup

Auto polling can be setup for each outstation

After startup is completed with each outstation the data from stMUL_Poll_Type_Setup is loaded into the auto poll tables. Changes made to the table stMUL_Poll_Type_Setup while the strategy is running will not be applied until the variable nMUL_Reset_Polling is set to true.

There are several examples included.

Examples:

- stMUL_Poll_Type_Setup[0] = "0,0,15,Integrity
- stMUL_Poll_Type_Setup[1] = "0,0,5,ClassPoll,1,1,1;
- stMUL_Poll_Type_Setup[2] = "0,0,5,Custom,40,2,6";
- stMUL_Poll_Type_Setup[3] = "0,0,5,Custom,22,0,7,10";
- stMUL_Poll_Type_Setup[4] = "1,0,6,Custom,30,1,6";
- stMUL_Poll_Type_Setup[5] = "0,0,15,CROB,1,3,17,1,1,250,250";
- stMUL_Poll_Type_Setup[6] = "0,0,10,AOB,0,3,2,17,1";
- stMUL_Poll_Type_Setup[7] = "1,0,6,Custom,1,0,1,0,5";
- stMUL_Poll_Type_Setup[8] = "1,0,5,Custom,10,0,6";
- stMUL_Poll_Type_Setup[9] = "1,0,5,ClassPoll,1,1,1";
- stMUL_Poll_Type_Setup[10] = "0,0,5,Custom,30,1,6";
- stMUL_Poll_Type_Setup[11] = "1,0,5,Custom,20,0,6";
- stMUL_Poll_Type_Setup[12] = "1,0,5,Custom,21,0,6";
- stMUL_Poll_Type_Setup[13] = "1,0,5,Custom,32,0,6";
- stMUL_Poll_Type_Setup[14] = "1,0,5,Custom,30,5,6";
- stMUL_Poll_Type_Setup[15] = "2,0,5,ClassPoll,1,1,1";
- stMUL_Poll_Type_Setup[16] = "2,0,4,Custom,30,1,6";

Index,DLL Confirm,Time,Poll Type,

Index = Index of Outstation Com Handle

DLL Confirm = 0 = UnConfirmed User Data 1 = Confirmed User Data

Time = Refresh interval in seconds

Poll Type

   Integrity = This will poll Class 1, Class 2, Class 3 and Class 0

ClassPoll = This will poll Class 0 or Class 1, Class 2, and Class 3

ClassPoll,1,1,1 = will poll class 1 class 2 and class 3

ClassPoll,1,0,1 = will poll class 1 and class 3

ClassPoll,0,1,1 = will poll class 2 and class 3

ClassPoll,0,0,0 = will poll class 0

Custom = custom read   Custom,Group,Variation,Qualifier,Range Start,Range Stop

Group = 1, 2, 10, 20, 21, 22, 30, 32, 40

Qualifier = 0 or 1 for Group 1, 10, 20, 21, 30, 40

Qualifier = 6 for Group 1, 2, 10, 20, 21, 22, 30, 32, 40

Qualifier = 7 or 8 for Group 2, 22, 23, 32

CROB = This is Control Relay Output Block Group 12 Variation 1

CROB, Point Index, Function, Qualifier, CC, Count, On Time, Off Time

Point Index, Same index as Binary Output Group 10 of Outstation

Function 3 = Select/Operate   5 = Direct Operate   6 = Direct Operate NR

Qualifier = 17, 28

CC 1 = Nul Pulse On   3 = Nul Latch On   4 = Nul Latch Off   20 = Nul Nul

CC 21 = Nul Pulse On   23 = Nul Latch On   24 = Nul Latch Off

41 = Close Pulse On CC 61 = Close Pulse On 81 = Trip Pulse On

A1 = Trip Pulse On   0 = Nul Nul

Count = Number of Time the Outstation will Execute the Operation

On Time in milliseconds

Off Time in milliseconds

AOB = Analog Output Group 41 Variation 2 or 3

AOB, Point Index, Function, Variation, Qualifier, Requested Value Table Index

Point Index, Same index as Analog Output Group 40 of Outstation

Function 3 = Select/Operate   5 = Direct Operate   6 = Direct Operate NR

Variation = 2 = 16-bit   3 = floating-point

Qualifier = 17, 28

Requested Value -32768 to 32767 or float

DNP3 Master Manual Polling

There are several manual requests setup and can be selected by index number.

To send a manual request set:

nMULMaster_Manual_Request (from stMULManualCommandsDesc)(0-42)

nMULManual_Request_Enabled (Outstation Index of Com Handle)(0-9)

After (nMULManual_Request_Enabled) is set the request is sent

The master chart checks the variable **nMULManual_Request_Enabled**.

When the value is 0 – 9 it will send the master request before sending the next auto poll request. After sending the manual request it will set the value of **nMULManual_Request_Enabled** to -1.

For Manual CROB Set

nMULMaster_Manual_CROB_Function (3,5,6)

nMULMaster_Manual_CROB_Point_Index

nMULMaster_Manual_CROB_Qualifier

nMULMaster_Manual_CROB_Count

nMULMaster_Manual_CROB_OnTime

nMULMaster_Manual_CROB_OffTime

nMULMaster_Manual_Request (from stMULManualCommandsDesc) (50- 61)

nMULManual_Request_Enabled (Outstation Index of Com Handle)(0-9)

After **nMULManual_Request_Enabled** is set the request is sent.

For Manual AOB Set

nMULMaster_Manual_AOB_Function (3,5,6)

nMULMaster_Manual_AOB_Point_Index

fMULManual_AOB_Requested_Value

nMULMaster_Manual_Request (from stMULManualCommandsDesc)(70 - 73)

nMULManual_Request_Enabled (Outstation Index of Com Handle)(0-9)

After (nMULManual_Request_Enabled) is set the request is sent

```
//Manual command list
stMULManualCommandsDesc[0] = "Reset Link States";
stMULManualCommandsDesc[1] = "Test Link States";
stMULManualCommandsDesc[2] = "Request Link Status";
stMULManualCommandsDesc[3] = "Read Binary Input g1v0q6";
```

```
stMULManualCommandsDesc[4] = "Read Binary Input Events g2v0q6";
stMULManualCommandsDesc[5] = "Read Binary Input Events g2v1q6";
stMULManualCommandsDesc[6] = "Read Binary Input Events g2v2q6";
stMULManualCommandsDesc[7] = "Read Binary Input Events g2v3q6";
stMULManualCommandsDesc[8] = "Read Binary Outputs g10v0q6";
stMULManualCommandsDesc[9] = "Read Counter g20v0q6";
stMULManualCommandsDesc[10] = "Read Frozen Counter g21v0q6";
stMULManualCommandsDesc[11] = "Read Counter Events g22v0g6";
stMULManualCommandsDesc[12] = "Read Analog Input g30v0q6";
stMULManualCommandsDesc[13] = "Read Analog Input Events g32v0q6";
stMULManualCommandsDesc[14] = "Read Analog Output g40v0q6";
stMULManualCommandsDesc[15] = "Read Class 0 Objects g60v1q6";
stMULManualCommandsDesc[16] = "Read Class 1 Objects g60v2q6";
stMULManualCommandsDesc[17] = "Read Class 2 Objects g60v3q6";
stMULManualCommandsDesc[18] = "Read Class 3 Objects g60v4q6";
stMULManualCommandsDesc[19] = "Read Class 1,2,3 Objects";
stMULManualCommandsDesc[20] = "Read Integrity Poll";

stMULManualCommandsDesc[30] = "Write Time and Date g50v1q7";
stMULManualCommandsDesc[31] = "Write Clear IIN1.7 Device Restart g80v1q7";
stMULManualCommandsDesc[32] = "Enable Unsolicited Class 1, 2 , 3 Function 20";
stMULManualCommandsDesc[33] = "Disable Unsolicited Class 1, 2 , 3 Function 21";
stMULManualCommandsDesc[34] = "Immediate Freeze Counters Function 7";
stMULManualCommandsDesc[35] = "Immediate Freeze Analog Inputs Function 7";
stMULManualCommandsDesc[36] = "Immediate Freeze NR Counters Function 8";
stMULManualCommandsDesc[37] = "Immediate Freeze NR Analog Inputs Function 8";
stMULManualCommandsDesc[38] = "Freeze and Clear Counters Function 9";
stMULManualCommandsDesc[39] = "Freeze and Clear NR Counters Function 10";
stMULManualCommandsDesc[40] = "Cold Restart Function 13";
stMULManualCommandsDesc[41] = "Warm Restart Function 14";
stMULManualCommandsDesc[42] = "Delay Measurement Function 23";

stMULManualCommandsDesc[50] = "CROB 0x00 NUL/NUL/ClearF=0";
stMULManualCommandsDesc[51] = "CROB 0x01 NUL/Pulse On/ClearF=0";
stMULManualCommandsDesc[52] = "CROB 0x03 NUL/Latch On/ClearF=0";
stMULManualCommandsDesc[53] = "CROB 0x04 NUL/Latch Off/ClearF=0";
stMULManualCommandsDesc[54] = "CROB 0x41 Close/Pulse On/ClearF=0";
```

stMULManualCommandsDesc[55] = "CROB 0x81 Trip/Pulse On/ClearF=0";

stMULManualCommandsDesc[56] = "CROB 0x20 NUL/NUL/ClearF=1";
stMULManualCommandsDesc[57] = "CROB 0x21 NUL/Pulse On/ClearF=1";
stMULManualCommandsDesc[58] = "CROB 0x23 NUL/Latch On/ClearF=1";
stMULManualCommandsDesc[59] = "CROB 0x24 NUL/Latch Off/ClearF=1";
stMULManualCommandsDesc[60] = "CROB 0x61 Close/Pulse On/ClearF=1";
stMULManualCommandsDesc[61] = "CROB 0xA1 Trip/Pulse On/ClearF=1";

stMULManualCommandsDesc[70] = "AOB g41v2q17";
stMULManualCommandsDesc[71] = "AOB g41v2q28";
stMULManualCommandsDesc[72] = "AOB g41v3q17";
stMULManualCommandsDesc[73] = "AOB g41v3q28";

Adding or Modifying Manual Requests

Block 225 MInit lists the description of the commands. This is used by the operator to select the needed request.

Block 61 MAL State moves the needed variables. The case numbers matches the indexes of the description table.

Example

In block 225

```
stMULManualCommandsDesc[13] = "Read Analog Input Events g32v0q6";
```

In block 61

```
case 13://Read Analog Input Events g32v0q6
   ntMDLL_Request_CodePri[nMNL_Current_Outstation_Index] = 4;//DLL Unconfirmed
   nMAL_Fragment_Send_Message = 32;
   nMULMaster_Current_Variation = 0;
   nMULMaster_Current_Qualifier = 6;
break
```

The variable *nMAL_Fragment_Send_Message* is a case number that is built in block 103.

Block 103 MBuild AL Fragment builds the AL section of the request using the variables from block 61.

This block builds the AL section for both the manual and auto requests. There is logic to determine which variable to use.

Example

```
case 32://Analog Input Event Read
  ntMAL_Fragment_Send_Buffer_0[2] = 32;
  ntMAL_Fragment_Send_Buffer_0[3] = nMULMaster_Current_Variation;
  ntMAL_Fragment_Send_Buffer_0[4] = nMULMaster_Current_Qualifier;
  if (ntMAL_Fragment_Send_Buffer_0[4] == 7)then
     ntMAL_Fragment_Send_Buffer_0[5]=ntMULMaster_Auto_Range_Start[nMULRequestIndex];
     nMAL_Fragment_LastIndexUsed_0 = 5;
  elseif (ntMAL_Fragment_Send_Buffer_0[4] == 8)then
     ntMAL_Fragment_Send_Buffer_0[5] = ntMULMaster_Auto_Range_Start[nMULRequestIndex] bitand 0xFF;
     ntMAL Fragment Send Buffer 0[6] = ntMULMaster Auto Range Start[nMULRequestIndex] >> 8 bitand 0xFF
     nMAL_Fragment_LastIndexUsed_0 = 6;
  else
     nMAL_Fragment_LastIndexUsed_0 = 4;
  endif
break
```

In this example the operator would set a value of 13 (from description table index) in the variable **nMULMaster_Manual_Request**

Then set the value 0 -9 depending on the outstation com handle needed in the variable **nMULManual_Request_Enabled**. When this variable is >= 0 block 225 will load the variables for the manual request. After the request is processed the strategy will set **nMULManual_Request_Enabled** to a value of -1.

DNP3 Master Events

Events are processed and the last 1000 are stored in the string table **stMUL_Event_Log**.

Com Handle Index/gvq or Ugvq/Object Name Index = Value/Flag = Value

gvq = Group Variation Qualifier

Ugvq = Group Variation Qualifier This was received by the **Master_UnSol** chart.

DNP3 Master Errors

The last 100 DNP errors are stored string table **stMDNPError_Log**.

Error Code,Com Handle Index,Send Message #,Error Block #

ntMDNP_Received_Frame_Status list he last error for each com handle.

Error Codes

0 = OK

1 = No Response

2 = Bad Start Octets

3 = Not From Outstation

4 = Bad Header CRC

5 = Bad CRC bit

6 = Retransmit Ack

7 = Start Octets Not Found

8 = NACK Sent

9 = Not Supported

10 = NACK Received

11 = Busy

12 = TL Error Assembly

13 = AL State Error

14 = Sequence Number No Match

15 = AL State Error Uns

16 = Com Handle Error

17 = Incorrect Function

18 = FIR = 0

19 = FIR = 1

20 = TL Error Idle

Additional Details

The table **ntMDLL_Erroneous_Response** lists the number of erroneous responses for each com handle.

The table **ntMALRequest_Response_Status_Count** keeps a request/response count.

When a request is sent the value at the index for the outstation com handle is incremented.

When the response is received it is decremented.

An Increasing count indicates that no valid response is received for a request.

Set the variable **nMDNPResetErrorTables** true to reset these tables:

ntMALRequest_Response_Status_Count

ntMDLL_Erroneous_Response

stMDNPError_Log

Using the Subroutines

Adding Subroutines

The ADD Event subroutines allow an Opto 22 controller to add Events to the DNP Event Buffer. Include the subroutines before importing the charts into your strategy. Each subroutine in the integration kit supports one group and can function independently of the other subroutines. Therefore, you need only use the subroutines for the group that you require.

The subroutines can be used as needed in a customer chart. Or, if Auto Events is configured, they will be used by the DNP3_Outstation_Auto_Events chart. The DNP3_Manual_Event_Example chart is included as an example of a customer chart using the subroutines.

If Auto Events are configured the subroutine for that group must be included.

The subroutine Master_TimeStamp is used by the DNP3_Master chart and must be included for the master to function correctly.

For more information about subroutines, see form 1700, the *PAC Control User's Guide*.

When you decide which subroutines you need, include them in your strategy as follows:

1. Start PAC Control in Configure mode and open the strategy that you intend to use with the integration kit.
2. Choose Configure > Subroutines Included to open the Subroutine Files dialog.
3. Click the Add button and use the browser to select each subroutine file (.ISB extension) you want to include in your strategy.
4. Click OK.

The subroutines appear in the Subroutines Included folder and are ready to be used in your strategy.

Configuring the Subroutines

The subroutine for groups 2, 22, and 32 allows you to set the class, variation, and point to add to the Master Station event buffers. The subroutine can add the event to any or all the Master Stations event buffer.

The following tables list the parameters for each group and describe the type of data for each parameter.

- "Add Group 2 Event" on page 26
- "Add Group 22 Event" on page 27
- "Add Group 23 Event" on page 28
- "Add Group 32 Event" on page 29

Add Group 2 Event

Parameter	Description
Class	Integer 32 Variable (1,2,3)
Variation	Integer 32 Variable (1,2,3)
Index	Integer 32 Variable (Position of Point In Pointer Table)
ntAL Internal in	pntAL_Internal_Indications (must use this table)
Gp 1 Binary Pt	potDNP_Binary_Inputs_GP_1 (must use this table)
Master Buffer #	Integer 32 Variable (Set Bit To include in Master Buffer) Bit 0 = add to master 0 buffer Bit 1 = add to master 1 buffer Bit 2 = add to master 2 buffer Bit 3 = add to master 3 buffer
Event Buffer pt	potEvent_Buffer (must use this table)
Group_1_Flag pt	potDNP_Group_1_Flag (must use this table)
Lock Flag pt	potUL_Event_Buffer_Flag_Lock (must use this table)
potTimeStamp	potTimeStamp (must use this table)
ntNew_Event_Flag	ntNew_Event_Flag (must use this table)
Sub Status	Integer 32 Variable (Table Lock Status)
Put Status In	Integer 32 Variable

Add Group 22 Event

Parameter	Description
Class	Integer 32 Variable (1,2,3)
Variation	Integer 32 Variable (1,2,5,6)
Index	Integer 32 Variable (Position of Point In Pointer Table)
ntAL Internal in	pntAL_Internal_Indications (must use this table)
Gp 20 Counter pt	potDNP_Counters_Gp_20 (must use this table)
Master Buffer #	Integer 32 Variable (Set Bit To include in Master Buffer) Bit 0 = add to master 0 buffer Bit 1 = add to master 1 buffer Bit 2 = add to master 2 buffer Bit 3 = add to master 3 buffer
Event Buffer pt	potEvent_Buffer (must use this table)
Group 21 Flag t	ntDNP_Variable_32_Group_20_Flag (must use this table)
Lock Flag pt	potUL_Event_Buffer_Flag_Lock (must use this table)
potTimeStamp	potTimeStamp (must use this table)
ntNew_Event_Flag	ntNew_Event_Flag (must use this table)
Sub Status	Integer 32 Variable (Table Lock Status)
Put Status In	Integer 32 Variable

Add Group 23 Event

Parameter	Description
Class	Integer 32 Variable (1,2,3)
Variation	Integer 32 Variable (1,2,5,6)
Index	Integer 32 Variable (Position of Point In Pointer Table)
ntAL Internal in p	ntAL_Internal_Indications (must use this table)
Gp 21 Frozen Ct	ntDNP_Frozen_Counters_Gp_21 (must use this table)
Master Buffer #	Integer 32 Variable (Set Bit To include in Master Buffer) Bit 0 = add to master 0 buffer Bit 1 = add to master 1 buffer Bit 2 = add to master 2 buffer Bit 3 = add to master 3 buffer
Event Buffer pt	potEvent_Buffer (must use this table)
Lock Flag pt	potUL_Event_Buffer_Flag_Lock (must use this table)
potTimeStamp	potTimeStamp (must use this table)
ntNew_Event_Flag	ntNew_Event_Flag (must use this table)
Sub Status	Integer 32 Variable (Table Lock Status)
Put Status In	Integer 32 Variable

Add Group 32 Event

Parameter	Description
Class	Integer 32 Variable (1,2,3)
Variation	Integer 32 Variable (1,2,3,4,5,7)
Index	Integer 32 Variable (Position of Point In Pointer Table)
ntAL Internal in	pntAL_Internal_Indications (must use this table)
Gp 30 Analog pt	potDNP_Analog_Inputs_Gp_30 (must use this table)
Master Buffer #	Integer 32 Variable (Set Bit To include in Master Buffer) Bit 0 = add to master 0 buffer Bit 1 = add to master 1 buffer Bit 2 = add to master 2 buffer Bit 3 = add to master 3 buffer
Event Buffer pt	potEvent_Buffer(must use this table)
Group_30_Flag pt	potDNP_Group_30_Flag(must use this table)
Lock Flag pt	potUL_Event_Buffer_Flag_Lock (must use this table)
potTimeStamp	potTimeStamp (must use this table)
ntNew_Event_Flag	ntNew_Event_Flag (must use this table)
Sub Status	Integer 32 Variable (Table Lock Status)
Put Status In	Integer 32 Variable