Kepware Technologies

DNP3 & the Control Relay Output Block Command

May, 2011
V. 1.000

Table of Contents

1. Overview

This document intends to discuss DNP3's Control Relay Output Block - Binary Output Command.

2. DNP3 Object Group 12 for Binary Output Points

DNP3 Object Group 12 for Binary Output Points is used to provide a digital On-Off drive signal or a pulsed drive signal for controlling a real or pseudo output device.

To perform digital control operations (such as electro-mechanical relays) at binary output points, use tags addressed with Object Group 10. An operation on an Object Group 10 Tag will issue the CROB command using Object Group 12, variation 1.

2.1 Output Models

There are three output models that apply to binary outputs. Descriptions of the models are as follows:

- **Activation:** This output type initiates an action. For example, Initiate Test, Acknowledge Alarm, and Trip Breaker.
- **Complementary Latch:** This output type remains latched in an active or non-active state depending on which command is received. For example, Illumination On-Off, Enable-Disable, and Auto-Manual.
- **Complementary Two-Output:** This output type sets either a close or trip output momentarily active depending on which command is received. For example, Run-Stop Motor, Trip-Close Breaker, Raise-Lower Transformer Tap, and Open-Close Valve.

2.2 Vendors' Roles

Device vendors are expected to assign the device appropriate control codes. For example, a manufacturer may assign "CLOSE -- PULSE_ON" and "TRIP -- PULSE_ON" to a breaker, and "NUL -- LATCH_ON" and "NUL -- LATCH_OFF" to a pseudo point.

When performing On-Off and Close-Trip controls, it is recommended that users utilize the **Direct Operate** or the **Select then Operate** function codes with Group 12 objects. For more information, refer to Direct Operate and Select then Operate.

2.3 Device Properties

Descriptions of the Device Properties are as follows:

- **Operate Mode:** This parameter specifies whether the writable I/O points (Object Group 10 - Binary Outputs and Object Group 40 - Analog Outputs) will use the Direct Operate or the Select then Operate sequence. The default setting is Direct Operate.

  Note: Individual tags' write behavior can override this setting by writing a Boolean True to the output's corresponding .DO or .SO sub-type tags.
- **Enable Feedback Poll after Write:** When checked, a feedback poll will occur after a write. The default setting is checked.

2.3.1 Direct Operate

DIRECT_OPERATE function codes are similar to the OPERATE function code except that they do not require a preceding select command. A Direct Operate function code is used for outputting Control Relay Output Blocks, Pattern Control Blocks, and Analog Outputs when the extra security provided by a two-step control command is not necessary. It also optimizes bandwidth utilization in a closed loop control when other feedback is present.

- Direct Operate request messages look identical to Select and Operate request messages except for the function code. They contain one or more objects that describe the desired output state or level.
- Direct Operate responses contain an identical set of object headers and objects in the same order as they appear in the master's request (unless an error condition exists). The response to a DIRECT_OPERATE command does not guarantee that the execution occurred: it indicates that the request was received. For this reason, systems employing a DIRECT_OPERATE function code are encouraged to provide another way to detect whether the execution occurred.

Note 1: In the case of an error, the slave will set a status code within an object that possibly omits objects and/or sets IIN bits.

Note 2: When the value written during a CROB command is not understood by the DNP slave, the following error message will be displayed in the Event Log: "Unable to write to '<address>' on device '<device>'. Control-Related Status Code ' <status code>'". For more information, refer to Control-Related Status Codes.

2.3.2 Select then Operate

The SELECT function code is used with the OPERATE function code as part of the two step Select-before-Operate method for issuing control requests. This procedure is used for controlling binary and analog outputs. In the general Select then Operate procedure, the master first sends a select request containing all of the necessary parameters (such as indexes, timings and values). The master then compares the Select response with the request. If they match, the master will issue an Operate command with the identical object set of headers and objects that it sent in the select message. This approach assures that the slave understands which control the master intended.

A slave's Select the Operate response contains an identical set of object headers and objects in the same order as they appear in the master's request (unless an error condition exists).

Note 1: In the case of an error, the slave will set a status code within an object that possibly omits objects and/or sets IIN bits from the response.

Note 2: When the DNP slave does not understand the value written during a CROB command, the following error message will be displayed in the Event Log: "Unable to write to '<address>' on device '<device>'. Control-Related Status Code '<status code>'". For information on the status codes, refer to Control-Related Status Codes.

2.3.2.1 Verifications

A complete two-step procedure has the two following verifications:

1. The master must receive a Select response that matches the request. That response must indicate no errors; otherwise, the master will abort the control.
2. The slave is obligated to compare the Operate request with the Select request. If the two match, and if there are no errors detected in the request, the slave will activate the outputs.

3.CROB Operate Command Example

The following example shows how to issue a Select then Operate request for Binary Output point 8 that closes the point one time for 250 milliseconds.

1. To start, configure the point's Operate Mode. This can be done for the entire device or on an individual point basis.

    a. To configure it for the entire device, click **Device Properties | Advanced**. Then, set the **Operate Mode** to **Select Then Operate**.

    b. To configure it for an individual point (if the device Operate Mode is Direct Operate), first create a tag with the address "10.0.8.SO". Then, write "1" to the .SO Tag. This will set the point's Operate Mode point at index 8 to **Select Then Operate**.

2. Next, click **Device Properties | Advanced**. Then, check or uncheck **Enable Feedback Poll after Write** to specify the desired behavior.
3. Next, create a tag with the address "10.0.8.Operate.Optype".
4. Write "1" to the .Operate.Optype Tag. This will set the Control Code's Operation Type field to **Pulse On**.
5. Next, create a tag with the address "10.0.8.Operate.Tripclosecode".
6. Write "1" to the .Operate.Tripclosecode Tag. This will set the Control Code's Trip-Close Code field to **Close**.
7. Next, create a tag with the address "10.0.8.Operate.Ontime".
8. Write "250" to the .Operate.Ontime Tag. This specifies the duration in milliseconds that the output drive will remain active.
9. Next, create a tag with the address "10.0.8.Operate.Set".
10. Write "1" to .Operate.Set Tag. This will trigger the master to send the Object Group 12, variation 1 request that will perform the digital control operation.

Note: Channel Diagnostics should display the following three transactions:

1. **TX:** The master sends a request using function code 0x03 to select the output point.
   RX: The slave responds echoing the request if everything is okay.

2. **TX:** The master then sends the Operate request using function code 0x04.
   RX: The slave responds echoing the request if everything is okay.

3. **TX:** The master then sends the feedback poll.
   RX: The slave responds with the current static value for all binary outputs.

3.1.1 TX and RX Parsing Example

The following example displays an attempt to transmit a Direct Operate that closes the binary output point index 100 one time for 800 milliseconds. The response indicates that the slave has an issue with accepting the control operation to set binary output point 100 to active.

Note: For more information, refer to the color-coordinated subtopic.

<table>
<tr><td><b>TX</b></td><td>05 | 64 | 18 | C4 | 0A | 00 | 00 | 00 | 73 | 91 | D8 | C9 | 05 | 0C | 01 |17 | 01 | 64 | 41 |
01 | 20 | 03 | 00 | 00 | 00 | 00 |A3 | 92 | 00 | 00 | 00 | FF | FF |</td></tr>
<tr><td><b>RX</b></td><td>05 | 64 | 1A | 44 | 00 | 00 | 0A | 00 | 47 | 9F | F9 | C9 | 81 | 00 | 00 | 0C | 01 | 17 | 01 |
64 | 41 | 01 | 20 | 03 | 00 | 00 | 08 | 09 | 00 | 00 | 00 | 00 | 04 | 87 | 26 |</td></tr>
</table>

Transmit Packet - Data Link Layer Header Block

Hex Bytes	Description
05 64	Starts message.
18	Length of request - indicates 24 bytes.
C4	Data link layer control byte. The binary representation is 11000100. The bits are defined as follows: 7 DIR - 1 indicates direction from master. 6 PRM - 1 indicates primary request. 5 FCB - frame count bit. 4 FCV - frame count valid, 0 indicates ignore frame count bit. 3-0 function code - 4 for unconfirmed user data.
0A 00	Destination address - 0x000A.
00 00	Source address - 0x0000.
73 91	Check sum - 0x9173.

Transmit Packet - Transport Header

Hex Bytes	Description
D8	Transport control byte. The binary representation is 11011000. The bits are defined as follows: 7 FIN - 1 indicates final transport segment. 6 FIR - 1 indicates first transport segment. 5-0 sequence number - 24 used to verify correct order.

Transmit Packet - Application Header

Hex Bytes	Description
C9	Application control byte. The binary representation is 11001001. The bits are defined as follows: 7 FIR - 1 indicates first fragment of message. 6 FIN - 1 indicates final fragment of message. 5 CON - 0 indicates a confirmation message is not returned. 4 UNS - 0 indicates the sequence number is associated with a solicited request. 3-0 sequence number - 9 used to verify correct order.

05	Function code - 5 direct operate.
0C	Object Group: 12.
01	Variation: 1.
17	Qualifier Field. The binary representation is 00010111. The bits are defined as follows: 7 always 0. 6-4 object prefix code 1 indicates objects prefixed with 1-octet index. 3-0 range specifier code 7 indicates 1-octet count of objects.
01	Range field 1 indicates count of 1 object.
64	Data point index 100.

Transmit Packet - Data Object

Hex Bytes	Description
41	Control Code. The binary representation is 01000001. The bits are defined as follows: 7-6 TCC - Trip Close Code - 01 - close. 5 CR - Clear - 0. 4 QU - Queue always 0. 3-0 Op Type - 1 - pulse on.
01	Count - number of times to execute.
20 03 00 00	On Time - 800 ms (0x0000320).
00 00	First part of off time.
A3 92	Check sum - 0x92A3.
00 00	Second part of off time.
00	Always 0 in request.
FF FF	Check sum - 0xFFFF.

Receive Packet - Data Link Layer Header Block

Hex Bytes	Description
05 64	Starts message.
1A	Length of request - indicates 26 bytes.
44	Data link layer control byte. The binary representation is 01000100. The bits are defined as follows: 7 DIR - 0 indicates direction from slave. 6 PRM - 1 indicates primary request. 5 FCB - frame count bit. 4 FCV - frame count valid, 0 indicates ignore frame count bit. 3-0 function code - 4 for unconfirmed user data.
00 00	Source address - 0x0000.
0A 00	Destination address - 0x000A.
47 9F	Check sum - 0x9F47.

Receive Packet - Transport Header

Hex Bytes	Description
F9	Transport control byte. The binary representation is 11111001. The bits are defined as follows: 7 FIN - 1 indicates final transport segment. 6 FIR - 1 indicates first transport segment. 5-0 sequence number - 57 used to verify correct order.

Receive Packet - Application Header

Hex Bytes	Description
C9	Application control byte. The binary representation is 11001001. The bits are defined as follows: 7 FIR - 1 indicates first fragment of message. 6 FIN - 1 indicates final fragment of message. 5 CON - 0 indicates a confirmation message is not returned. 4 UNS - 0 indicates the sequence number is associated with a solicited request. 3-0 sequence number - 9 used to verify correct order.
81	Function code - 129 Response.
00 00	Internal Indication bits, none set.
0C	Object group: 12.
01	Variation: 1.
17	Qualifier Field. The binary representation is 00010111. The bits are defined as follows: 7 always 0. 6-4 object prefix code 1 indicates objects prefixed with 1-octet index. 3-0 range specifier code 7 indicates 1-octet count of objects.
01	Range Field 1 indicates count of 1 object.
64	Data point index 100.

Receive Packet - Data Object

Hex Bytes	Description
41	Control Code. The binary representation is 01000001. The bits are defined as follows: 7-6 TCC - Trip Close Code - 01 - close. 5 CR - Clear - 0. 4 QU - Queue always 0. 3-0 Op Type - 1 - pulse on.
01	Count - number of times to execute.
20 03 00 00	On Time - 800 ms (0x00000320).
08 09	Check sum - 0x0908.
00 00 00 00	Off time.
04	Status code 4.
87 26	Check sum - 0x2687.

3.1.2 Tag Addressing Examples

Tag Address	Definition	Description
10.0.2.DO	Boolean value indicating if operations on binary output point 2 should be Direct Operate or Select then Operate.	Writing to this tag will not cause an explicit write to the device. It also will not change the overall device property for Operate Mode: it only changes it for Binary Output point 2. The variation of the tag does not matter. *
10.0.2.SO	Boolean value indicating if operations on binary output point 2 should be Direct Operate or Select then Operate.	Writing to this tag will not cause an explicit write to the device. It also will not change the overall device property for Operate Mode: it only changes it for Binary Output point 2. The variation of the tag does not matter.*
10.0.2.Operate.Clear	Value of the CROB control code bit 5.	This Boolean tag will display a 0 or 1, depending on the last update from the client. The variation of the tag does not matter. Writing to this tag will not cause an explicit write to the device. It will be used in building the CROB control code that will be written to the object group 12 point 2 with the Operate.Set tag.
10.0.2.Operate.OpType	Value of the CROB control code bits 0-3.	This Byte tag will display the operation type, depending on the last update from the client. Operation types are as follows: Nul (0) Pulse_On (1) Pulse_Off (2) Latch_On (3) Latch_Off (4) The variation of the tag does not matter. **
10.0.2.Operate.TripCloseCode	Value of the CROB control code bits 6 & 7.	This byte tag will display the Trip-Close Field, depending on the last update from the client. Trip-close fields are as follows: Nul (0) Paired_Close (1) Paired_Trip (2) The variation of the tag does not matter. **
10.0.2.Operate.OnTime	Value in milliseconds that the operation on the binary output point	This DWord tag will display the on time last updated from the client.

	2 will remain active.	The variation of the tag does not matter.***
10.0.2.Operate.OffTime	Value in milliseconds that the operation on the binary output point 2 will remain non-active.	This DWord tag will display the off time last updated from the client. The variation of the tag does not matter.***
10.0.2.Operate.FeedbackDelay	Value in milliseconds to delay after receiving the response before issuing feedback poll.	This DWord tag will display the feedback delay last updated from the client. The variation of the tag does not matter. Writing to this tag will not cause an explicit write to the device. It will be used to delay before issuing a feedback poll after receiving a response to an Object Group 12 control operation on binary output point 2.
10.0.2.Operate.Set	Always displays a Boolean value of 0 with Good quality.	The variation of the tag does not matter. Writing a 1 to this tag causes an Object Group 12 control operation on binary output point 2. The CROB control code is built from the values of the Operate.Clear, Operate.OpType, and Operate.TripCloseCode tags. The Values of the Operate.OnTime, Operate.OffTime, and Operate.FeedbackDelay tags are used in the operation as well.

*The value of this tag will be used when an operation is performed on binary output point 2 when using the Operate sub-type commands or a synchronous/asynchronous write to a 10.x.2.Value or 10.x.2.Explicit Tag.

**Writing to this tag will not cause an explicit write to the device. It will be used in building the CROB control code that will be written to the Object Group 12 point 2 with the Operate.Set Tag.

***Writing to this tag does not cause an explicit write to the device. It will be used in an Object Group 12 control operation on binary output point 2 when the Operate.Set Tag is toggled to 1.

4. Slave-Performed Actions for Interoperable Commands

For more information, refer to the table below.

Row #	Control Code	TCC Field	OP Type Field	Clear Field	Action
1	0x00	NUL	NUL	0	Does not initiate an action or change an in-progress or pending command. Values in On-time and Off-time fields are ignored.
2	0x20	NUL	NUL	1	Cancel in-progress and pending commands. Values in On-time and Off-time fields are ignored.
3	0x01	NUL	PULSE_ON	0	For activation model, set output to active for the duration of the On-time. For both complementary models, return NOT_SUPPORTED status.
4	0x21	NUL	PULSE_ON	1	For activation model, cancel in-progress and pending commands and then set output to active for the duration of the On-time. For both complementary models, return NOT_SUPPORTED status.
5	0x03	NUL	LATCH_ON	0	For activation model, set output to active for the duration of the On-time. For complementary latch model, set the output to active. For complementary two-output model, set the close output to active for the duration of the On-time.
6	0x23	NUL	LATCH_ON	1	Cancel in-progress and pending commands. Afterwards, initiate the action specified in row 5.
7	0x04	NUL	LATCH_OFF	0	For activation model, set output to active for the duration of the On-time. For complementary latch model, set the output to inactive. For complementary two-output model, set the trip output to active for the duration of the On-time.
8	0x24	NUL	LATCH_OFF	1	Cancel in-progress and pending commands. Afterwards, initiate the action specified in row 7.
9	0x41	CLOSE	PULSE_ON	0	For activation model, set output to active for the duration of the On-time. For complementary latch model, set the output to active. For complementary two-output model, set the close output to active for the duration of the On-time.
10	0x61	CLOSE	PULSE_ON	1	Cancel in-progress and pending commands. Afterwards, initiate the action specified in row 9.
11	0x81	TRIP	PULSE_ON	0	For activation model, set output to active for the duration of the On-time.

					For complementary latch model, set the output to inactive. For complementary two-output model, set the trip output to active for the duration of the On-time.
12	0xA1	TRIP	PULSE_ON	1	Cancel in-progress and pending commands. Afterwards, initiate the action specified in row 11.

5.Control-Related Event Log Messages

For more information, refer to the table below.

Message	Error Type	Possible Cause	Solution
Unable to write to '<address>' on device '<device>'. Control-Related Status Code ' <status code>'	Warning	The value written to the .Operate sub-type was not understood by the DNP slave.	For information on a specific code number, refer to Control-Related Status Codes.
Unable to write to '<address>' on device '<device>'. Device does not support requested operation for objects in the request (IIN2.1)	Warning	The device does not support the requested operation for the objects in the request.	Verify that the slave supports the requested operation.
Unable to write to '<address>' on device '<device>'. Device does not support the function code (IIN2.0)	Warning	The device does not support the function code on the objects in the request.	For more information, refer to the device's documentation.
Unable to write to '<address>' on device '<device>'. Device reports that some output points are in local mode (IIN1.5)	Warning	Some output points are in local mode.	Correct the mode in the slave's configuration. For more information, refer to the device's documentation.*
Unable to write to '<address>' on device '<device>'. Select Operate response invalid	Warning	The device did not return an acceptable response to a Select then Operate request.	Verify that the slave is configured to operate on the point.

*This bit must be set whenever one of the slave’s output points is in local operation mode. This indication is not intended to inhibit or prevent controls from the master: it is a warning that controls might not succeed. The system implementer must provide suitable lockouts that disable control operation from the master when points are in local mode. If the master directs a control command to a point in local mode, the slave shall return a status code indicating failure due to the point being in local mode. Masters may send control requests to slaves that set IIN1.5 because this bit does not necessarily mean that all points are disabled for operation from the master. For example, in a data concentrator application, only one IED may be in the local mode. This requires the data concentrator to set IIN1.5; however, control commands to the other IEDs should succeed. Slaves that have a local-remote switch or logic are responsible for preventing control actions on those points that it places in local mode.

6. Control-Related Status Codes

For more information, refer to the table below.

Code #	Identifier	Description
0	Success	Request accepted, initiated, or queued.
1	Timeout	Request not accepted because the operate message was received after the arm timer timed out. The arm timer was started when the select operation for the same point was received.
2	No_Select	Request not accepted because no previous matching select request exists. An operate message was sent to activate an output that was not previously armed with a matching select message.
3	Format_Error	Request not accepted because there were formatting errors in the control request (select, operate, or direct operate).
4	Not_Supported	Request not accepted because a control operation is not supported for this point.
5	Already_Active	Request not accepted because the control queue is full (or the point is already active).
6	Hardware_Error	Request not accepted because of control hardware problems.
7	Local	Request not accepted because the Local/Remote switch is in the Local position.
8	Too_Many_Objs	Request not accepted because too many objects appeared in the same request.
9	Not_Authorized	Request not accepted because of insufficient authorization.
10	Automation_Inhibit	Request not accepted because it was prevented or inhibited by a local automation process.
11	Processing_Limited	Request not accepted because the device cannot process any more activities than are presently in progress.
12	Out_Of_Range	Request not accepted because the value is outside the acceptable range permitted for this point.
13 to 125	Reserved	Reserved for future use.
126	Non_Participating	Sent in request messages indicating that the outstation will neither issue nor perform the control operation.
127	Undefined	Request not accepted due to an undefined reason.

7. Summary

At this point, users should have a better understanding of the DNPs Control Relay Output Block - Binary Output Command.