ROTO-CONTROL SERIAL API

Version 1.1

Disclaimer

The software API is provided "as is" without any warranties or guarantees of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. All parts of this API are subject to change, additions or deletions without notice. The use of the API is at your own risk. The authors or contributors of this API do not offer any technical support, updates, or maintenance, and make no representations regarding the performance, reliability, or suitability of the API for any particular purpose. In no event shall the authors or contributors be liable for any damages arising from the use or inability to use this API, including but not limited to direct, indirect, incidental, special, or consequential damages, even if advised of the possibility of such damages.

Revision History

Version	Date	Change
1.0	23/02/2025	First release.
1.1	07/04/2025	**This version is NOT backwards compatible with v1.0.** Updates to support MACRO PLUGINs and MACRO params: • GET CURRENT PLUGIN • GET FIRST PLUGIN • GET NEXT PLUGIN • GET PLUGIN • GET PLUGIN KNOB CONFIG • SET PLUGIN KNOB CONFIG

Compatibility

Version	Compatibility
1.0	• ROTO-SETUP v1.0.0 • ROTO-SETUP v1.1.1 • ROTO-SETUP v1.1.2
1.1	• ROTO-SETUP v1.1.3

1. Introduction

The ROTO-CONTROL SERIAL API allows an external device to query and configure a ROTO-CONTROL device via a simple binary format. The interface used is a serial (COM) port via USB.

The interface is bi-directional in that commands can be sent to ROTO-CONTROL and received from ROTO-CONTROL asynchronously:

> **TO ROTO**: External device sends command -> ROTO-CONTROL sends back response.
>
> **FROM ROTO**: ROTO-CONTROL sends a command to the external device asynchronously; no external device response is needed.

The serial port configuration is as follows:

- 115200 baud, 8-bit data, no parity, 1 stop-bit.

Note 1: For conciseness ROTO-CONTROL is also referred to as ROTO throughout this document.

Note 2: All values are specified in hexadecimal.

1.2 Commands Overview

Type	Sub-type	Description	To ROTO	From ROTO
01: GENERAL	**01: GET FW VERSION**	Returns the ROTO-CONTROL firmware version.	Y	N
	02: GET MODE	Get the current ROTO-CONTROL mode.	Y	N
	03: SET MODE	Sets the ROTO-CONTROL mode.	Y	Y
	04: START CONFIG UPDATE	Start an update of a ROTO-CONTROL config	Y	N
	05: END CONFIG UPDATE	End an update of a ROTO-CONTROL config	Y	N
	06: FACTORY RESET	Performs a factory rest of the ROTO-CONTROL unit.	Y	N
02: MIDI MODE	**01: GET CURRENT SETUP**	Returns the current MIDI setup	Y	N
	02: GET SETUP	Returns the specified MIDI setup	Y	N
	03: SET SETUP	Selects the specified MIDI setup	Y	Y
	04: SET SETUP NAME	Sets the current MIDI setup name		
	05: GET KNOB CONTROL CONFIG	Returns the configuration of a MIDI knob control.	Y	N
	06: GET SWITCH CONTROL CONFIG	Returns the configuration of a MIDI switch control	Y	N
	07: SET KNOB CONTROL CONFIG	Sets the configuration of a MIDI knob control.	Y	N

	08: SET SWITCH CONTROL CONFIG	Sets the configuration of a MIDI switch control.	Y	N
	09: CLEAR CONTROL CONFIG	Clears a knob or switch control config	Y	N
	0A: CLEAR MIDI SETUP	Clears a MIDI setup	Y	N
	0B: MIDI CONTROL LEARNED	A MIDI control was learned on ROTO-CONTROL	N	Y
03: PLUGIN MODE	**01: GET CURRENT PLUGIN**	Returns the current PLUGIN config	Y	N
	02: GET FIRST PLUGIN	Returns the first PLUGIN config	Y	N
	03: GET NEXT PLUGIN	Returns the next PLUGIN config, call multiple times to get all device PLUGIN configs	Y	N
	04: GET PLUGIN	Gets the specified PLUGIN	Y	N
	05: SET PLUGIN	The specified PLUGIN has been selected	N	Y
	06: ADD PLUGIN	Adds the specified PLUGIN	Y	N
	07: SET PLUGIN NAME	Set the PLUGIN name	Y	N
	08: CLEAR PLUGIN	Clear (delete) the PLUGIN	Y	N
	09: GET PLUGIN KNOB CONFIG	Get the PLUGIN knob control config	Y	N
	0A: GET SWITCH CONFIG	Gets the PLUGIN switch control config	Y	N
	0B: SET PLUGIN KNOB CONFIG	Sets the PLUGIN knob control config	Y	N
	0C: SET PLUGIN SWITCH CONFIG	Sets the PLUGIN switch control config	Y	N
	0D: CLEAR PLUGIN CONTROL CONFIG	Clears the PLUGIN knob or switch control config	Y	N
	0E: PLUGIN CONTROL LEARNED	A PLUGIN control was learned on ROTO-CONTROL	N	Y

Note: For each command, the first byte of the response is A5 followed by the response code. If this byte indicates an error, no further specified bytes will follow.

2. GENERAL Commands: 01

2.1 GET FW VERSION: 01

TO ROTO: Y FROM ROTO: N

Command
5A 01 01 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
0xA5 <RC VX VY VZ GC:7> RC = Response code: SUCCESS (00), ERROR (all other values) VX = ROTO-CONTROL major version VY = ROTO-CONTROL minor version VZ = ROTO-CONTROL patch version GC = Short GIT commit in ASCII bytes

2.2 GET MODE: 02

TO ROTO: Y FROM ROTO: N

Command
0x5A 01 02 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC AM PI> RC = Response code: SUCCESS (00), ERROR (all other values) AM = ROTO-CONTROL Mode: MIDI (00), PLUGIN (01), MIX (02) PI = Page index in multiples of 8 (00 = Page 1, 08 = Page 2, etc.)

2.3 SET MODE: 03

TO ROTO: Y FROM ROTO: Y

Command
5A 01 03 <CL:2 AM PI> CL = Command data length, MSB followed by LSB = 0002 AM = ROTO-CONTROL Mode: MIDI (00), PLUGIN (01), MIX (02) PI = Page index in multiples of 8 (00 = Page 1, 08 = Page 2, etc.)
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

2.4 START CONFIG UPDATE: 04

TO ROTO: Y FROM ROTO: N

Command
5A 01 04 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC>

RC = Response code: SUCCESS (00), ERROR (all other values)

2.5 END CONFIG UPDATE: 05

TO ROTO: Y FROM ROTO: N

Command
5A 01 05 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

2.6 FACTORY RESET: 06

TO ROTO: Y FROM ROTO: N

Note: This command reformats the file system, all saved MIDI setups and PLUGIN configs will be erased.

Command
5A 01 06 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

3. MIDI Commands: 02

3.1 GET CURRENT SETUP: 01

TO ROTO: Y FROM ROTO: N

Command
5A 02 01 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC SI SN:0D> RC = Response code: SUCCESS (00), ERROR (all other values) SI = Setup index 00 - 3F SN = Setup name: 0D-byte NULL terminated ASCII string, padded with 00s if needed

3.2 GET SETUP: 02

TO ROTO: Y FROM ROTO: N

Command
5A 02 02 <CL:2 SI> CL = Command data length, MSB followed by LSB = 0001 SI = Setup index: 00 – 3F
Response
A5 <RC SI SN:0D> RC = Response code: SUCCESS (00), ERROR (all other values) SI = Setup index 00 - 3F SN = Setup name: 0D-byte NULL terminated ASCII string, padded with 00s if needed

3.3 SET SETUP: 03

TO ROTO: Y FROM ROTO: Y

Command
5A 02 03 <CL:2 SI> CL = Command data length, MSB followed by LSB = 0001 SI = Setup index: 00 – 3F
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

3.4 SET SETUP NAME: 04

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 02 04 <CL:2 SI SN:0D> CL = Command data length, MSB followed by LSB = 000E SI = Setup index: 00 – 3F

SN = Setup name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), ERROR (all other values)

3.5 GET KNOB CONTROL CONFIG: 05

TO ROTO: Y FROM ROTO: N

Command
5A 02 05 <CL:2 SI CI>
CL = Command data length, MSB followed by LSB = 0001
SI = Setup index: 00 – 3F
CI = Control index: 00 – 1F
Response
A5 <RC SI CI CM CC CP NA:2 MN:2 MX:2 CN:0D CS HM IP1 IP2 HS SN:10*0D>
RC = Response code: SUCCESS (00), ERROR (all other values)
SI = Setup index: 00 – 3F
CI = Control index: 00 - 1F
CM = Control Mode: CC-7BIT (00), CC-14BIT (01), NRPN-7BIT (02), NRPN-14-BIT (03)
CC = Control channel: 01 - 10
CP = Control param: set to FF if unused
NA = NRPN address
MN = Min value, set the MSB to 00 for 7-BIT mode
MX = Max value, set the MSB to 00 for 7-BIT mode
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = Colour scheme: 00 - 52
HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01)
IP1 = Indent position 1: 00 – 7F, FF if unused, only applies for KNOB_300
IP2 = Indent position 2: 00 – 7F, FF if unused, only applies for KNOB_300
HS = Haptic steps: 02 - 10, only applies for KNOB_N_STEP
SN = An array of 10 x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed

3.6 GET SWITCH CONTROL CONFIG: 06

TO ROTO: Y FROM ROTO: N

Command
5A 02 06 <CL:2 SI CI>
CL = Command data length, MSB followed by LSB = 0002
SI = Setup index: 00 – 3F
CI = Control index: 00 – 1F
Response
A5 <RC SI CI CM CC CP NA:2 MN:2 MX:2 CN:0D CS LN LF HM HS SN:10*0D>
RC = Response code: SUCCESS (00), ERROR (all other values)
SI = Setup index: 00 – 3F
CI = Control index: 00 – 1F
CM = Control Mode: CC-7BIT (00), CC-14BIT (01), NRPN-7BIT (02), NRPN-14-BIT (03)
CC = Control channel: 01 - 10
CP = Control param

NA = NRPN address
MN = Min value, set the MSB to 00 for 7-BIT mode
MX = Max value, set the MSB to 00 for 7-BIT mode
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = Colour scheme: 00 - 52
LN = LED ON colour: 00 - 52
LF = LED OFF colour: 00 - 52
HM = Haptic mode: PUSH (00), TOGGLE (01)
HS = Haptic steps: 00 or 02 - 10
SN = An array of 10 x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed

3.7 SET KNOB CONTROL CONFIG: 07

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 02 07 <CL:2 SI CI CM CC CP NA:2 MN:2 MX:2 CN:0D CS HM IP1 IP2 HS SN:HS*0D>
CL = Command data length, MSB followed by LSB = 001D + (HS * 0D)
SI = Setup index: 00 – 3F
CI = Control index: 00 – 1F
CM = Control Mode: CC-7BIT (00), CC-14BIT (01), NRPN-7BIT (02), NRPN-14-BIT (03)
CC = Control channel: 01 - 10
CP = Control param: Set to FF if unused
NA = NRPN address
MN = Min value, set the MSB to 00 for 7-BIT mode
MX = Max value, set the MSB to 00 for 7-BIT mode
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = CS = Colour scheme: 00 - 52
HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01)
IP1 = Indent position 1: 00 – 7F, FF if unused, only applies for KNOB_300
IP2 = Indent position 2: 00 – 7F, FF if unused, only applies for KNOB_300
HS = Haptic steps: 02 - 10, only applies for KNOB_N_STEP
SN = An array of HS x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), ERROR (all other values)

3.8 SET SWITCH CONTROL CONFIG: 08

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 02 08 <CL:2 SI CI CM CC CP NA:2 MN:2 MX:2 CN:0D CS LN LF HM HS SN:HS*0D>
CL = Command data length, MSB followed by LSB = 001D + (HS * 0D)

SI = Setup index 00 - 3F CI = Control index: 00 – 1F CM = Control Mode: CC-7BIT (00), CC-14BIT (01), NRPN-7BIT (02), NRPN-14-BIT (03) CC = Control channel: 01 - 10 CP = Control param NA = NRPN address MN = Min value, set the MSB to 00 for 7-BIT mode MX = Max value, set the MSB to 00 for 7-BIT mode CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed CS = CS = Colour scheme: 00 - 52 LN = LED ON colour: 00 - 52 LF = LED OFF colour: 00 - 52 HM = Haptic mode: PUSH (00), TOGGLE (01) HS = Haptic steps: 00 or 02 – 10, set to 00 if a normal two position switch with no haptic strings SN = An array of HS x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

3.9 CLEAR CONTROL CONFIG: 09

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 02 09 <CL:2 SI CT CI> CL = Command data length, MSB followed by LSB = 0003 SI = Setup index 00 - 3F CT = Control type: KNOB (00), SWITCH (01) CI = Control index: 00 – 1F
Response
A5 <RC> RC = Response code: SUCCESS (00), ERROR (all other values)

3.10 CLEAR MIDI SETUP: 0A

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 02 0A <CL:2 SI> CL = Command data length, MSB followed by LSB = 0001 SI = Setup index 00 - 3F
Response
A5 <RC>

RC = Response code: SUCCESS (00), ERROR (all other values)

3.11 MIDI CONTROL LEARNED: 0B

TO ROTO: N FROM ROTO: Y

Note: This command is also sent from ROTO-CONTROL when a MIDI control is cleared.

Command
5A 02 0B <CL:2 SI CT CI>
CL = Command data length, MSB followed by LSB = 0003
SI = Setup index 00 - 3F
CT = Control type: KNOB (00), SWITCH (01)
CI = Control index: 00 – 1F
Response
N/A

4. PLUGIN Commands: 03

4.1 GET CURRENT PLUGIN: 01

TO ROTO: Y FROM ROTO: N

Command
5A 03 01 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 PN:0D PM> RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values) PH = Plugin hash PN = Plugin name: 0D-byte NULL terminated ASCII string, padded with 00s if needed PM = MACRO Plugin: NO (00), YES (01)

4.2 GET FIRST PLUGIN: 02

TO ROTO: Y FROM ROTO: N

Command
5A 03 02 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 SN:0D PM> RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values) PH = Plugin hash SN = Setup name: 0D-byte NULL terminated ASCII string, padded with 00s if needed PM = MACRO Plugin: NO (00), YES (01)

4.3 GET NEXT PLUGIN: 03

TO ROTO: Y FROM ROTO: N

Command
5A 03 03 <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 PN:0D PM> RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values) PH = Plugin hash PN = Plugin name: 0D-byte NULL terminated ASCII string, padded with 00s if needed PM = MACRO Plugin: NO (00), YES (01)

4.4 GET PLUGIN: 04

TO ROTO: Y FROM ROTO: N

Command
5A 03 04 <CL:2 PH:8 PM> CL = Command data length, MSB followed by LSB = 0008 PH = Plugin hash

Response
A5 <RC PH:8 PN:0D PM>
RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values)
PH = Plugin hash
PN = Plugin name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
PM = MACRO Plugin: NO (00), YES (01)

4.5 SET PLUGIN: 05

TO ROTO: N FROM ROTO: Y

Note: This command is sent from ROTO-CONTROL when a PLUGIN (or no PLUGIN) has been selected.

Command
5A 03 05 <CL:2 PH:8>
CL = Command data length, MSB followed by LSB = 0008
PH = Plugin hash (all FFs if no PLUGIN is currently set)
Response
N/A

4.6 ADD PLUGIN: 06

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 05 <CL:2 PH:8 PN:0D>
CL = Command data length, MSB followed by LSB = 0015
PH = Plugin hash
PN = Plugin name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), PLUGIN EXISTS (FC), ERROR (all other values)

4.7 SET PLUGIN NAME: 07

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 07 <CL:2 PH:8 PN:0D>
CL = Command data length, MSB followed by LSB = 0015
PH = Plugin hash
PN = Plugin name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values)

4.8 CLEAR PLUGIN: 08

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 08 <CL:2 PH:8> CL = Command data length, MSB followed by LSB = 0008 PH = Plugin hash
Response
A5 <RC> RC = Response code: SUCCESS (00), NO PLUGIN (FD), ERROR (all other values)

4.9 GET PLUGIN KNOB CONFIG: 09

TO ROTO: Y FROM ROTO: N

Command
5A 03 09 <CL:2 PH:8 CI> CL = Command data length, MSB followed by LSB = 0009 PH = Plugin hash CI = Control index: 00 – 3F
Response
A5 <RC PH:8 CI MI:2 MH:6 MA MN:2 MX:2 CN:0D CS HM IP1 IP2 HS SN:10*0D> RC = Response code: SUCCESS (00), NO PLUGIN/CONTROL (FD), ERROR (all other values) PH = Plugin hash CI = Control index: 00 – 3F MI = Mapped param index MH = Mapped param hash MA = MACRO param: NO (00), YES (01) MN = Min value MX = Max value CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed CS = CS = Colour scheme: 00 - 52 HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01) IP1 = Indent position 1: 00 – 7F, FF if unused, only applies for KNOB_300 IP2 = Indent position 2: 00 – 7F, FF if unused, only applies for KNOB_300 HS = Haptic steps: 02 - 10, only applies for KNOB_N_STEP SN = An array of 10 x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed

4.10 GET PLUGIN SWITCH CONFIG: 0A

TO ROTO: Y FROM ROTO: N

Command
5A 03 0A <CL:2 PH:8 CI> CL = Command data length, MSB followed by LSB = 0009 PH = Plugin hash

CI = Control index: 00 – 3F
Response
A5 <RC PH:8 CI MI:2 MH:6 MN MX CN:0D CS LN LF HM HS SN:10*0D>
RC = Response code: SUCCESS (00), NO PLUGIN/CONTROL (FD), ERROR (all other values)
PH = Plugin hash
CI = Control index: 00 – 3F
MI = Mapped param index
MH = Mapped param hash
MN = Min value
MX = Max value
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = CS = Colour scheme: 00 - 52
LN = LED ON colour: 00 - 52
LF = LED OFF colour: 00 - 52
HM = Haptic mode: PUSH (00), TOGGLE (01)
HS = Haptic steps: 00 or 02 - 10
SN = An array of 10 x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed

4.11 SET PLUGIN KNOB CONFIG: 0B

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 0B <CL:2 PH:8 CI MI:2 MH:6 MN:2 MX:2 CN:0D CS HM IP1 IP2 HS SN:HS*0D>
CL = Command data length, MSB followed by LSB = 0027 + (HS * 0D)
PH = Plugin hash
CI = Control index: 00 – 3F
MI = Mapped param index
MH = Mapped param hash
MA = MACRO param: RFU, always set to 00
MN = Min value
MX = Max value
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = CS = Colour scheme: 00 - 52
HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01)
IP1 = Indent position 1: 00 – 7F, FF if unused, only applies for KNOB_300
IP2 = Indent position 2: 00 – 7F, FF if unused, only applies for KNOB_300
HS = Haptic steps: 02 - 10, only applies for KNOB_N_STEP
SN = An array of HS x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), NO PLUGIN/CONTROL (FD), ERROR (all other values)

4.12 SET PLUGIN SWITCH CONFIG: 0C

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 0C <CL:2 PH:8 CI MI:2 MH:6 MN MX CN:0D CS LN LF HM HS SN:HS*0D>
CL = Command data length, MSB followed by LSB = 0025 + (HS * 0D)
PH = Plugin hash
CI = Control index: 00 – 3F
MI = Mapped param index
MH = Mapped param hash
MN = Min value
MX = Max value
CN = Control name: 0D-byte NULL terminated ASCII string, padded with 00s if needed
CS = CS = Colour scheme: 00 - 52
LN = LED ON colour: 00 - 52
LF = LED OFF colour: 00 - 52
HM = Haptic mode: PUSH (00), TOGGLE (01)
HS = Haptic steps: 00 or 02 – 10, set to 00 if a normal two position switch with no haptic strings
SN = An array of HS x 0D-byte NULL terminated ASCII strings, each string padded with 00s if needed
Response
A5 <RC>
RC = Response code: SUCCESS (00), NO PLUGIN/CONTROL (FD), ERROR (all other values)

4.13 CLEAR PLUGIN CONTROL CONFIG: 0D

TO ROTO: Y FROM ROTO: N

Note: A config update session must be started using START CONFIG UPDATE for this command to be processed.

Command
5A 03 0D <CL:2 PH:8 CT CI>
CL = Command data length, MSB followed by LSB = 000A
PH = Plugin hash
CT = Control type: KNOB (00), SWITCH (01)
CI = Control index: 00 – 3F
Response
A5 <RC>
RC = Response code: SUCCESS (00), NO PLUGIN/CONTROL (FD), ERROR (all other values)

4.14 PLUGIN CONTROL LEARNED: 0E

TO ROTO: N FROM ROTO: Y

Note: This command is also sent from ROTO-CONTROL when a PLUGIN control is cleared.

Command
5A 03 0E <CL:2 PH:8 CT CI>
CL = Command data length, MSB followed by LSB = 000A
PH = Plugin hash

CT = Control type: KNOB (00), SWITCH (01) CI = Control index: 00 – 3F
Response
N/A