ROTO-CONTROL SERIAL API

Version 1.3

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	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
1.2	23/06/2025	MIDI and PLUGIN mode updates, including new MIDI switch control modes (PROGRAM CHANGE and NOTE), supporting different DAW types, and a new knob haptic mode: • GET KNOB CONTROL CONFIG • GET SWITCH CONTROL CONFIG • SET KNOB CONTROL CONFIG • SET SWITCH CONTROL CONFIG • GET CURRENT PLUGIN • GET FIRST PLUGIN • GET NEXT PLUGIN • GET PLUGIN • GET PLUGIN KNOB CONFIG • SET PLUGIN KNOB CONFIG Document formatting improvements.
1.3	26/11/2025	Added Appendix A: Colour Scheme Table.

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 • ROTO-SETUP v1.1.4
1.2 1.3	• ROTO-SETUP v2.0.0 • ROTO-SETUP v2.1.0

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

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

Description
Request the ROTO firmware version.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

2.2 GET MODE: 02

Command
5A **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.)

Description
Retrieve the current ROTO mode.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

2.3 SET MODE: 03

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)

Description
Set the current mode of ROTO.

Direction	
TO ROTO	Y
FROM ROTO	Y

2.4 START CONFIG UPDATE: 04

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)

Description
Start a configuration update session with ROTO.

Direction	
TO ROTO	Y
FROM ROTO	N

2.5 END CONFIG UPDATE: 05

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)

Description
End the config update session with ROTO. If the update was to the currently selected MIDI setup or PLUGIN, it will be updated on ROTO.

Direction	
TO ROTO	Y
FROM ROTO	N

2.6 FACTORY RESET: 06

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)
```

Description
Perform a factory reset of ROTO. Please note this command reformats the file system, all saved MIDI setups and PLUGIN configs will be erased.

Direction	
TO ROTO	Y
FROM ROTO	N

3. MIDI Commands: 02

3.1 GET CURRENT SETUP: 01

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

Description
Get the current MIDI setup selected by ROTO.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.2 GET SETUP: 02

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

Description
Retrieve the specified MIDI setup from ROTO.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.3 SET SETUP: 03

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)

Description

Used to select a MIDI setup in ROTO or sent from ROTO when a MIDI setup is selected via ROTO.

Direction	
TO ROTO	**Y**
FROM ROTO	**Y**

3.4 SET SETUP NAME: 04

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)

Description
Set the name of a MIDI setup. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.5 GET KNOB CONTROL CONFIG: 05

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 NA = Control Mode is NRPN: NRPN address All other Control Modes this is unused, set to 0000 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 (see Appendix A) HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01), KNOB_300_CENTRE_INDENT (02) 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

Description
Retrieve a PLUGIN knob control config.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.6 GET SWITCH CONTROL CONFIG: 06

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), PROGRAM CHANGE (04), NOTE (05) CC = Control channel: 01 - 10 CP = Control Mode is CC: Control param Control Mode is NRPN: Unused, set to FF Control Mode is PROGRAM CHANGE: Program number Control Mode is NOTE: Note value NA = Control Mode is NRPN: NRPN address Control Mode is PROGRAM CHANGE: Bank select if required, set to FFFF if not All other Control Modes this is unused, set to 0000 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 (see Appendix A) LN = LED ON colour: 00 - 52 (see Appendix A) LF = LED OFF colour: 00 - 52 (see Appendix A) 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

Description
Retrieve a MIDI setup switch control config.

Direction

TO ROTO	Y
FROM ROTO	N

3.7 SET KNOB CONTROL CONFIG: 07

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

NA = Control Mode is NRPN: NRPN address

All other Control Modes this is unused, set to 0000

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 (see Appendix A)

HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01), KNOB_300_CENTRE_INDENT (02)

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)

Description

Set a MIDI setup knob control config. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	Y
FROM ROTO	N

3.8 SET SWITCH CONTROL CONFIG: 08

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), PROGRAM CHANGE (04), NOTE (05)

CC = Control channel: 01 - 10

CP = Control Mode is CC: Control param

Control Mode is NRPN: Unused, set to FF

Control Mode is PROGRAM CHANGE: Program number

Control Mode is NOTE: Note value NA = Control Mode is NRPN: NRPN address Control Mode is PROGRAM CHANGE: Bank select if required, set to FFFF if not All other Control Modes this is unused, set to 0000 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 (see Appendix A) LN = LED ON colour: 00 - 52 (see Appendix A) LF = LED OFF colour: 00 - 52 (see Appendix A) 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)

Description
Set a MIDI setup switch control config. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.9 CLEAR CONTROL CONFIG: 09

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)

Description
Clear a configured MIDI setup control. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.10 CLEAR MIDI SETUP: 0A

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)

Description
Clear a MIDI setup, including all associated knob and switch configs. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

3.11 MIDI CONTROL LEARNED: 0B

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

Description
ROTO has learned or cleared a MIDI setup control.

Direction	
TO ROTO	**N**
FROM ROTO	**Y**

4. PLUGIN Commands: 03

4.1 GET CURRENT PLUGIN: 01

Command
5A **03 01** <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 PN:0D PT DT> 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 PT = Plugin type: Normal (00), MACRO Learned (01), MACRO Auto-config (02), MACRO Default (03), Third-party (04) DT = DAW Type: Ableton Live (01), Bitwig Studio (02)

Description
Get the current PLUGIN selected by ROTO, if any.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

4.2 GET FIRST PLUGIN: 02

Command
5A **03 02** <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 SN:0D PT> 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 PT = Plugin type: Normal (00), MACRO Learned (01), MACRO Auto-config (02), MACRO Default (03), Third-party (04)

Description
Get the first PLUGIN stored in ROTO, if any.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

4.3 GET NEXT PLUGIN: 03

Command
5A **03 03** <CL:2> CL = Command data length, MSB followed by LSB = 0000
Response
A5 <RC PH:8 PN:0D PT>

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
PT = Plugin type: Normal (00), MACRO Learned (01), MACRO Auto-config (02), MACRO Default (03), Third-party (04)

Description
Get the next PLUGIN stored in ROTO, if any.

Direction	
TO ROTO	Y
FROM ROTO	N

4.4 GET PLUGIN: 04

Command
5A **03 04** <CL:2 PH:8 PT> 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 PT = Plugin type: Normal (00), MACRO Learned (01), MACRO Auto-config (02), MACRO Default (03), Third-party (04)

Description
Retrieve the specified PLUGIN from ROTO.

Direction	
TO ROTO	Y
FROM ROTO	N

4.5 SET PLUGIN: 05

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

Description
A PLUGIN (or no PLUGIN) has been selected in ROTO.

Direction	
TO ROTO	N

FROM ROTO	Y

4.6 ADD PLUGIN: 06

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)

Description
Add an (empty) PLUGIN to ROTO. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	Y
FROM ROTO	N

4.7 SET PLUGIN NAME: 07

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)

Description
Set the name of a PLUGIN. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	Y
FROM ROTO	N

4.8 CLEAR PLUGIN: 08

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)

Description
Clear a PLUGIN, including all associated knob and switch configs. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	Y
FROM ROTO	N

4.9 GET PLUGIN KNOB CONFIG: 09

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 (see Appendix A) HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01), KNOB_300_TOP_INDENT (02) 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

Description
Retrieve a PLUGIN knob control config.

Direction	
TO ROTO	Y
FROM ROTO	N

4.10 GET PLUGIN SWITCH CONFIG: 0A

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 (see Appendix A) LN = LED ON colour: 00 - 52 (see Appendix A) LF = LED OFF colour: 00 - 52 (see Appendix A) 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

Description
Retrieve a PLUGIN switch control config.

Direction	
TO ROTO	Y
FROM ROTO	N

4.11 SET PLUGIN KNOB CONFIG: 0B

Command
5A **03 0B** <CL:2 PH:8 CI MI:2 MH:6 MA 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: 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 (see Appendix A) HM = Haptic mode: KNOB_300 (00), KNOB_N_STEP (01), KNOB_300_TOP_INDENT (02) 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)

Description

Set a PLUGIN knob control config. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

4.12 SET PLUGIN SWITCH CONFIG: 0C

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 (see Appendix A) LN = LED ON colour: 00 - 52 (see Appendix A) LF = LED OFF colour: 00 - 52 (see Appendix A) 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)

Description
Set a PLUGIN switch control config. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	**Y**
FROM ROTO	**N**

4.13 CLEAR PLUGIN CONTROL CONFIG: 0D

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)

Description
Clear a configured PLUGIN control. Note a config update session must be started using START CONFIG UPDATE for this command to be processed.

Direction	
TO ROTO	Y
FROM ROTO	N

4.14 PLUGIN CONTROL LEARNED: 0E

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

Description
ROTO has learned or cleared a PLUGIN control.

Direction	
TO ROTO	N
FROM ROTO	Y

Appendix A: Colour Scheme Table

The Colour Scheme table is specified below, with the background colour in RGB and the text colour as black or white. Note the text colour does not apply when specifying a LED colour scheme.

Index (decimal)	Background Colour			Text Colour
	Red	Green	Blue	
0	FF	94	A0	Black
1	FF	A5	27	Black
2	D3	99	25	Black
3	FF	F4	77	Black
4	C6	FB	00	Black
5	1B	FF	2D	Black
6	26	FF	A2	Black
7	5F	FF	DF	Black
8	90	C5	F6	Black
9	57	80	DB	Black
10	97	A7	F6	Black
11	E0	6C	DB	Black
12	ED	53	9A	Black
13	FF	FF	F6	Black
14	FF	36	34	Black
15	FF	6C	02	Black
16	9E	62	48	Black
17	FF	F0	32	Black
18	8C	FF	63	Black
19	3F	C3	00	Black
20	00	BF	A8	Black
21	19	E9	F6	Black
22	10	A4	E5	Black
23	00	7D	B9	Black
24	8D	6C	DB	Black
25	BD	77	BF	Black
26	FF	39	CC	Black
27	D8	D0	C8	Black
28	EC	68	56	Black
29	FF	A3	6F	Black
30	DB	AD	6D	Black
31	F6	FF	A7	Black
32	DA	E4	92	Black
33	C1	D0	6F	Black
34	A1	C4	88	Black
35	DC	FD	D9	Black
36	D4	F1	EF	Black
37	C0	C1	DA	Black
38	D4	BB	DB	Black

39	B4	98	DC	Black
40	ED	DC	D9	Black
41	AF	A9	A3	Black
42	EE	92	86	Black
43	BE	82	52	Black
44	9E	83	66	Black
45	C6	BA	65	Black
46	AC	BE	00	Black
47	81	BO	4A	Black
48	8D	C2	B3	Black
49	A1	B3	BD	Black
50	8A	A5	BF	Black
51	88	93	C4	Black
52	AB	95	AE	Black
53	C6	9F	B7	Black
54	C3	71	90	Black
55	7F	7B	76	Black
56	B5	33	31	White
57	AF	51	2F	White
58	76	4F	3E	White
59	E3	C3	00	White
60	8A	96	1D	White
61	56	9F	2F	White
62	0A	9C	88	White
63	24	63	7F	White
64	1B	2F	90	White
65	30	52	9C	White
66	65	4B	A6	White
67	A9	4B	A6	White
68	D3	2E	6A	White
69	3E	3C	39	White
70	10	10	10	White
71	FF	00	00	White
72	00	FF	00	Black
73	FF	FF	00	Black
74	00	00	F6	White
75	FF	00	F6	White
76	00	FF	F6	Black
77	84	00	00	White
78	84	80	00	Black
79	00	80	00	White
80	00	80	7B	Black
81	00	00	7B	White
82	84	00	7B	White