

# PKONE System Commands

Last Updated: 18 June 2021 4:15:00 PM

Notes: All commands and responses will be terminated with the character “E”. Commands from the host to the Nano will be ASCII-encoded.

## Detect Connected Nano

Description	Returns information about the connected Nano (including firmware and hardware rev)
Opcode	PCN
Payload	None
Example:	PCNE – return information about the connected Nano controller

Response:

Opcode	PCN
Payload	<ol style="list-style-type: none"><li>“F” + firmware version number (last digit is the minor version) (ex: F10 is firmware v1.0, F114 is firmware v11.4)</li><li>“H” + hardware revision number (ex: H1 is hardware rev 1, H12 is hardware version 12)</li></ol>
Examples:	<ul style="list-style-type: none"><li>PCNF11H2E – The connected Nano controller board has firmware v1.1, hardware rev 2</li><li>PCNF112H10E – The connected Nano controller board has firmware v11.2, hardware rev 10</li></ul>

## Detect Connected Boards

Description	Returns connected board at specified Address ID (including firmware and hardware rev)
Opcode	PCB
Payload	<ol style="list-style-type: none"><li>Address ID (0-7) (1 character)</li></ol>
Example:	PCB1E – return the board connected at address ID 1

Response:

Opcode	PCB (or no response when there is no board connected with the specified address ID)
Payload	<ol style="list-style-type: none"><li>Address ID (0-7) (1 character)</li><li>Board Type (X = Extension, L = Lightshow) (1 character)</li><li>“F” + firmware version number (last digit is the minor version) (ex: F10 is firmware v1.0, F114 is firmware v11.4) (3+ characters) – not present if no board connected</li><li>“H” + hardware revision number (ex: H1 is hardware rev 1, H12 is hardware version 12) (2+ characters) – not present if no board connected</li><li>“P” + (Y = 48 volts connected, N = 48 volts not present) (this is optional and only applies to Extension board) (2 characters, optional)</li><li>“RGB” = Lightshow running RGB firmware, “RGBW” = Lightshow running RGBW firmware (this is optional and only applies to Lightshow boards) (3 or 4 characters)</li></ol>
Examples:	<ul style="list-style-type: none"><li>No response – No board is connected at the specified address ID (when issuing the PCB command, it is suggested the host set a short timeout value and if no response is received before the timeout elapses assume there is no board connected)</li><li>PCB0LF11H2RGBE – A Lightshow board with RGB firmware v1.1, hardware rev 2 is connected with an address ID of 0</li></ul>

	<ul style="list-style-type: none"> <li>• PCB1LF11H2RGBWE – A Lightshow board with RGBW firmware v1.1, hardware rev 2 is connected with an address ID of 1</li> <li>• PCB2XF12H1PYE – An Extension board with firmware 1.2, hardware rev 1, and 48 volt power on is connected as address ID 2</li> </ul>
--	---

NOTE: Lightshow boards only have address ID values from 0 to 3.

### Watchdog Start

Description	Instructs the controller to start the watchdog process using the specified timeout value. The controller will expect to receive a watchdog signal command (PWD) before the watchdog timer elapses (the timer is reset every time a PWD command is received). When the watchdog timer elapses without receiving a PWD command, the controller will disable all coils, servos, and lights. This is a safety feature in case of a communications loss (software crash/freeze, other host system issue).
Opcode	PWS
Payload	1. Watchdog timeout (ms) (0000-9999) (4 characters) – the number of milliseconds to set the watchdog timeout. A value of 0000 will stop the watchdog process (same as issuing a PWF command).
Example:	PWS1000E – instruct the controller to start the watchdog process with a timeout of 1000 ms (1 second)

Response:

Opcode	PWS
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWSE – The connected Nano controller board has successfully started the watchdog process</li> </ul>

### Watchdog Stop

Description	Instructs the controller to stop the watchdog process.
Opcode	PWF
Payload	None
Example:	PWFE – instruct the controller to stop the watchdog process

Response:

Opcode	PWF
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWFE – The connected Nano controller board has successfully stopped the watchdog process</li> </ul>

### Watchdog Signal/Ping

Description	Notifies the controller to restart the watchdog process timeout timer (the connected computer is active and still able to communicate with the controller). This must be sent to the Nano at least every second once the watchdog process has been started or a watchdog timeout error will occur and all coils, servos, and lights will be disabled. It is recommended
-------------	---

	that this be send about every 500ms to ensure communication delays do not cause a watchdog timeout.
Opcode	PWD
Payload	None
Example:	PWDE – notify the controller to restart the watchdog timeout timer

Response:

Opcode	PWD
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWDE – The connected Nano controller board has successfully reset the watchdog process timeout timer</li> </ul>

### Watchdog Timeout

Description	<p>A watchdog process timeout has occurred on the controller and the watchdog process has been stopped.</p> <p><b>NOTE:</b> This message is only sent from the controller to the host.</p>
-------------	--

Response:

Opcode	PWT
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWTE – A watchdog process timeout has occurred and the watchdog process has been stopped on the controller. All coils, servos, and lights have been disabled.</li> </ul>

### Initialize/Reset System

Description	Reset and initialize the hardware (all coils, servos, simple LEDs, WS281X LEDs, and the watchdog process should be stopped or disabled and reset back to their initial states).
Opcode	PRS
Payload	None
Example:	PRSE – instruct the controller to reset and reinitialize all the hardware

Response:

Opcode	PRS (One response is sent from the Nano controller and for each board connected in the PKONE system).
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character) (not sent when the board is the Nano)</li> <li>2. Board Type (N = Nano, X = Extension, L = Lightshow) (1 character)</li> </ol>
Example:	<ul style="list-style-type: none"> <li>• PRSNEPRS0XEPRS1LE – The connected boards (Extension at address ID 0, Lightshow at address ID 1, and the Nano controller) have all been successfully reset and initialized.</li> <li>• PRSNE - The connected boards (only a Nano controller) have all been successfully reset and initialized.</li> <li>• PRSNEPRS0XEPRS1XEPRS2LE – The connected boards (Extension at address ID 0, Extension at address ID 1, Lightshow at address ID 2, and the Nano controller) have all been successfully reset and initialized</li> </ul>

### Current Switch State

Description	Retrieve the current state of all the switches (and optos) for the Extension board at the specified Address ID. This is typically utilized right after initialization to determine the locations of balls in the pinball machine.
Opcode	PSA
Payload	1. Address ID (0-7) (1 character)
Example:	PSA1E – Request the status of all switches and optos for the Extension board at Address ID 1

Response:

Opcode	PSA
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. One character corresponding to each switch on the board (0 = open/inactive, 1 = closed/active) in order of switch number (35 characters [30 switches + 5 optos on the current Extension board])</li> </ol>
Example:	<ul style="list-style-type: none"> <li>• PSA200001001000000000000000000000000000000000000000E – The states of all switches and optos on the board at address ID 2 (35 status characters). In this example the 5<sup>th</sup> and 8<sup>th</sup> switches are active/closed while the rest are inactive/open.</li> <li>• PSA101100E – The states of all switches and optos on the Extension board at address ID 1. In this example the 2<sup>nd</sup> and 3<sup>rd</sup> switches on board with ID 1 are active/closed while the rest are inactive/open.</li> </ul>

### Switch State Change Notification

Description	Whenever a switch changes states (from active/closed to inactive/open or from inactive/open to active/closed), a switch state change notification message is sent from the controller. These messages are only sent upon a change in state. No message will be sent while a switch continues to be in the same state. <b>NOTE:</b> This message is only sent from the controller to the host.
-------------	--

Response:

Opcode	PSW
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Switch/opto number (01-35) (2 characters)</li> <li>3. Switch state (1 = active/closed, 0 = inactive/open) (1 character)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PSW2171E – Switch 17 on board with address id 2 has just become active/closed.</li> <li>• PSW1020E – Switch 2 on board with address id 1 has just become inactive/open.</li> </ul>

### Coil Configuration

Description	Configures the specified coil settings (which will be stored for the coil)
Opcode	PCC
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> <li>3. Pulse time (ms) (000-250) (3 characters)</li> <li>4. Pulse power (%) (00-99) (2 characters)</li> <li>5. Hold power (%) (00-99) (2 characters)</li> <li>6. Recycle time (ms) (000-500) (3 characters)</li> </ol>

Examples:	<ul style="list-style-type: none"> <li>● PCC1040759900150E – Set the following settings for coil 4 on board at address id 1: <ul style="list-style-type: none"> <li>○ Pulse time: 75ms</li> <li>○ Pulse power: 99%</li> <li>○ Hold power: 0%</li> <li>○ Recycle time: 150ms</li> </ul> </li> <li>● PCC2101009950200E – Set the following settings for coil 10 on board at address id 2: <ul style="list-style-type: none"> <li>○ Pulse time: 100ms</li> <li>○ Pulse power: 99%</li> <li>○ Hold power: 50%</li> <li>○ Recycle time: 200ms</li> </ul> </li> </ul>
-----------	---

Response:

Opcode	PCC
Payload	None
Example:	<ul style="list-style-type: none"> <li>● PCCE – The specified coil has been configured successfully</li> </ul>

Definitions:

- **Pulse time:** The default amount of time, in milliseconds, that this coil will pulse for when it is activated.
- **Pulse power:** The power percentage which controls how much power is applied during the initial pulse phase of the coil’s activation.
- **Hold power:** This setting lets you control how much power is sent to the coil when it is “held” in the on position (using the Enable/Hold Coil command PCH).
- **Recycle time:** Controls the amount of time, in milliseconds, before the coil can be fired again (commands during the recycle time period after the coil is released are ignored). This is used on things like pop bumpers and slingshots to prevent “machine gunning.”

### Pulse Coil

Description	Pulses the specified coil using the coil’s current configuration settings. <b>NOTE:</b> The PCC command must be sent to set the coil’s configuration settings prior to sending this command.
Opcode	PCP
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>● PCP104E – Pulse coil 4 on board at address id 1 using the coil’s current configuration settings</li> <li>● PCP210E – Pulse coil 10 on board at address id 2 using the coil’s current configuration settings</li> </ul>

Response:

Opcode	PCP
Payload	None
Example:	<ul style="list-style-type: none"> <li>● PCPE – The specified coil has been pulsed successfully</li> </ul>

### Enable/Hold Coil

Description	Pulses the specified coil using the coil's current configuration settings. The coil is initially activated using the pulse power and pulse time configuration settings and then is held using the hold power setting. If the pulse time configuration setting is 0, the coil will simply be activated and held using the hold power configuration setting. <b>NOTE:</b> The PCC command must be sent to set the coil's configuration settings prior to sending this command.
Opcode	PCH
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PCH104E – Enable and hold coil 4 on board at address id 1 using the coil's current configuration settings</li> <li>• PCH210E – Enable and hold coil 10 on board at address id 2 using the coil's current configuration settings</li> </ul>

Response:

Opcode	PCH
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PCHE – The specified coil has been enabled/held successfully</li> </ul>

### Disable/Release Coil

Description	Disables/released the specified coil <b>NOTE:</b> This command will have no effect if the specified coil is not currently enabled (in hold state).
Opcode	PCR
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PCR104E – Disable/release coil 4 on board at address id 1</li> <li>• PCR210E – Disable/release coil 10 on board at address id 2</li> </ul>

Response:

Opcode	PCR
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PCRE – The specified coil has been disabled/released successfully</li> </ul>

### Hardware Rule Configuration

Description	Configures a hardware rule for the specified coil (this will be stored for the coil). Hardware rules allow for low latency responses to switch inputs without the need for external software. There can only be one rule per solenoid. A new rule will always overwrite an existing rule for the coil.
Opcode	PHR
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> <li>3. Mode (1-5, see Rule Modes below) (1 character)</li> <li>4. Switch 1 (00 = ignored, 01-35) (2 characters)</li> <li>5. Switch 1 inverted (0 = false, 1 = true) (1 character) – when true, inverts the state of the switch for the purposes of this rule</li> </ol>

	<ol style="list-style-type: none"> <li>6. Switch 2 (00 = ignored, 01-35) (2 characters)</li> <li>7. Switch 2 inverted (0 = false, 1 = true) (1 character) – when true, inverts the state of the switch for the purposes of this rule</li> <li>8. Coil delay time (ms) (000-999) (3 characters) – the number of milliseconds to delay before activating the coil when the rule is activated by a switch</li> <li>9. Coil pulse time (ms) (000-250) (3 characters)</li> <li>10. Coil pulse power (%) (00-99) (2 characters)</li> <li>11. Coil hold power (%) (00-99) (2 characters)</li> <li>12. Coil recycle time (ms) (000-500) (3 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PHR10130700000001009930000E – Set the following hardware rule for coil 1 on board at address id 1: <ul style="list-style-type: none"> <li>○ Mode 3 (pulse on switch activation and release)</li> <li>○ Switch 1: switch 7 (inverted = false)</li> <li>○ Switch 2: ignored in mode 3 (set to 0, inverted = false)</li> <li>○ Coil delay time: ignored in mode 3 (set to 0)</li> <li>○ Pulse time: 100ms</li> <li>○ Pulse power: 99%</li> <li>○ Hold power: 30%</li> <li>○ Recycle time: 0ms</li> </ul> <p>This means coil 1 will be pulsed for 100ms at full power when switch 7 is activated (flipper button) then enabled/held at 30% power while the switch remains activated. The coil will be disabled/released when switch 7 is deactivated. This is a typical rule for a main flipper coil with no EOS switch.</p> </li> <li>• PHR10150700800001009930000E – Set the following hardware rule for coil 1 on board at address id 1: <ul style="list-style-type: none"> <li>○ Mode 5 (Pulse and enable/hold on switch activation and release on switch release and release on second switch)</li> <li>○ Switch 1: switch 7 (inverted = false)</li> <li>○ Switch 2: switch 8 (inverted = false)</li> <li>○ Coil delay time: ignored in mode 5 (set to 0)</li> <li>○ Pulse time: 100ms</li> <li>○ Pulse power: 99%</li> <li>○ Hold power: 30%</li> <li>○ Recycle time: 0ms</li> </ul> <p>This means coil 1 will be pulsed for 100ms at full power when switch 7 is activated (flipper button) then enabled/held at 30% power while the switch remains activated. The coil will be disabled/released when switch 7 is deactivated. The pulse will be canceled, and the coil will be disabled/released when switch 8 is activated. This is a typical rule for a main flipper coil with an EOS switch (switch 8 in this example).</p> </li> <li>• PHR21012400000000459900090E – Set the following hardware rule for coil 10 on board at address id 2: <ul style="list-style-type: none"> <li>○ Mode 1 (Pulse on switch activation)</li> <li>○ Switch 1: switch 24 (inverted = false)</li> <li>○ Switch 2: ignored in mode 1 (set to 0, inverted = false)</li> <li>○ Pulse time: 45ms</li> <li>○ Pulse power: 99%</li> <li>○ Hold power: 0%</li> <li>○ Recycle time: 90ms</li> </ul> </li> </ul>

	This means coil 10 will be pulsed for 45ms at full power when switch 24 is activated. Coil 10 cannot be reactivated for 90ms due to the recycle time setting. This is a typical rule for a pop bumper.
--	--

Response:

Opcode	PHR
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PHRE – The hardware rule for the specified coil has been configured successfully</li> </ul>

Hardware Rule Modes:

1. Pulse on switch activation – Pulsed the coil when the switch is activated. The pulse continues after the switch is released. Typically used for autofire coils such as pop bumpers. Switch 2 is ignored in this mode. Coil delay time is ignored in this mode.
2. Delayed pulse on switch activation – Pulsed the coil after the specified delay time elapses when the switch is activated. The pulse continues after the switch is released. Switch 2 is ignored in this mode.
3. Pulse on switch activation and release – Pulsed the coil when the switch is activated. When the switch is released, the pulse is canceled. Switch 2 is ignored in this mode. Coil delay time is ignored in this mode. Typically used on the main coil for dual coil flippers without eos switch.
4. Pulse and enable/hold on switch activation and release on switch release – Pulsed the coil when the switch is activated then enables/holds the coil. When the switch is released, the pulse is canceled, and the coil is disabled/released. Switch 2 is ignored in this mode. Coil delay time is ignored in this mode. Typically used for single coil flippers.
5. Pulse and enable/hold on switch activation and release on switch release and release on second switch – Pulsed the coil when the switch is activated then enables/holds the coil. When the switch is released, the pulse is canceled, and the coil is disabled/released. When switch 2 is activated, the pulse is canceled, and the coil is disabled/released. Coil delay time is ignored in this mode. Typically used on the main coil for dual-wound coil flippers with eos switch.

See the Coil Configuration (PCC) command for more information on coil settings.

Additional Notes:

- There are separate coil settings for the normal coil configuration and the hardware rule settings. It is important to note that changing the coil configuration will not affect any hardware rule associated with that coil. To change the settings for the hardware rule, you must set the rule again with the new settings (it will overwrite the old rule). Likewise, establishing a hardware rule does not affect the existing coil settings.

### Delete Hardware Rule

Description	Deletes the hardware rule for the specified coil
Opcode	PHD
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Coil number (01-10) (2 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PHD104E – Deletes the hardware rule for coil 4 on board at address id 1</li> <li>• PHD210E – Deletes the hardware rule for coil 10 on board at address id 2</li> </ul>

Response:

Opcode	PHD
--------	-----



Payload	None
Example:	<ul style="list-style-type: none"> <li>• PHDE – The hardware rule for the specified coil has been deleted successfully</li> </ul>

### Servo Control

Description	Controls the specified servo motor
Opcode	PSC
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-7) (1 character)</li> <li>2. Servo number (11-14) (2 characters)</li> <li>3. Position/degrees to move (03-27) (2 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PSC11415E – Move servo 14 on board at address id 1 to the middle position (90 degrees)</li> <li>• PSC21103E – Move servo 11 on board at address id 2 to 0 degrees</li> </ul>

Response:

Opcode	PSC
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PSCE – The servo command has been processed successfully</li> </ul>

### Standard LED Control

Description	Controls the specified standard LED
Opcode	PLS
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. LED number (01-40) (2 characters)</li> <li>3. On/Off (0 = Off, 1 = On) (1 character)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PLS1091E – Turns on standard LED 9 on board at address id 1</li> <li>• PLS1341E – Turns on standard LED 34 on board at address id 1</li> <li>• PLS2100E – Turns off standard LED 10 on board at address id 2</li> </ul>

Response:

Opcode	PLS
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PLSE – The specified standard LED command has been processed successfully</li> </ul>

### Control All Standard LEDs

Description	Controls all standard LEDs on the board with the specified Address ID
Opcode	PLO
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. On/Off (0 = Off, 1 = On) (1 character)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PLO11E – Turns on all standard LEDs on board at address id 1</li> <li>• PLO01E – Turns on all standard LEDs on board at address id 0</li> <li>• PLO20E – Turns off all standard LEDs on board at address id 2</li> </ul>

Response:

Opcode	PLO
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PLOE – The Control All Standard LEDs command has been processed successfully</li> </ul>

## RGB LED

### WS281X RGB LED Control (Lightshow RGB firmware only)

Description	Controls the specified WS281X RGB LED (may only be used with Lightshow boards running RGB firmware).
Opcode	PLW
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. LED number (01-64) (2 characters)</li> <li>4. Animation (0 – Off, 1 – On, 2 – Blink, 3 – Fade) (1 character)</li> <li>5. First RED color (000-255) (3 characters)</li> <li>6. First GREEN color (000-255) (3 characters)</li> <li>7. First BLUE color (000-255) (3 characters)</li> <li>8. Second RED color (000-255) (3 characters)</li> <li>9. Second GREEN color (000-255) (3 characters)</li> <li>10. Second BLUE color (000-255) (3 characters)</li> <li>11. Animation time in 10ms (0000-4096) (4 characters) (used for Blink and Fade animations)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PLW1109325500000000000002551000E – Fades WS281X RGB LED 9 in Group 1 from red (255,0,0) to blue (0,0,255) over 10 seconds on board at address id 1</li> <li>• PLW1234100025500000000000000000E – Turns on WS281X RGB LED 34 in Group 2 to green (0,255,0) on board at address id 1 (second color and fade time are ignored when animation is On)</li> <li>• PLW2810000000000000000000000000E – Turns off WS281X RGB LED 10 in Group 8 on board at address id 2 (first and second color and time are all ignored when animation is Off)</li> </ul>

#### Response:

Opcode	PLW
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PLWE – The specified WS281X RGB LED command has been processed successfully</li> </ul>

### WS281X RGB LED Simple Control (Lightshow RGB firmware only)

Description	Sets the color and optional fade time for the specified WS281X (may only be used with Lightshow boards running RGB firmware). The LED will fade from the current color to the specified color over the specified fade time interval (0 is set the color immediately with no fade).
Opcode	PLC
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. LED number (01-64) (2 characters)</li> <li>4. RED color (000-255) (3 characters)</li> </ol>

	<ul style="list-style-type: none"> <li>5. GREEN color (000-255) (3 characters)</li> <li>6. BLUE color (000-255) (3 characters)</li> <li>7. Fade time in 10ms (0000-4096) – 0 to 40960ms, 0 to 40.96 seconds (4 characters)</li> </ul>
Examples:	<ul style="list-style-type: none"> <li>• PLC11422550000000050E – Fades group 1 RGB LED 42 to red (255, 0, 0) over 500ms on board at address id 1</li> <li>• PLC02030002552550100E – Fades group 2 RGB LED 3 to cyan (0, 255, 255) over 1 second on board at address id 0</li> <li>• PLC11012552552550000E – Fades group 1 RGB LED 1 to white (255, 255, 255) instantly on board at address id 1</li> </ul>

Response:

Opcode	PLC
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PLCE – The specified WS281X RGB LED simple control command has been processed successfully</li> </ul>

### WS281X RGB LED Batch Control (Lightshow RGB firmware only)

Description	Sets the colors and optional fade time for a consecutive group of WS281X RGB LEDs (may only be used with Lightshow boards running RGB firmware). The LEDs will fade from the current color to the specified color over the specified fade time interval (0 is set the color immediately with no fade). All LEDs in the batch will use the same fade time value, but independent color values.
Opcode	PLB
Payload	<ul style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. First LED number (01-64) (2 characters)</li> <li>4. LED count (01-64) (2 characters)</li> <li>5. Fade time in 10ms (0000-4096) – 0 to 40960ms, 0 to 40.96 seconds (4 characters)</li> <li>6. RED color (000-255) for first LED (3 characters)</li> <li>7. GREEN color (000-255) for first LED (3 characters)</li> <li>8. BLUE color (000-255) for first LED (3 characters)</li> <li>9. Add RGB values for each additional LED in the batch group (9 characters for each additional LED, up to the number specified in the LED count value)</li> </ul>
Examples:	<ul style="list-style-type: none"> <li>• PLB1142030050128043226000255255255000000E – Fades group 1 RGB LED 42 to blue violet (138, 43, 226), RGB LED 43 to cyan (0, 255, 255), and RGB LED 44 to red (255, 0, 0) over 500ms on board at address id 1</li> <li>• PWB0203020100000255255255000255E – Fades group 2 RGB LED 3 to cyan (0, 255, 255) and RGB LED 4 to magenta (255, 0, 255) over 1 second on board at address id 0</li> <li>• PWB1101040000255255255255255000050255050147112219E – Sets group 1 RGB LED 1 to white (255, 255, 255), RGB LED 2 to yellow (255, 255, 0), RGB LED 3 to lime green (50, 255, 50), and RGB LED 4 to medium purple (147, 112, 219) instantly on board at address id 1</li> </ul>

Response:

Opcode	PLB
Payload	None

Example:	<ul style="list-style-type: none"> <li>PLBE – The specified WS281X RGB LED batch control command has been processed successfully</li> </ul>
----------	---

### Set All WS281X RGB LEDs (Lightshow RGB firmware only)

Description	Sets the color for all WS281X LEDs in the specified group number on the Lightshow board at the specified Address ID (may only be used with Lightshow boards running RGB firmware).
Opcode	PLA
Payload	<ol style="list-style-type: none"> <li>Address ID (0-3) (1 character)</li> <li>Group number (1-8) (1 character)</li> <li>RED color (000-255) (3 characters)</li> <li>GREEN color (000-255) (3 characters)</li> <li>BLUE color (000-255) (3 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>PLA11255255255E – Sets all RGB LEDs in group 1 to white (255, 255, 255) on board at address id 1</li> <li>PLA03000255255E – Sets all RGB LEDs in group 3 to cyan (0, 255, 255) on board at address id 0</li> <li>PLA28000000000E – Sets all RGB LEDs in group 8 to black/off (0, 0, 0) on board at address id 2</li> </ul>

Response:

Opcode	PLA
Payload	None
Example:	<ul style="list-style-type: none"> <li>PLAE – The Set All WS281X RGB LEDs command has been processed successfully</li> </ul>

## RGBW ONLY

### WS281X RGBW LED Control (Lightshow RGBW firmware only)

Description	Controls the specified WS281X RGBW LED (may only be used with Lightshow boards running RGBW firmware)
Opcode	PWW
Payload	<ol style="list-style-type: none"> <li>Address ID (0-3) (1 character)</li> <li>Group number (1-8) (1 character)</li> <li>LED number (01-64) (2 characters)</li> <li>Animation (0 – Off, 1 – On, 2 – Blink, 3 – Fade) (1 character)</li> <li>First RED color (000-255) (3 characters)</li> <li>First GREEN color (000-255) (3 characters)</li> <li>First BLUE color (000-255) (3 characters)</li> <li>First WHITE color (000-255) (3 characters)</li> <li>Second RED color (000-255) (3 characters)</li> <li>Second GREEN color (000-255) (3 characters)</li> <li>Second BLUE color (000-255) (3 characters)</li> <li>Second WHITE color (000-255) (3 characters)</li> <li>Animation time in 10ms (0000-4096) (4 characters) (used for Blink and Fade animations)</li> </ol>

Examples:	<ul style="list-style-type: none"> <li>• PWW1109325500000000000000002550001000E – Fades WS281X RGBW LED 9 in Group 1 from red (255,0,0,0) to blue (0,0,255,0) over 10 seconds on board at address id 1</li> <li>• PWW12341255255255255000000000000000E – Turns on WS281X RGBW LED 34 in Group 2 to white (255,255,255,255) on board at address id 1 (second color and fade time are ignored when animation is On)</li> <li>• PWW28100000000000000000000000000000E – Turns off WS281X RGBW LED 10 in Group 8 on board at address id 2 (first and second color and time are all ignored when animation is Off)</li> </ul>
-----------	---

Response:

Opcode	PWW
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PLWE – The specified WS281X RGB LED command has been processed successfully</li> </ul>

### WS281X RGBW LED Simple Control (Lightshow RGBW firmware only)

Description	Sets the color and optional fade time for the specified WS281X RGBW LED (may only be used with Lightshow boards running RGBW firmware). The LED will fade from the current color to the specified color over the specified fade time interval (0 is set the color immediately with no fade).
Opcode	PWC
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. LED number (01-64) (2 characters)</li> <li>4. RED color (000-255) (3 characters)</li> <li>5. GREEN color (000-255) (3 characters)</li> <li>6. BLUE color (000-255) (3 characters)</li> <li>7. WHITE color (000-255) (3 characters)</li> <li>8. Fade time in 10ms (0000-4096) – 0 to 40960ms, 0 to 40.96 seconds (4 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PWC11421380432260430050E – Fades group 1 RGBW LED 42 to blue violet (138, 43, 226, 43) over 500ms on board at address id 1</li> <li>• PWC02030002552550000100E – Fades group 2 RGBW LED 3 to cyan (0, 255, 255, 0) over 1 second on board at address id 0</li> <li>• PWC11012552552552550000E – Fades group 1 RGBW LED 1 to white (255, 255, 255, 255) instantly on board at address id 1</li> </ul>

Response:

Opcode	PWC
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWCE – The specified WS281X RGBW LED simple control command has been processed successfully</li> </ul>

### WS281X RGBW LED Batch Control (Lightshow RGBW firmware only)

Description	Sets the colors and optional fade time for a consecutive group of WS281X RGBW LEDs (may only be used with Lightshow boards running RGBW firmware). The LEDs will fade
-------------	---

	from the current color to the specified color over the specified fade time interval (0 is set the color immediately with no fade). All LEDs in the batch will use the same fade time value.
Opcode	PWB
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. First LED number (01-64) (2 characters)</li> <li>4. LED count (01-64) (2 characters)</li> <li>5. Fade time in 10ms (0000-4096) – 0 to 40960ms, 0 to 40.96 seconds (4 characters)</li> <li>6. RED color (000-255) for first LED (3 characters)</li> <li>7. GREEN color (000-255) for first LED (3 characters)</li> <li>8. BLUE color (000-255) for first LED (3 characters)</li> <li>9. WHITE color (000-255) for first LED (3 characters)</li> <li>10. Add RGBW values for each additional LED in the batch group (12 characters for each additional LED, up to the number specified in the LED count value)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PWB114203005013804322604300025525500025500000000E – Fades group 1 RGBW LED 42 to blue violet (138, 43, 226, 43), RGBW LED 43 to cyan (0, 255, 255, 0), and RGBW LED 44 to red (255, 0, 0, 0) over 500ms on board at address id 1</li> <li>• PWB0203020100000255255000255000255000E – Fades group 2 RGBW LED 3 to cyan (0, 255, 255, 0) and RGBW LED 4 to magenta (255, 0, 255, 0) over 1 second on board at address id 0</li> <li>• PWB1101040000255255255255255255000000050255050050147112219112E – Sets group 1 RGBW LED 1 to white (255, 255, 255, 255), RGBW LED 2 to yellow (255, 255, 0, 0), RGBW LED 3 to lime green (50, 255, 50, 50), and RGBW LED 4 to medium purple (147, 112, 219, 112) instantly on board at address id 1</li> </ul>

**Response:**

Opcode	PWB
Payload	None
Example:	<ul style="list-style-type: none"> <li>• PWBE – The specified WS281X RGBW LED batch control command has been processed successfully</li> </ul>

**Set All WS281X RGBW LEDs (Lightshow RGBW firmware only)**

Description	Sets the color for all WS281X RGBW LEDs in the specified group number on the Lightshow board at the specified Address ID (may only be used with Lightshow boards running RGBW firmware).
Opcode	PWA
Payload	<ol style="list-style-type: none"> <li>1. Address ID (0-3) (1 character)</li> <li>2. Group number (1-8) (1 character)</li> <li>3. RED color (000-255) (3 characters)</li> <li>4. GREEN color (000-255) (3 characters)</li> <li>5. BLUE color (000-255) (3 characters)</li> <li>6. WHITE color (000-255) (3 characters)</li> </ol>
Examples:	<ul style="list-style-type: none"> <li>• PWA11255255255255E – Sets all RGBW LEDs in group 1 to white (255, 255, 255, 255) on board at address id 1</li> <li>• PWA03000255255000E – Sets all RGBW LEDs in group 3 to cyan (0, 255, 255, 0) on board at address id 0</li> </ul>

	<ul style="list-style-type: none"> <li>PWA28000000000000E – Sets all RGBW LEDs in group 8 to black/off (0, 0, 0, 255) on board at address id 2</li> </ul>
--	---

Response:

Opcode	PWA
Payload	None
Example:	<ul style="list-style-type: none"> <li>PWAE – The Set All WS281X RGBW LEDs command has been processed successfully</li> </ul>

## Error

Description	<p>An error has occurred in the PKONE system.</p> <p><b>NOTE:</b> This message is only sent from the controller to the host.</p>
-------------	--

Response:

Opcode	PXX
Payload	1. Error code (0000-9999) (4 characters)
Example:	<ul style="list-style-type: none"> <li>PXX1000E – An error with code 1000 (unknown command opcode) has occurred.</li> </ul>

Error codes:

Code	Description
1000	Unknown command opcode
1001	Invalid command arguments