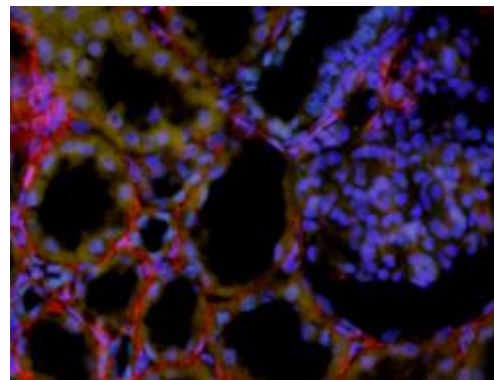
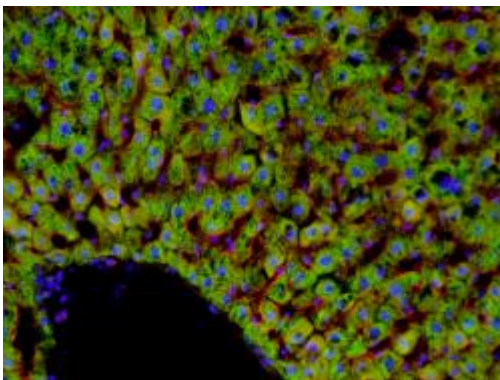
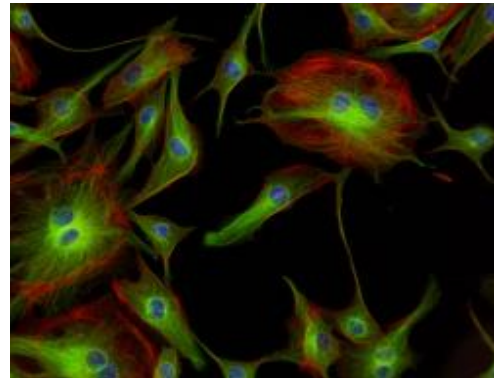
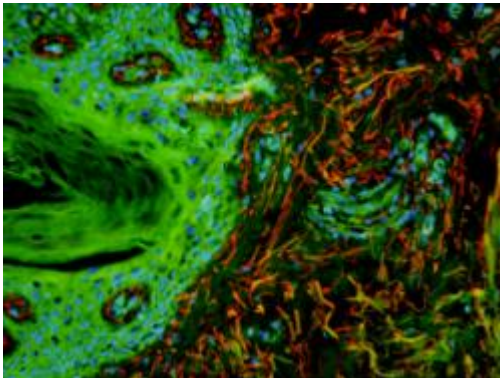
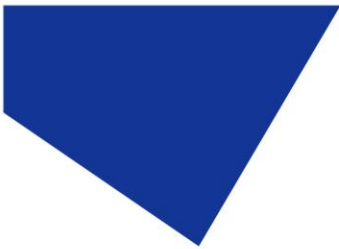


# Essential Commands: pE-800 Series & Amora Illumination Platform

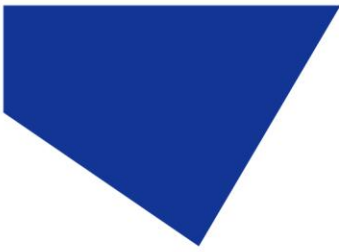




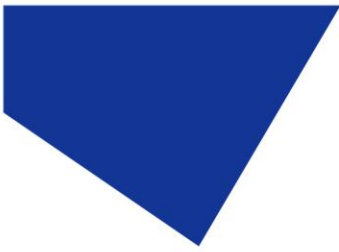
# Table of Contents

- 1 Hardware..... 3
  - 1.1 Supported CoolLED Products ..... 3
  - 1.2 Data connection via USB ..... 3
  - 1.3 TTL interface..... 3
  - 1.4 Analogue interface ..... 3
- 2 Setting up the Interface ..... 4
  - 2.1 Setting up the USB interface ..... 4
  - 2.2 Windows 10..... 4
  - 2.3 Windows 8 and earlier ..... 4
- 3 Software ..... 14
  - 3.1 Control Text Strings..... 14
  - 3.2 Using the Character String interface ..... 14
  - 3.3 Compatibility ..... 14
- 4 Status commands ..... 15
  - 4.1 Commands ..... 15
  - 4.2 Examples ..... 15
- 5 Monitoring commands..... 19
  - 5.1 Commands ..... 19
  - 5.2 Examples ..... 19
- 6 Output commands ..... 23
  - 6.1 Commands ..... 23
  - 6.2 Examples ..... 23
- 7 Global TTL commands ..... 25
  - 7.1 Commands ..... 25
- 8 Pod commands..... 26
  - 8.1 Commands ..... 26
  - 8.2 Examples ..... 26
- 9 Channel intensity commands..... 27
  - 9.1 Commands ..... 27
  - 9.2 Examples ..... 28

DOC-048 Iss 4 1



10	Preset commands.....	35
10.1	Commands .....	35
10.2	Examples .....	35
11	Sequence runner commands .....	37
11.1	Commands .....	37
11.2	Examples .....	38



# 1 Hardware

The pE-800 Series (henceforth referred to as 'pE-800') and Amora Illumination Platform has been designed to provide the user with a choice of ways of controlling the LED's functions from a PC with several connectors on the Illumination Platform:

- USB Virtual Serial Port.
- A 15-way D-type connector for TTL control.
- A 15-way D-type connector for Analogue control.
- A BNC connector for Global TTL control.

## 1.1 Supported CoolLED Products

The following CoolLED products support the commands set detailed in this document.

- pE-800
- pE-800<sup>fura</sup>
- Amora Illumination Platform

## 1.2 Data connection via USB

The pE-800/Amora Illumination Platform can be controlled via a USB connection as a virtual serial port.

## 1.3 TTL interface

This is the simplest method for remote control of the LED arrays. There is an individual TTL input for each physical light channel, which when set at a high, will switch on the specific array, overriding any manual settings or other remote-control signals. If rapid switching of the LEDs is required, this is the recommended method as it is independent of the state of the program cycle within the pE-800/Amora Illumination Platform. It also offers a global TTL that switches on all selected channels when supplied with a high input, and two TTL outputs that provide a TTL high signal when set to high.

## 1.4 Analogue interface

The pE-800/Amora Illumination Platform offers eight 0 V – 10 V analogue inputs, with which external equipment may control the intensity of the LED outputs. It also offers two 0 V – 10 V analogue outputs that may be used to control external equipment.



## 2 Setting up the Interface

### 2.1 Setting up the USB interface

The USB interface uses Microsoft's own built-in drivers for Virtual Serial Ports. This means that the pE-800/Amora Illumination Platform behaves very much as an ordinary COM port on your PC and can be controlled by any application that uses Windows COM ports correctly.

### 2.2 Windows 10

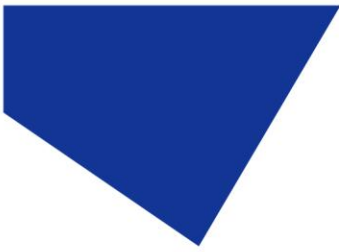
When you first plug your CoolLED system into your PC with the USB cable, Windows will automatically install the driver files.

### 2.3 Windows 8 and earlier

The first time your CoolLED Illumination System is connected to your PC via USB you shall need to follow the steps below to allow a virtual COM port to be assigned

#### 2.3.1

The CoolLED pE Driver needs to be downloaded from the CoolLED website. This file can be found on the following page: [www.cooled.com/support/imaging-software/](http://www.cooled.com/support/imaging-software/)  
Once downloaded this needs to be saved somewhere on your PC.

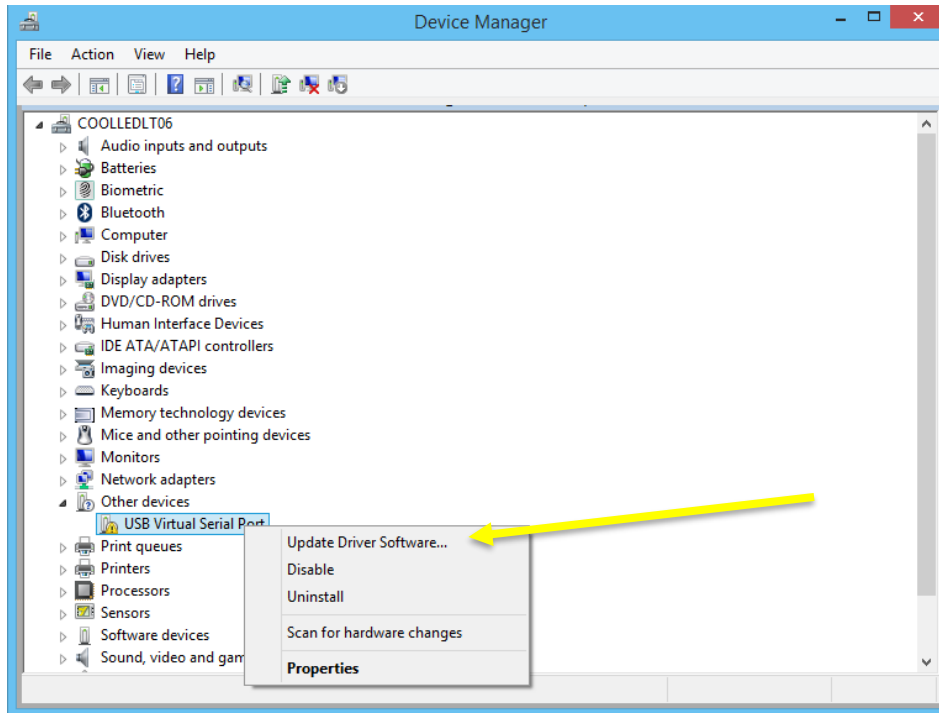


### 2.3.2

Navigate to the 'Device manager'.

The CoolLED Illumination System should initially be listed under 'Other devices' as 'USB Virtual Serial Port' with a yellow exclamation mark icon.

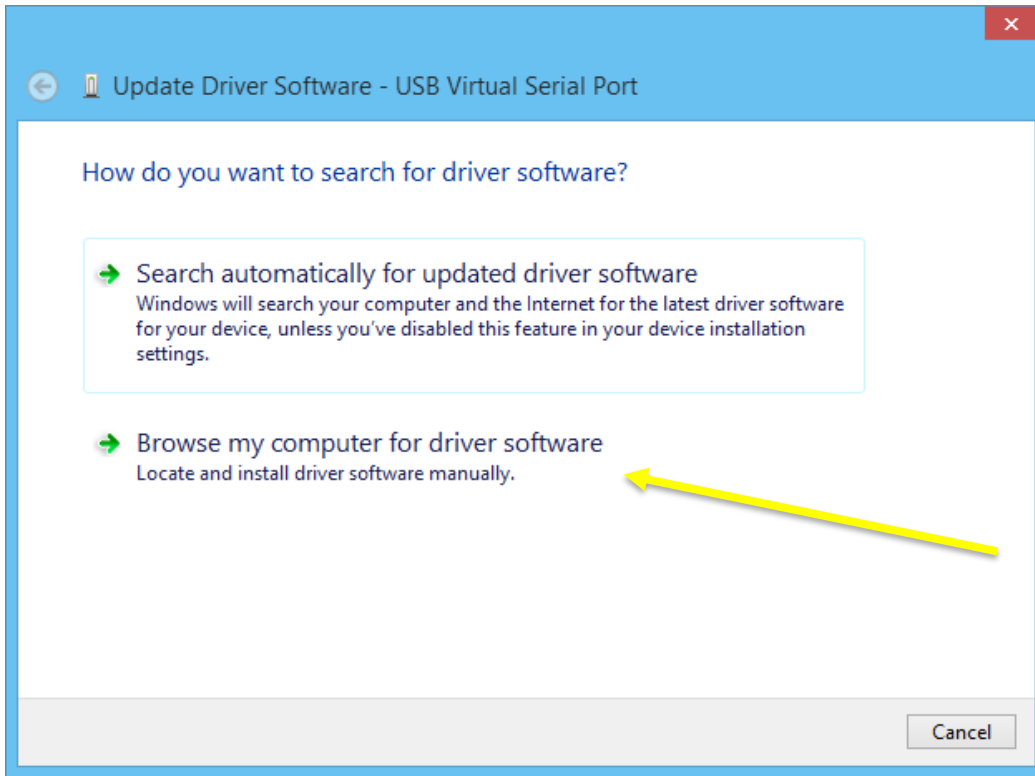
Right-click on this and select the 'Update Driver Software...' option from the list.





### 2.3.3

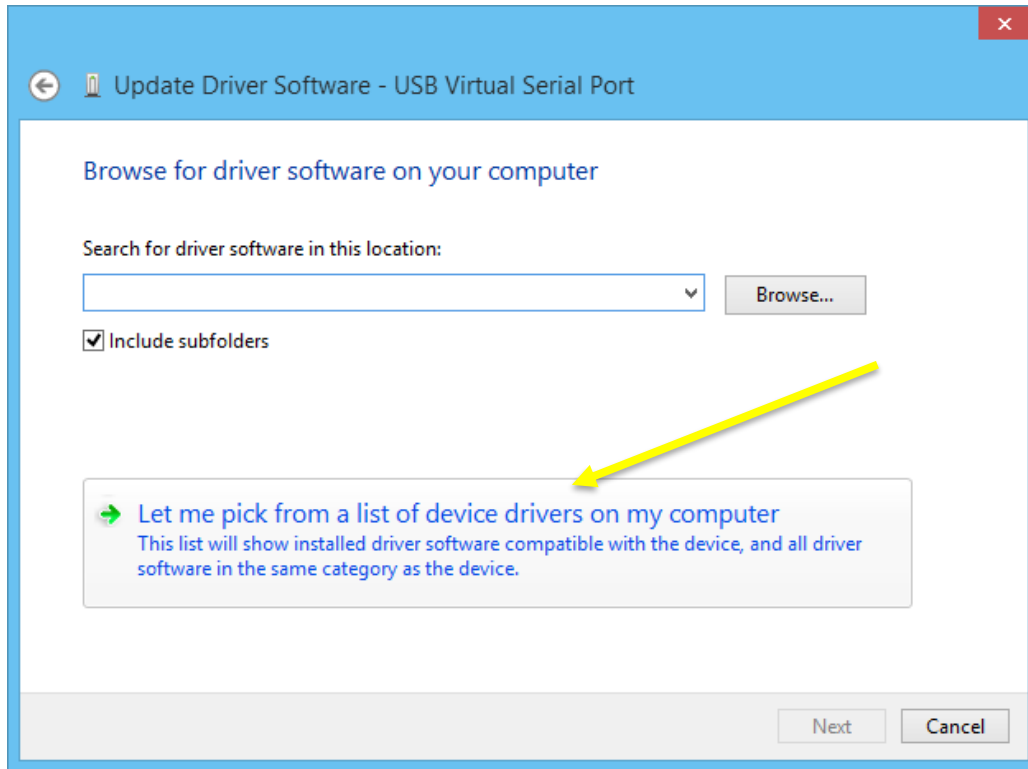
The window shown below will be displayed. Select the 'Browse my computer for driver software' option.

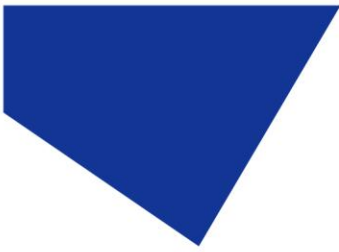




### 2.3.4

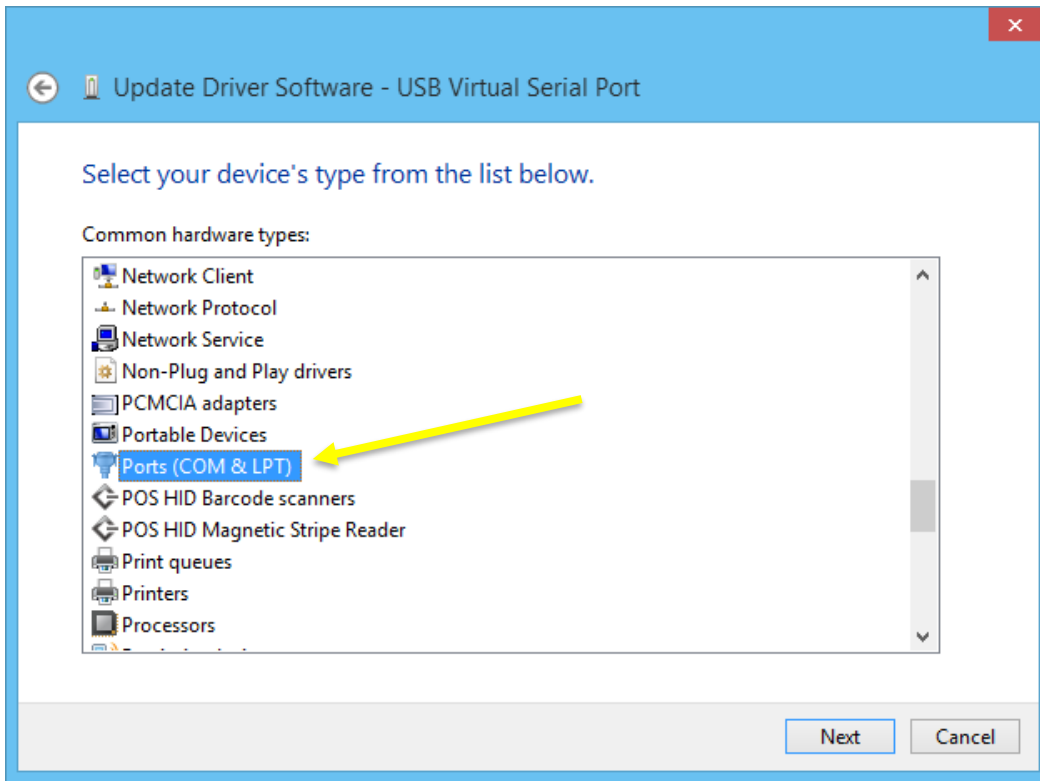
Select the 'Let me pick from a list of device drivers on my computer'.

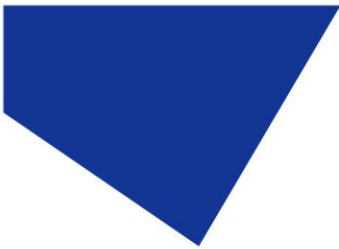




### 2.3.5

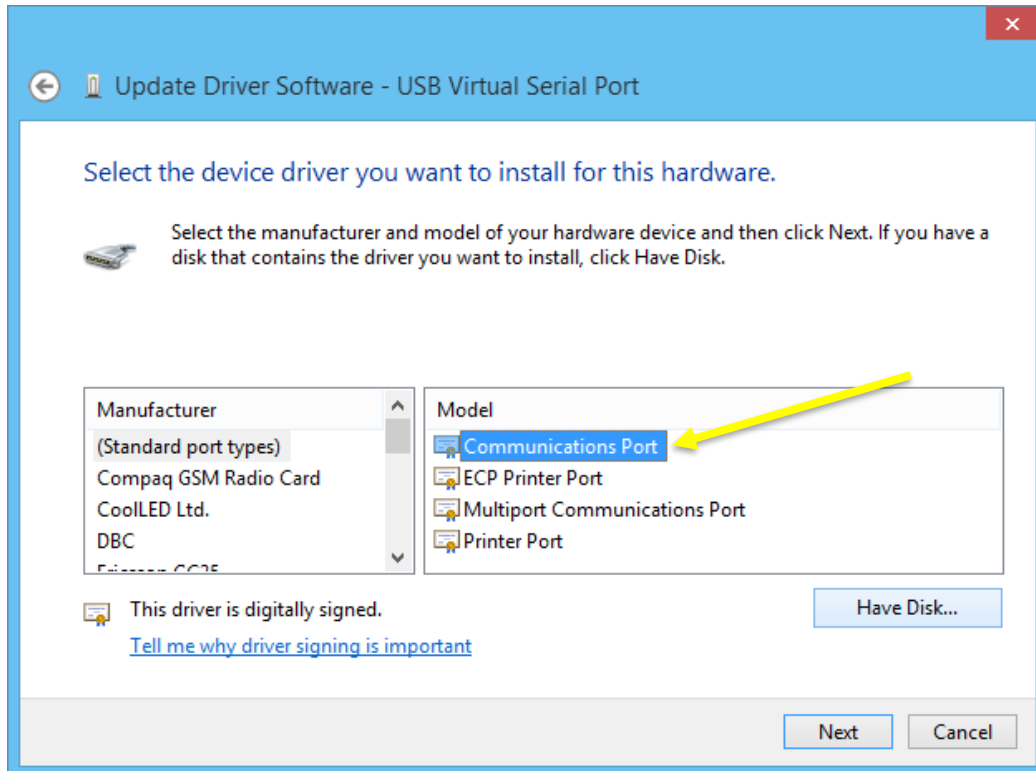
Select 'Port (COM & LPT)' and then click the 'Next' button.





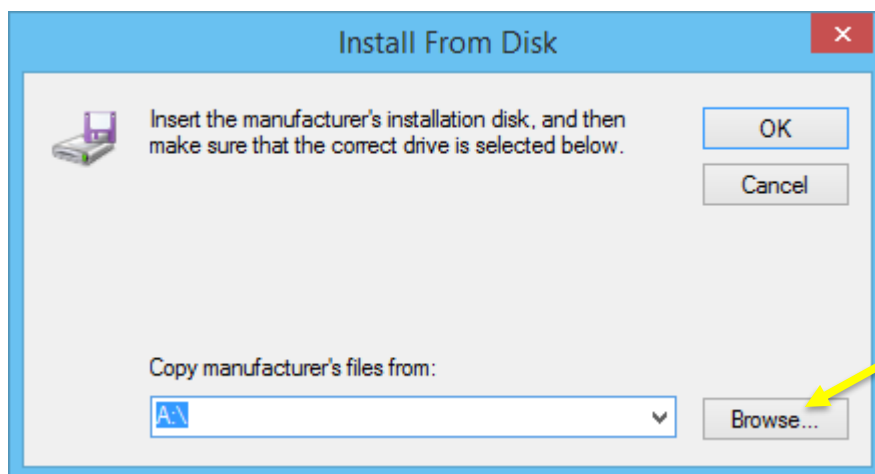
### 2.3.6

Select the '(Standard port types)' from the 'Manufacturer' field, select 'Communications Port' from the 'Model' field and press the 'Have Disk...' button.



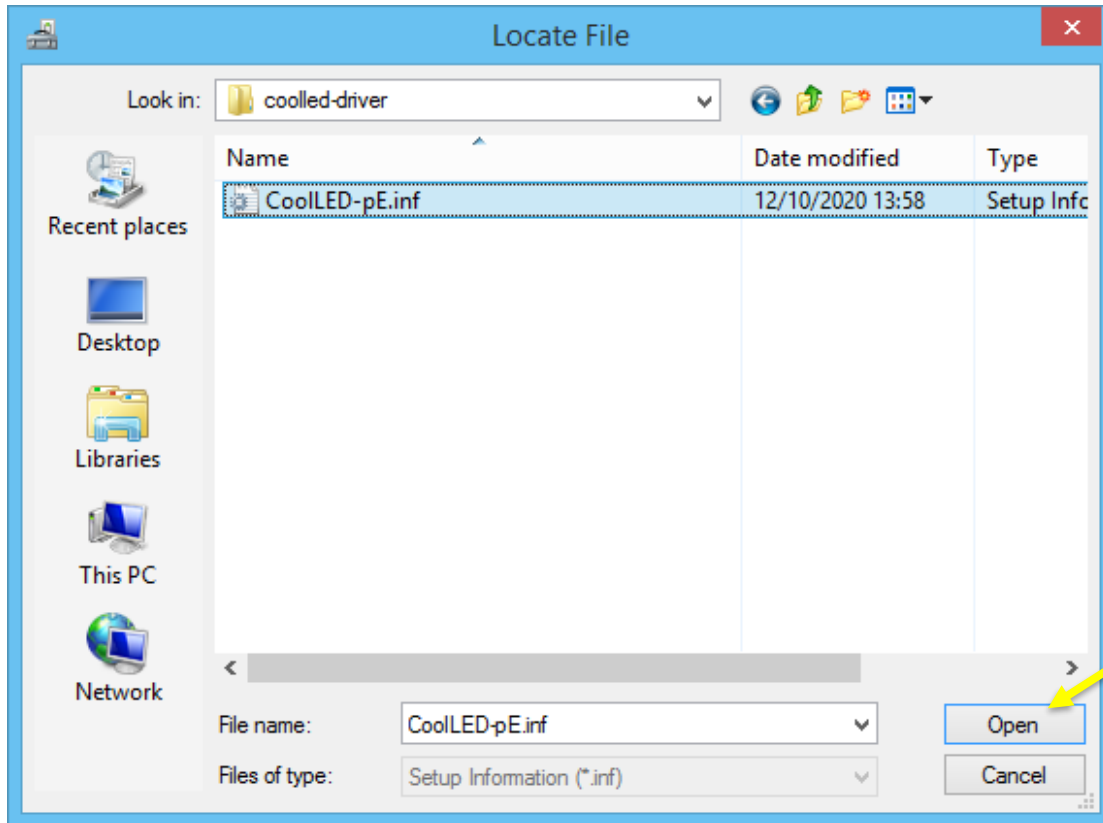
### 2.3.7

Press the 'Browse...' button on the 'Install From Disk' window.

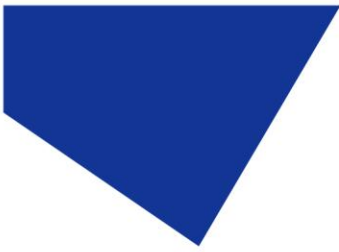




Locate the CoolLED pE Driver file and press the 'Open' button.

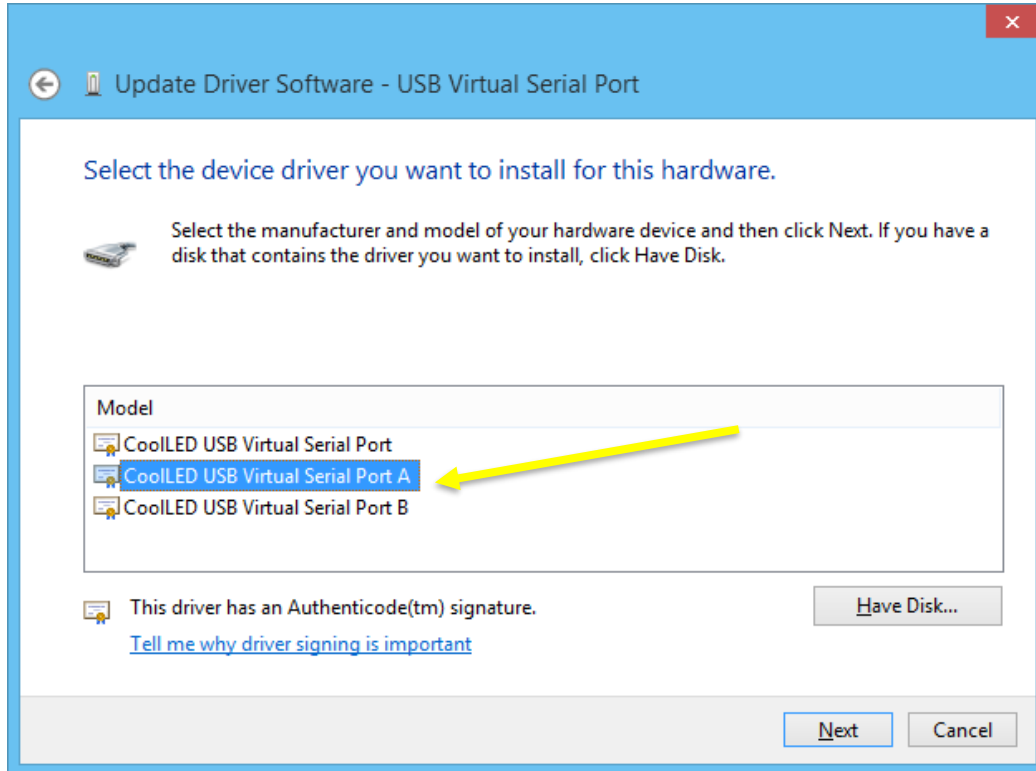


The 'Copy manufacturer's files from:' field will now be populated with the server location of the pE-Driver. Press 'OK' to confirm this.



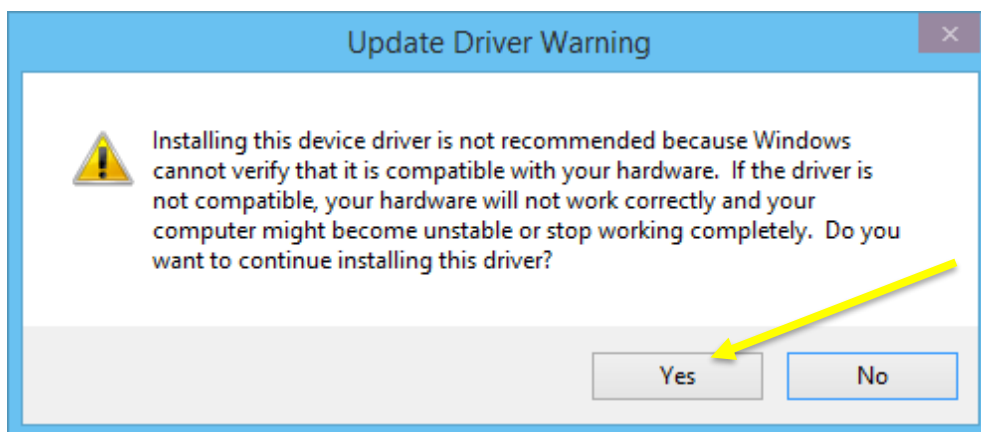
### 2.3.8

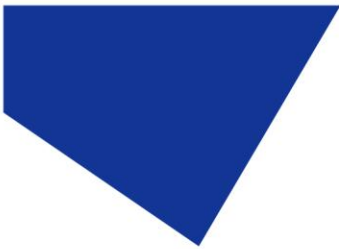
Select 'CoolLED USB Virtual Serial Port A' in the 'Model' field and press 'Next'.



### 2.3.9

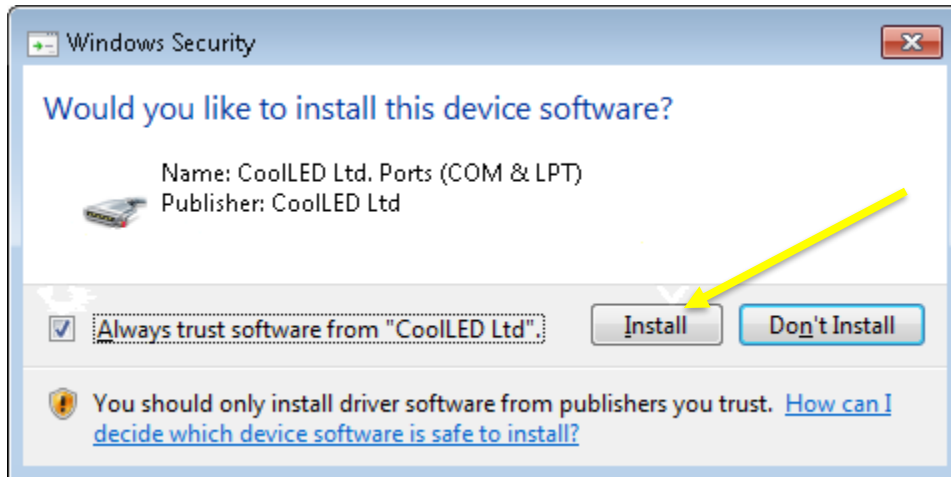
A warning window will be displayed. Press the 'Yes' button to confirm.





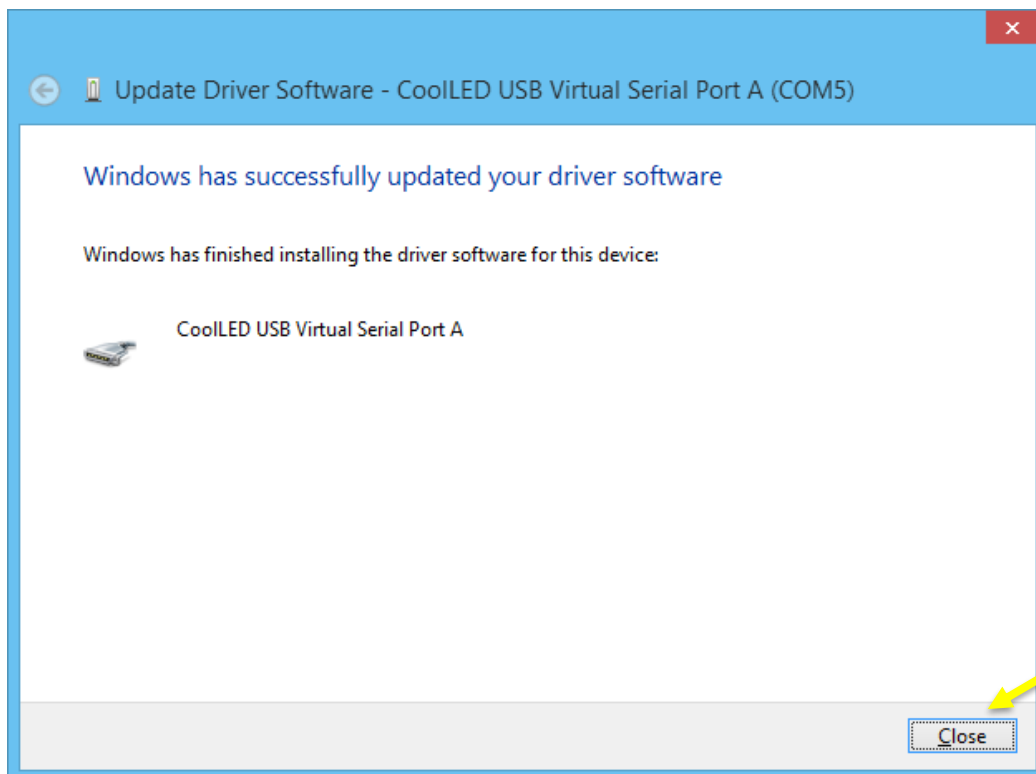
### 2.3.10

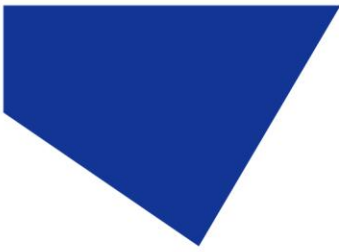
A Windows security page may be displayed at this stage. Press the check box labelled 'Always trust software from "CoolLED Ltd"' and press the 'Install' button.



### 2.3.11

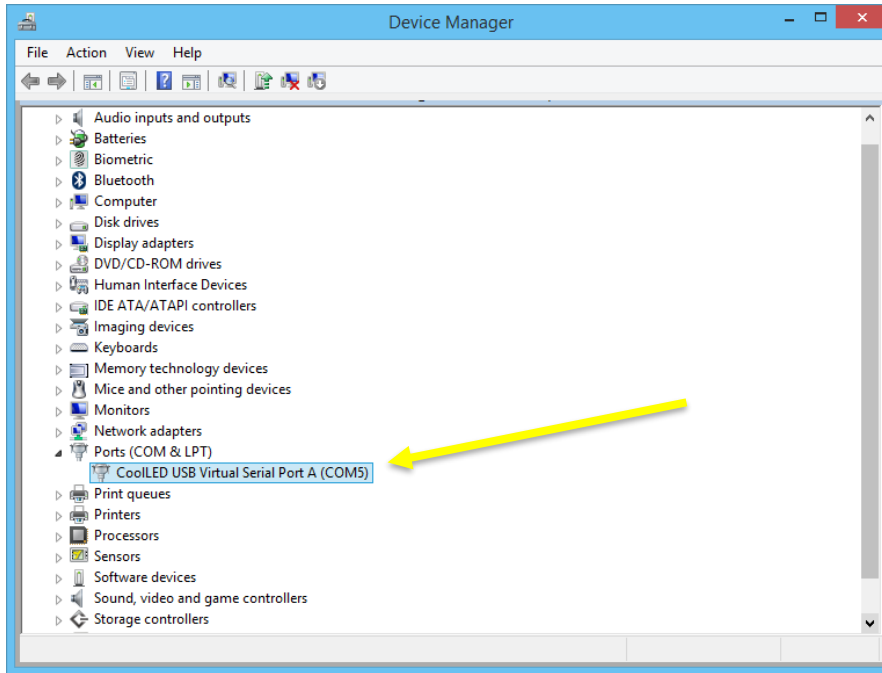
A window will be displayed confirming that the driver software has been updated. Press the 'Close' button. The pE-800 will now be ready to use via USB.

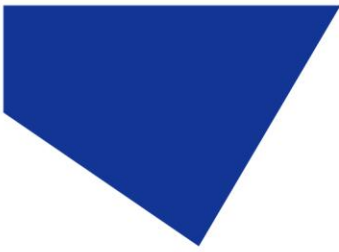




### 2.3.12

With the pE-800 now recognised by Windows, the specific Virtual COM port that has been assigned can be found in 'Device Manager' by expanding the 'Ports (COM & LPT)' field. The pE-800 will be listed as 'CoolLED USB Virtual Serial Port A'.





## 3 Software

### 3.1 Control Text Strings

The base interface between a controlling computer and the pE-800/Amora Illumination Platform is a simple set of text commands and responses usually sent via a USB “Virtual Serial Port” interface. These command strings are typically a few characters long, comprising of standard ASCII characters which are terminated by a character such as a null character, a carriage return, a newline, or a combination of carriage return and newline characters.

### 3.2 Using the Character String interface

Commands always consist of a string of characters terminated by a character such as a null character, a carriage return, a newline, or a combination of carriage return and newline characters. The commands have reasonable mnemonics for easier learning.

### 3.3 Compatibility

The pE-800/Amora Illumination Platform supports most of the legacy commands used for the pE-300 Series and pE-4000 Illumination Systems; the exceptions to this are the commands for the pE-4000 for the channels (LOAD, and LAMBDA), the regular reporting of the channels on the pE-300 Series and pE-4000 (XLIVE), and the global intensity commands (CS+ and CS-) for the pE-300 Series and pE-4000.

## 4 Status commands

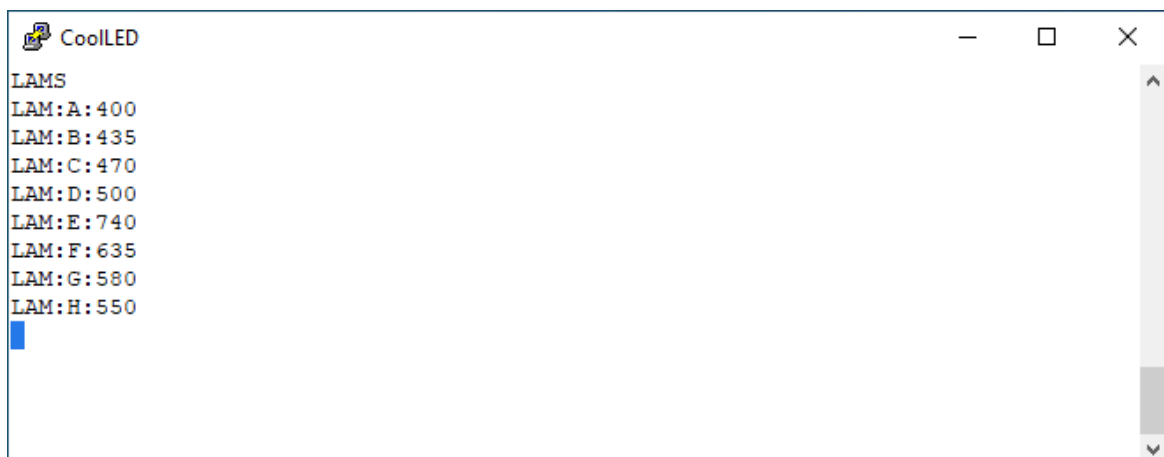
### 4.1 Commands

The following section is concerned with the commands that report on the general status of the pE-800/Amora Illumination Platform.

<b>LAMS</b>	Report wavelength of all channels
<b>XMODEL</b>	Report system model
<b>XVER</b>	Report system version
<b>XSERIAL</b>	Report system serial number
<b>XPART</b>	Report system part number
<b>LAMSN:x?</b>	Report LAM serial number, where <b>x</b> is a channel index A, B...H
<b>LAMPN:x?</b>	Report LAM part number, where <b>x</b> is a channel index A, B...H
<b>DRVSN:i?</b>	Report driver serial number, where <b>i</b> is a driver index number: 1 for channels A...D, and 2 for channels E...H
<b>DRVPN:i?</b>	Report driver part number, where <b>i</b> is a driver index number: 1 for channels A...D, and 2 for channels E...H

### 4.2 Examples

#### LAMS



```

CoolLED
LAMS
LAM:A: 400
LAM:B: 435
LAM:C: 470
LAM:D: 500
LAM:E: 740
LAM:F: 635
LAM:G: 580
LAM:H: 550

```

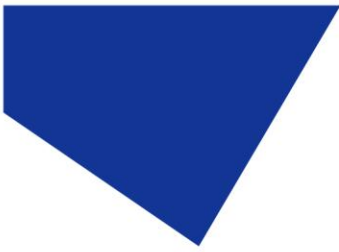
#### XMODEL



```

CoolLED
XMODEL
XMODEL=AMORA

```





**XVER**

```
CoolLED  
XVER  
XFW_VER=0.2.12
```

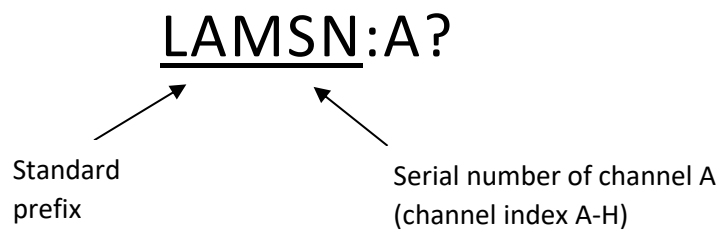
**XSERIAL**

```
CoolLED  
XSERIAL  
XSERIAL:UNIT L
```

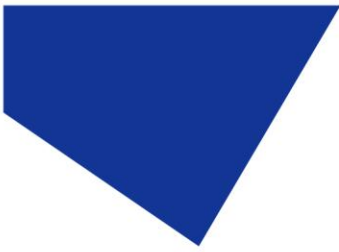
**XPART**

```
CoolLED  
XPART  
XPART:PART L
```

**LAMSN:x?**



```
CoolLED  
LAMSN:A?  
LAMSN:A=365LAM01234
```



LAMPN:x?

LAMPN:F?

Standard  
prefix

Serial number of channel F  
(channel index A-H)

```
CoolLED  
LAMPN:F?  
LAMPN:F=F1234567890
```

DRVSN:i?

DRVSN:1?

Standard  
prefix

Serial number of driver  
1 (channels A-D),  
2 (channels E-H)

```
CoolLED  
DRVSN:1?  
DRVSN:1=DRIVER L1
```

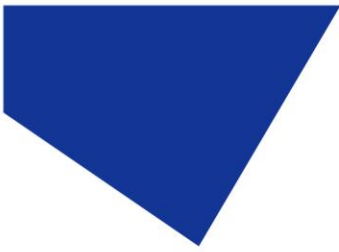
DRVPN:i?

DRVPN:2?

Standard  
prefix

Serial number of driver  
1 (channels A-D),  
2 (channels E-H)

```
CoolLED  
DRVPN:2?  
DRVPN:2=PART L2
```



## 5 Monitoring commands

### 5.1 Commands

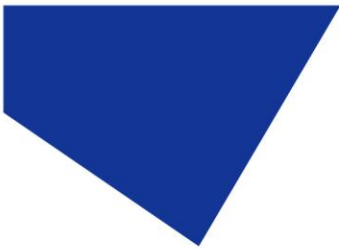
The following section is concerned with the commands that allow monitoring of the status of the pE-800/Amora Illumination Platform.

<b>FANMODE?</b>	Report fan mode
<b>FANMODE=i</b>	Set fan mode where <b>0</b> is auto mode and <b>1</b> is manual mode
<b>FAN:i?</b>	Report actual fan speed where <b>i</b> is the fan index 1 or 2
<b>FAN:i=d</b>	Set fan speed where <b>i</b> is the fan index 1 or 2, and <b>d</b> is the fan speed duty cycle 0 – 100 (manual operation mode only)
<b>TEMP:x?</b>	Report LAM thermistor temperature, where <b>x</b> is channel index A, B...H
<b>PHOTO:x?</b>	Report LAM photodiode result, where <b>x</b> is channel index A, B...H ( <b>currently does not give any response other than 0, but is for future implementation</b> )
<b>SYSTEM?</b>	Report system state. Will give state=0 if system is in ready state, state=1 if system is in warning state, and state=2 if system is in critical state
<b>USAGES</b>	Reports the system usage in hours
<b>FANFIT?</b>	Reports the number of fans fitted in the unit

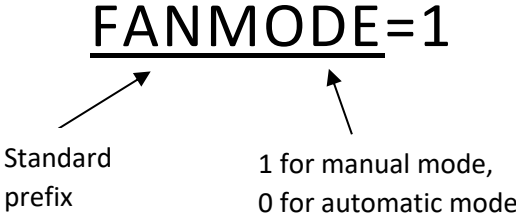
### 5.2 Examples

#### FANMODE?

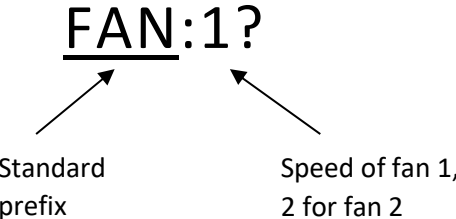
```
CoolLED
FANMODE?
FANMODE=MANUAL
```



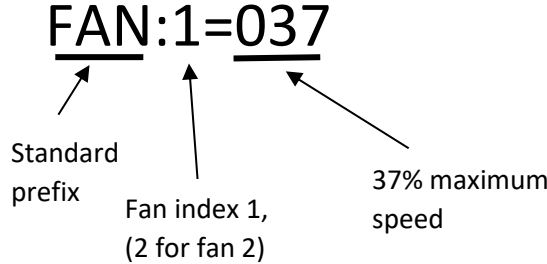
FANMODE=i

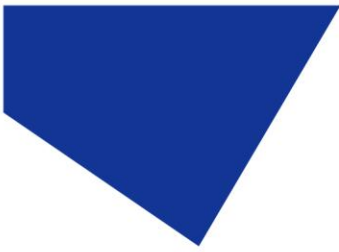


FAN:i?

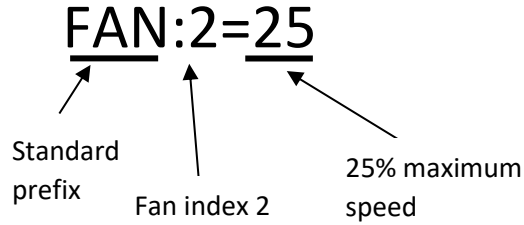


FAN:i=d



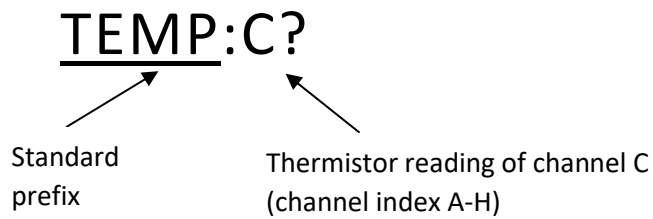


The input does not need to be a three-digit integer.



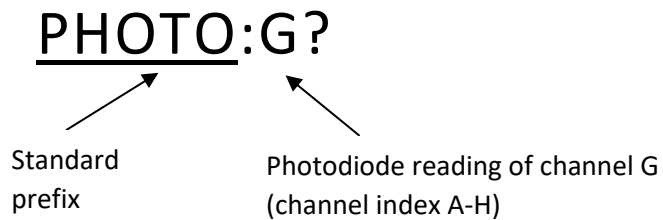
```
CoolLED
FAN:2=25
FAN:2=25
```

**TEMP:x?**

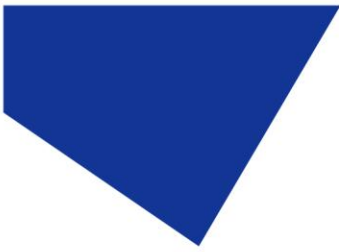


```
CoolLED
TEMP:C?
TEMP:C=31
```

**PHOTO:x?** (currently does not give any response other than 0, but is for future implementation)



```
CoolLED
PHOTO:G?
PHOTO:G=0
```



**SYSTEM?**

```
CoolLED  
SYSTEM?  
STATE=0
```

**USAGES**

```
CoolLED  
USAGES  
SYSTEM USAGE:1.8hr
```

**FANFIT?**

```
CoolLED  
FANFIT?  
FANFIT=2
```

## 6 Output commands

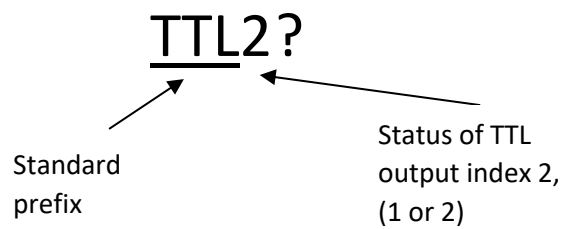
### 6.1 Commands

The following section is concerned with the commands that control the signal outputs of the pE-800/Amora Illumination Platform.

<b>TTLi?</b>	Report TTL output state, where <b>i</b> is the TTL index 1 or 2.
<b>TTLif</b>	Turn on/off specified TTL output, where <b>i</b> is the TTL index 1 or 2, and <b>f</b> is N for on, or F for off.
<b>AOi?</b>	Report analog output state, where <b>i</b> is the analog index 1 or 2.
<b>AOi:n</b>	Set analog output 0 V – 10 V, where <b>i</b> is the analog index 1 or 2, <b>n</b> is an integer number 0 (0 V) – 100 (10 V).

### 6.2 Examples

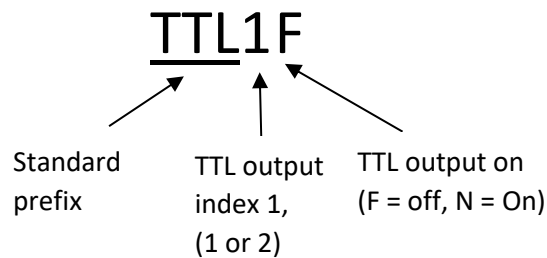
#### TTLi?




```

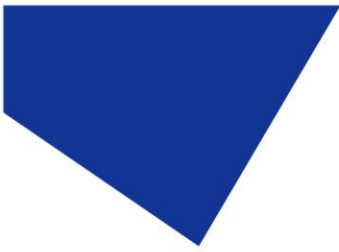
CoolLED
TTL2?
TTL2F
  
```

#### TTLif

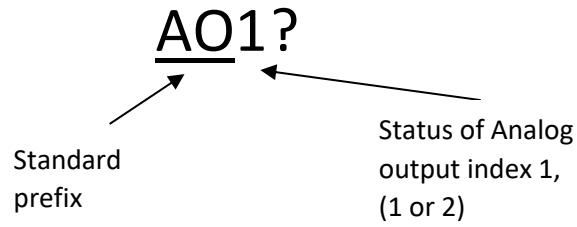



```

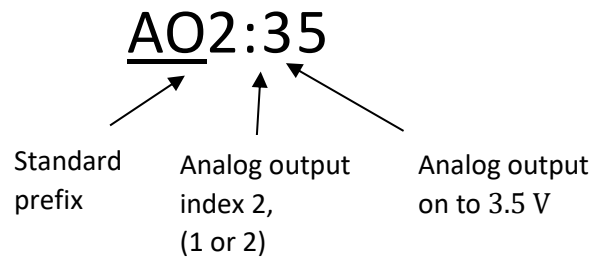
CoolLED
TTL1F
TTL1F
  
```



**AOi?**



**AOi:n**





## 7 Global TTL commands

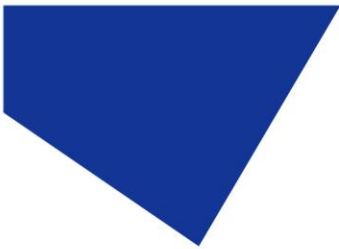
### 7.1 Commands

The following section is concerned with commands that allow to disable/enable the global TTL trigger.

**TTLGN** Enable global TTL trigger

**TTLGF** Disable global TTL trigger





## 8 Pod commands

### 8.1 Commands

The following section is concerned with commands that are enabled in the pE-800/Amora Illumination Platform, but are to be used with reporting to a control pod that is not currently available.

**PORT:P=ON**            Enable pod operation

**PORT:P=OFF**         Disable pod operation

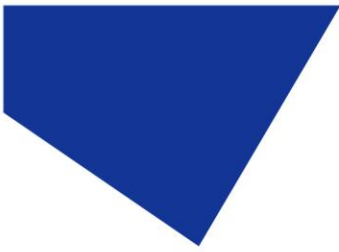
### 8.2 Examples

**PORT:P=ON**



**PORT:P=OFF**



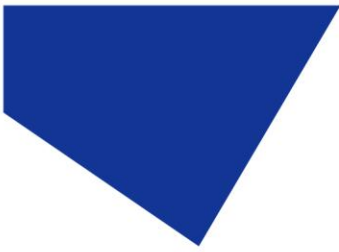


## 9 Channel intensity commands

### 9.1 Commands

The following section is concerned with the commands that control channel illumination of the pE-800/Amora Illumination Platform.

<b>C?</b>	Report on/off, selection status, and intensity of all channels separately in one line per channel.
<b>Cx?</b>	Report specified channels status and intensity to integer values (rounded down), where <b>x</b> is channel index A, B...H.
<b>CXx?</b>	Report specified channels status and intensity to one decimal place, where <b>x</b> is channel index A, B...H.
<b>Cxf</b>	Turn on/off specified channel at previous intensity, where <b>x</b> is channel index A, B...H, and <b>f</b> is N for on or F for off.
<b>Cxs</b>	Select/deselect specified channel, where <b>x</b> is channel index A, B...H, and <b>s</b> is S for selected or X for deselected.
<b>CSf</b>	Turn on/off all selected channels at previous intensity where <b>f</b> is N for on or F for off.
<b>CxIn</b>	Change specified channel intensity, where <b>x</b> is channel index A, B...H, and <b>n</b> is an integer intensity 0 (0%) – 100 (100%).
<b>CxIXn</b>	Change specified channel intensity, where <b>x</b> is channel index A, B...H, and <b>n</b> is a one decimal place intensity 0 (0%) – 1000 (100.0%).
<b>CSS?</b>	Report on/off, selection status, and integer intensity (rounded down) of all channels in one line.
<b>CSX?</b>	Report on/off, selection status, and one decimal place intensity of all channels in one line.
<b>CSSxsfm...</b>	Set on/off, selection status, and integer intensity of up to all 8 channels: where <b>x</b> is channel index A, B...H, <b>s</b> is S for selected or X for deselected, <b>f</b> is N for on or F for off, and <b>n</b> is an integer intensity 0 (0%) – 100 (100%).
<b>CSXxsfm...</b>	Set on/off, selection status, and integer intensity of up to all 8 channels: where <b>x</b> is channel index A, B...H, <b>s</b> is S for selected or X for deselected, <b>f</b> is N for on or F for off, and <b>n</b> is a one decimal place intensity 0 (0%) – 1000 (100.0%).



**ANx?** Report specified channels analog status, where **x** is channel index A, B...H.

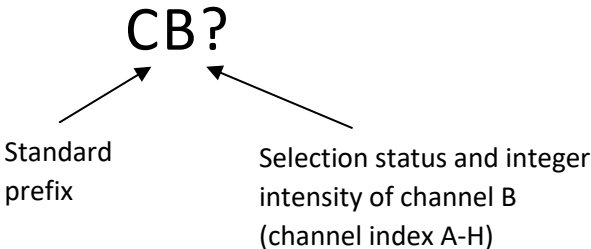
**ANxf** Enable/disable specified channel to be in analog mode, where **x** is channel index A, B...H, and **f** is N for on or F for off.

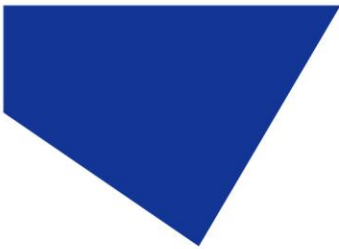
9.2 Examples

C?

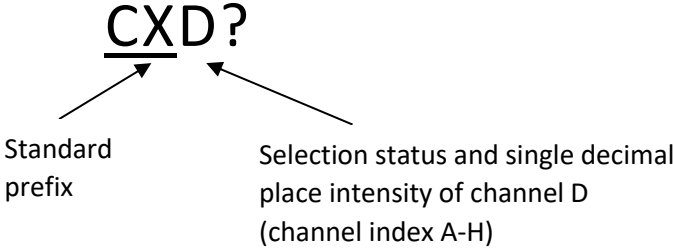


Cx?

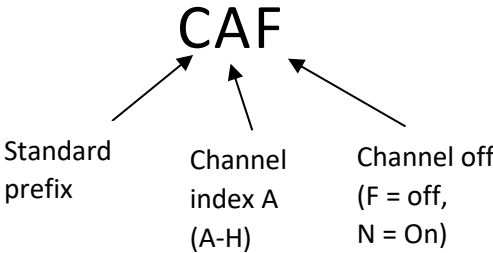




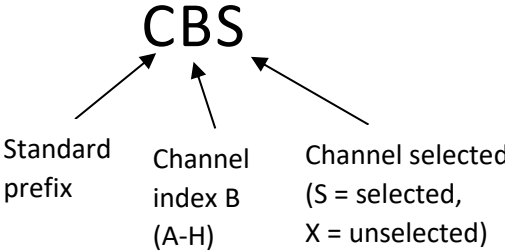
CXx?

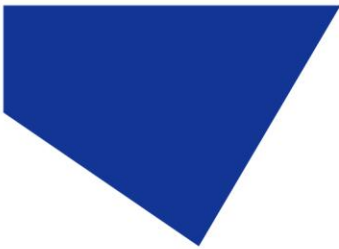


Cxf

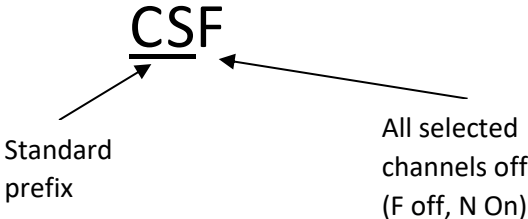


Cxs



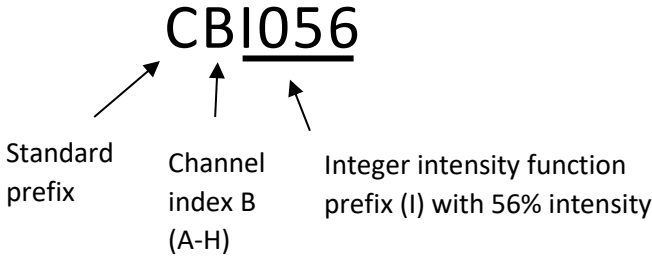


CSf



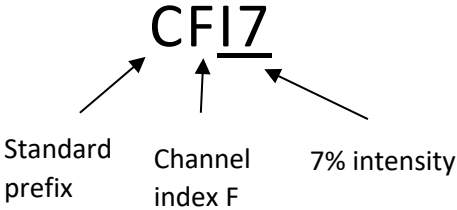
```
CoolLED
CSF
CSSASF003BSF056CSF050DXF048EXF000FSF007GSF029HSF055
```

CxIn



```
CoolLED
CBI056
CB056N
```

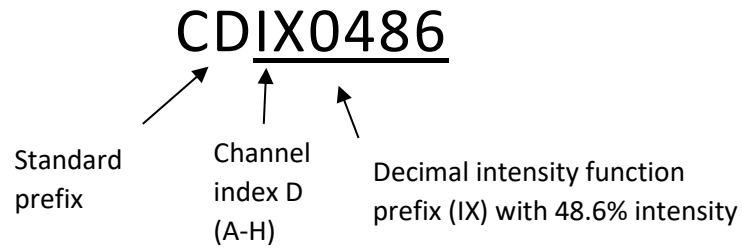
The response will always contain a three-digit integer.



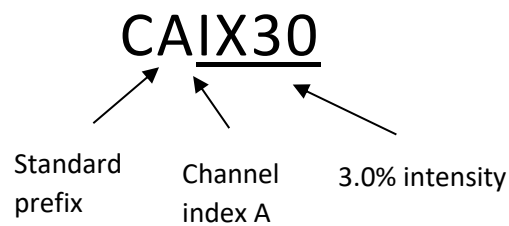
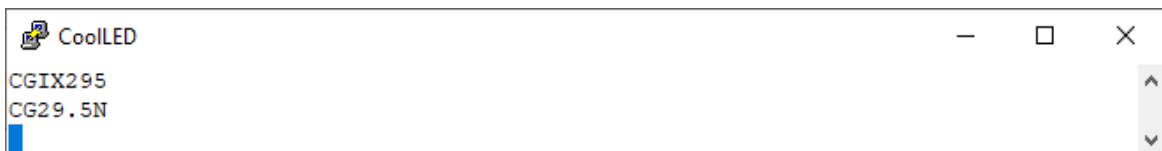
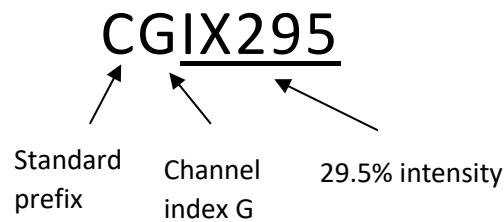
```
CoolLED
CFI7
CF007N
```



CxIXn



The response will always contain a single decimal place.





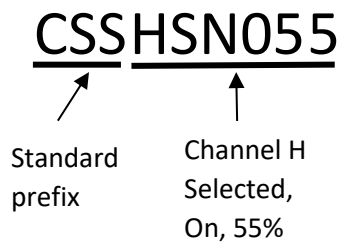
CSS?

```
CoolLED
CSS?
CSSASF030BSN050CSN050DXF000EXF000FSN075GSN063HSN055
```

CSX?

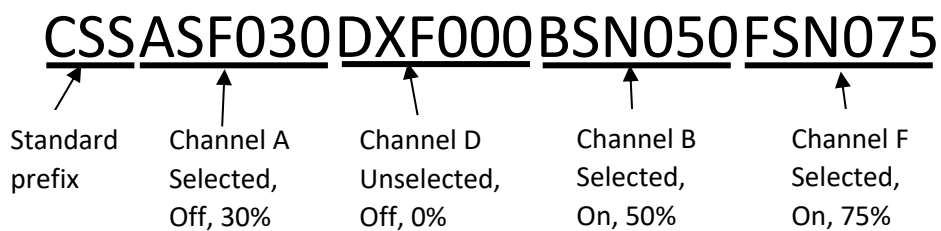
```
CoolLED
CSX?
CSXASF30.0BSN50.0CSN50.0DXF0.0EXF0.0FSN75.0GSN63.0HSN55.0
```

CSSxsf...



```
CoolLED
CSSHSN055
CSSASF030BSN050CSN050DXF000EXF000FSN075GSN063HSN055
```

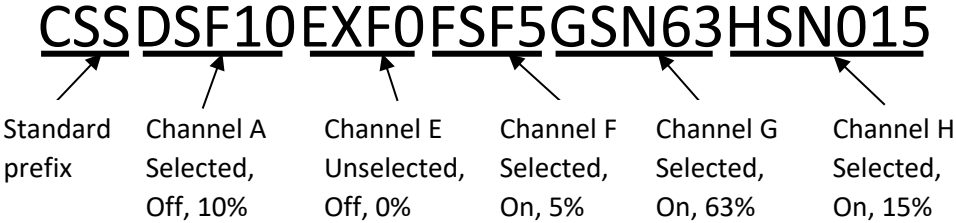
The response will always be in alphabetic order and contain all channels.



```
CoolLED
CSSASF030DXF000BSN050FSN075
CSSASF030BSN050CSN050DXF000EXF000FSN075GSN063HSN015
```

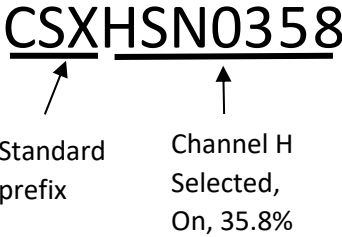


The response will always contain integers of three digits.



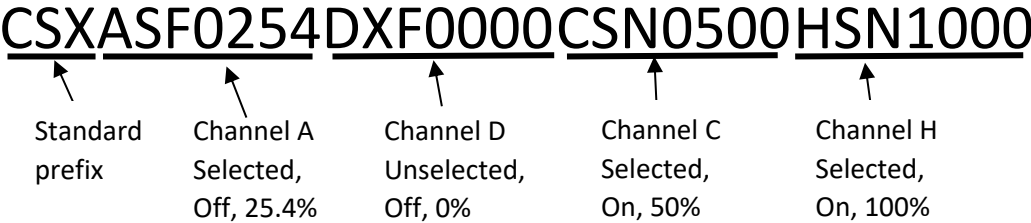
```
CoolLED
CSSDSF10EXF0FSF5GSN63HSN015
CSSASF025BXF050CSN050DSF010EXF000FSF005GSN063HSN015
```

CSXxsfn...

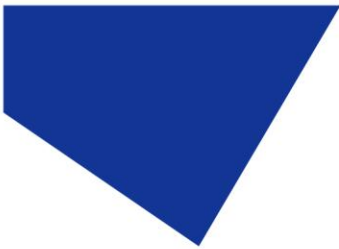


```
CoolLED
CSXHSN0358
CSXASF25.4BXF50.0CSN50.0DXF0.0EXF0.0FSN6.3GXF7.0HSN35.8
```

The response will always be in alphabetic order and contain all channels.

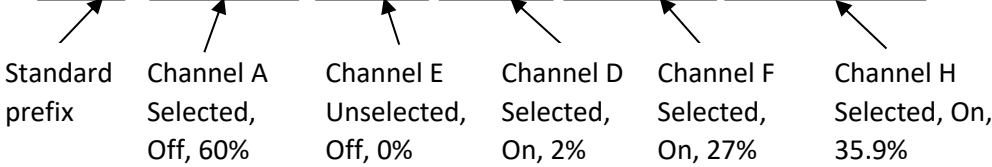


```
CoolLED
CSXASF0254DXF0000CSN0500HSN1000
CSXASF25.4BXF50.0CSN50.0DXF0.0EXF0.0FSN6.3GXF7.0HSN100.0
```



The response will always contain a single decimal place.

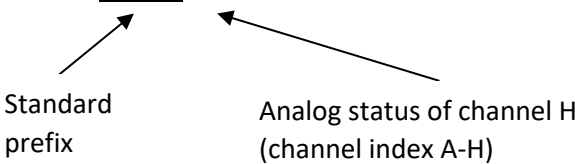
CSXASF60EXF0DSF2FSN63HSN359



```
CoolLED
CSXASF60EXF0DSF2FSN63HSN359
CSXASF6.0BXF50.0CXF100.0DSF0.2EXF0.0FSN6.3GXF7.0HSN35.9
```

ANx?

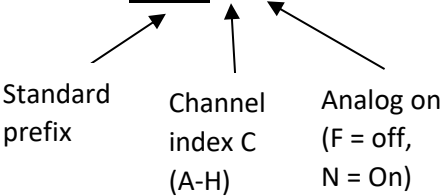
ANH?



```
CoolLED
ANH?
ANH?
```

ANxf

ANCN



```
CoolLED
ANCN
ANCN
```

## 10 Preset commands

### 10.1 Commands

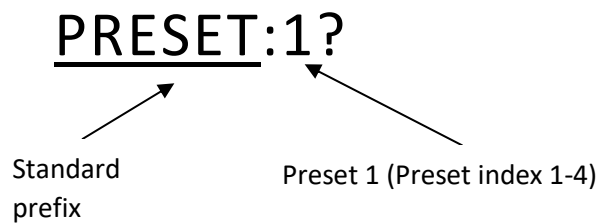
The following section is concerned with the commands that allow the setup of up to 4 presets of the pE-800/Amora Illumination Platform.

**PRESET:x?** Report a preset, where **x** is the preset index 1-4

**PRESET:x=c:s:d:n** Set up a preset where **x** is the preset index 1-4, **c** is the channel index, **s** is the channel selection state, **d** is an integer intensity 0 (0% - 100 (100%)), **n** is the preset name. A preset must address all channels

### 10.2 Examples

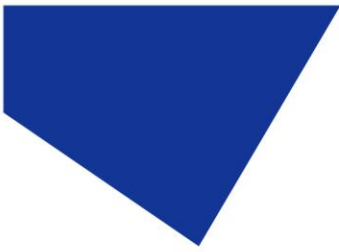
PRESET:x?



```

CoolLED
PRESET:1?
PRESET:1=AS100BS070CX000DX000EX000FX000GX000HX000:TEST
  
```





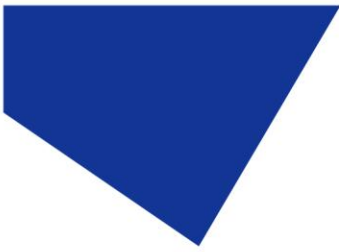
## 11 Sequence runner commands

### 11.1 Commands

The following section is concerned with the commands that allow sequences of illuminations triggered by an external TTL signal to the pE-800/Amora Illumination Platform. Some of these commands are covered in section 0, but they can also be used to monitor or affect the sequence runner function.

Note that with the exception of the command to set up a sequence (**CSSxSfn...**), any other command only reflects the on/off (**N** or **F**) of the channel being polled at the instant of polling. The behaviour of the **CSF** command is also slightly different in sequence mode as noted below.

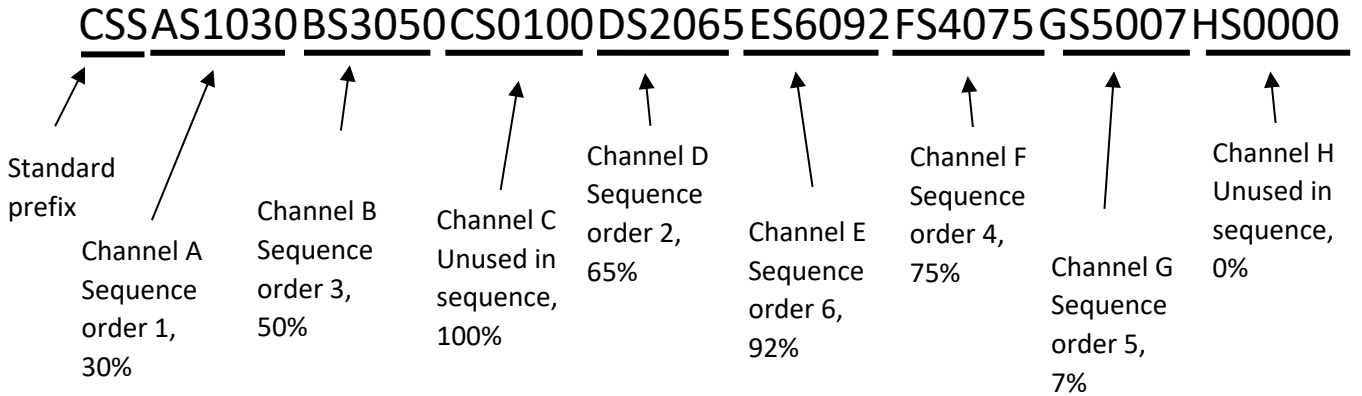
<b>CSSxSfn...</b>	Set on/off, selection status, and integer intensity of all 8 channels: where <b>x</b> is channel index A, B...H, <b>f</b> is the sequence index 0 – 8, and <b>n</b> is a three-digit integer intensity 0 (0%) – 100 (100%). Must address all channels.
<b>C?</b>	Report intensity and sequence order of all channels separately in one line per channel.
<b>CSS?</b>	Report integer intensity (rounded down) and sequence order of all channels in one line.
<b>CSX?</b>	Report one decimal place intensity and sequence order of all channels in one line.
<b>Cx?</b>	Report specified channels status and intensity and sequence order to integer values (rounded down), where <b>x</b> is channel index A, B...H.
<b>CXx?</b>	Report specified channels intensity to one decimal place and sequence order where <b>x</b> is channel index A, B...H.
<b>CSF</b>	Turns off and unselects all at previous intensities, stops the sequence.
<b>CxIn</b>	Change specified channel intensity, where <b>x</b> is channel index A, B...H, and <b>n</b> is an integer intensity 0 (0%) – 100 (100%).
<b>CxIXn</b>	Change specified channel intensity, where <b>x</b> is channel index A, B...H, and <b>n</b> is a one decimal place intensity 0 (0%) – 1000 (100.0%)



## 11.2 Examples

### CSSxsfm...

The input must always be in alphabetic order, contain all channels, and contain three-digit integers.



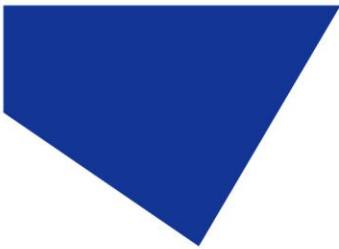
```
CoolLED
CSSAS1030BS3050CS0100DS2065ES6092FS4075GS5007HS0000
CSSAS1030BS3050CS0100DS2065ES6092FS4075GS5007HS0000
```

### C?

```
CoolLED
C?
CA0301
CB0503
CC1000
CD0652
CE0926
CF0754
CG0075
CH0000
```

### CSS?

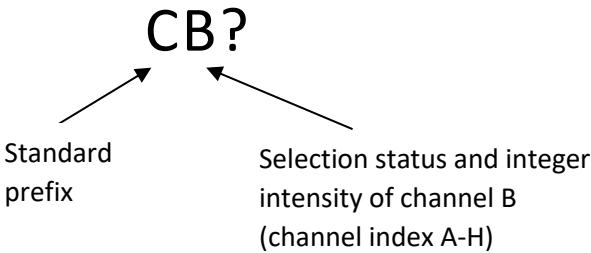
```
CoolLED
CSS?
CSSAS1030BS3050CS0100DS2065ES6092FS4075GS5007HS0000
```



CSX?

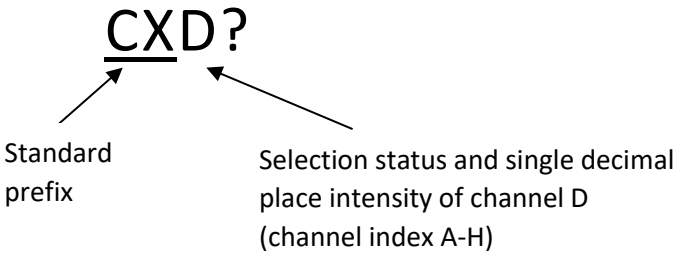
```
CoolLED  
CSX?  
CSXAS130.0BS350.0CS0100.0DS265.0ES692.0FS475.0GS57.0HS00.0
```

Cx?



```
CoolLED  
CB?  
CB0563
```

CXx?



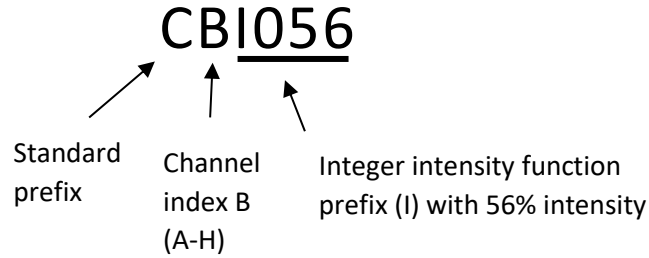
```
CoolLED  
CXD?  
CD48.62
```

CSF

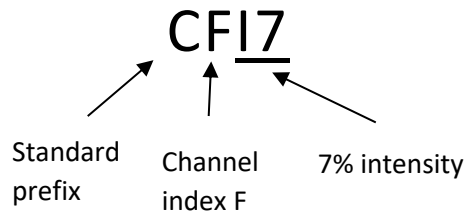
```
CoolLED  
CSF  
CSSAXF003BXF056CXF100DXF048EXF092FXF007GXF029HXF000
```



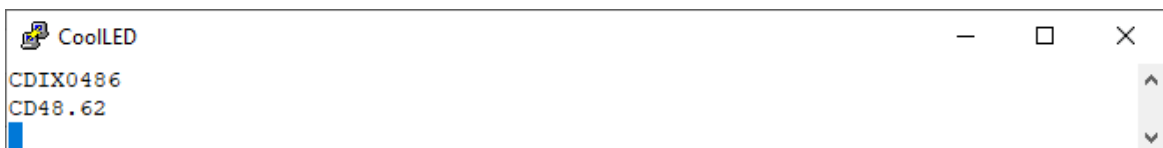
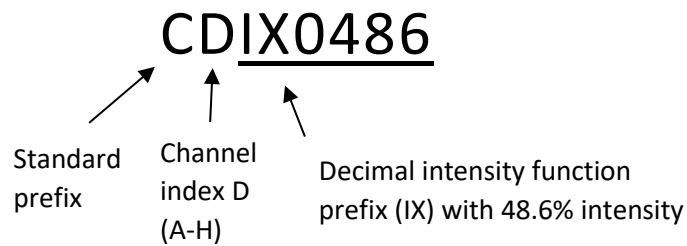
**CxIn**

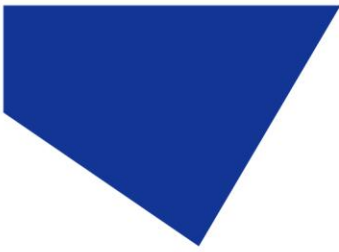


The response will always contain a three-digit integer.



**CxIXn**





The response will always contain a single decimal place.

