



Aphos RS485 packet specification

Document Number CA80-0160

Table of Contents

Introduction.....	1
COM setup.....	1
Packet structure	1
Host to Apos	1
Apos to Host	2
Controlling the light.....	2
Setting the control mode of the light	2
Enabling and disabling the light (RS485 control mode only).....	3
Controlling the strobe function on the light*.....	4
Setting the light level in RS485 mode.....	4
Setting the lights DAC scale factor	5
Querying the LED temperature	5
Querying the lights status.....	5
Broadcasting to multiple lights.....	6
Tips on using Putty	7
Command reference.....	7
Command line examples	9
Setting a light for triac dimming mode with the burden resistor enabled ^{Note 1/ Note 2}	9
Setting a light for RS485 mode with burden disabled, light level at 25%, LED's enabled ^{Note 2/Note 3}	10
Setting a light for RS485 mode with burden disabled, light level at 65%, LED's disabled ^{Note 2}	10
Setting a light to External DC control mode ^{Note 3}	10
Setting a light to External DC control mode with strobe enabled ^{Note 4}	10
Setting a lights DAC scale factor	10
Reading a lights serial number	10
Setting a lights serial number to '123'.....	10
Reading a lights firmware revision	10
Changing a lights RS485 address to 3.....	10
Communicating with a light that is not responding	10
Query the light at the two available baud rates.....	11
Use the broadcast address to configure the light	11
Document Revision History	13

Introduction

This document describes the RS485 packet structure defined for communications with the Aphos series of subsea lights. The Aphos lights implement a printable ASCII packet structure. The packet is variable length and uses start and stop delimiters. The protocol is a Master – Slave type. The light will not initiate a command by itself. It will only respond to the host.

COM setup

The baud rate used to communicate with the lights is 57600. The data format used is 8 data bits, no parity and 1 stop bit.

Packet structure

Host to Aphos

Figure 1 below illustrates the packet structure for querying or commanding the light.

1	2	3	4	5	6	7	8	9	10	11	12
<	C	,	Address	,	Register	,	Value 1	Value 2	*	Checksum	>

Figure 1 RS485 packet structure – Host to Aphos

When sending a query or command to a light the packet fields are broken down as follows;

Field	Value	Description
1	"<"	Packet start delimiter
2	"C"	Command or Query from Host to light.
3	","	comma - internal field delimiter
4	"address"	Number from 1 to 255. This is the address of the light on the wire
5	","	comma - internal field delimiter
6	"register"	Register you want to access
7	","	comma - internal field delimiter
8	"value 1"	Command specific value (LED Channel identifier)
9	"value 2"	Command specific value
10	"*"	Internal field delimiter to indicate that the following field is a checksum
11	"Checksum"	The checksum can be calculated at the host and sent down. It is a 16 number. Alternatively, by sending "ZZZZ" in place of the checksum, the command will still be executed by the light.
12	">"	Packet end delimiter

Aphos to Host

Figure 2 below illustrates the packet structure used by the light when responding to the host. The packet below is a single character response packet. In this case field 6 is the value returned from the light. Some queries from the light will result in more than one character being returned.

1	2	3	4	5	6	7	8	9
<	R	,	Address	,	value	*	Checksum	>

Figure 2 RS485 packet structure – Aphos to host

Field	Value	Description
1	"<"	Packet start delimiter
2	"R"	Response from Host to light.
3	","	comma - internal field delimiter
4	"address"	Number from 1 to 255. This is the address of the light on the wire
5	","	comma - internal field delimiter
6	"register"	Register you want to access
7	","	comma - internal field delimiter
8	"value 1"	Command specific value
9	"*"	Internal field delimiter to indicate that the following field is a checksum
10	"Checksum"	The Aphos will always return a checksum
11	">"	Packet end delimiter

Controlling the light

The light is relatively simple to control. The most frequently used commands are explained in this section.

Setting the control mode of the light

Aphos lights can be ordered with the following control interfaces;

1. Phase dimming
2. RS485
3. External DC level
4. Or all of the above

In the case where the light has been built with all control interfaces in place, you can choose the operational mode by sending a command over the RS485 interface. The control mode of the light is stored in EEPROM on the light. This means that you only have to send the command once, the light will always start up in the specified mode.

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,device_mode 0*ZZZZ>	None	This command sets the light to RS485 mode
<C,2,device_mode 1*ZZZZ>	None	This command sets the light to Triac (Phase Dimming) mode
<C,2,device_mode 2*ZZZZ>	None	This command sets the light to external DC control mode

Enabling and disabling the light (RS485 control mode only)

The commands below can be used to enable or disable the output of the light when operating in RS485 mode. The command is detailed in figure 2. Refer to the Command reference section for a list of all commands, their format and the expected reply.

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,c_state 0 1*ZZZZ>	None	This command enables the light
<C,2,c_state 0 0*ZZZZ>	None	This command disables the light

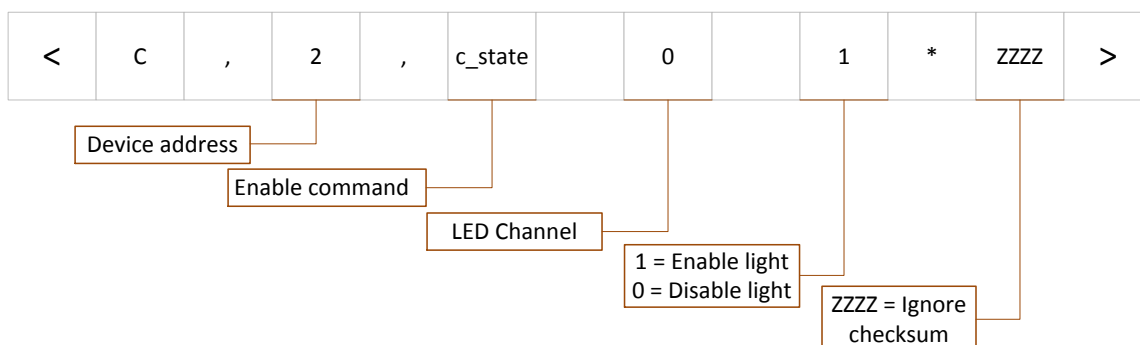


Figure 2 Enable command details

The LED channel identifier is only applicable to the Cathx Series 16 light. The Series 16 light has 4 independent channels that can be controlled individually. On the series 4 light, there is one LED channel, channel 0.

If you wish to find out if the light is enabled or disabled you query the light as follows;

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,c_state 0*ZZZZ>	<R,2,1*0172> <R,2,0*0171>	Aphos is enabled Aphos is disabled

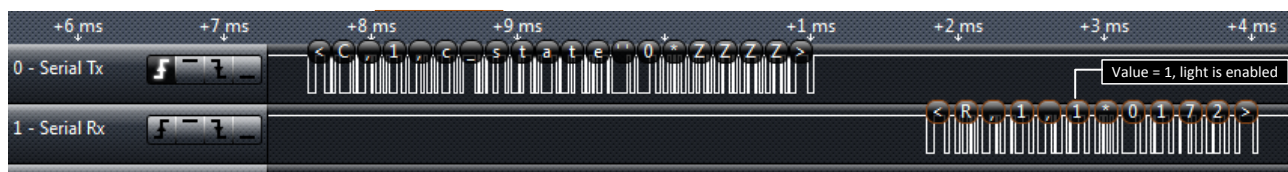


Figure 3 Logic analyser screen grab of host querying the light

Figure 3 shows a situation where the host sent a query of the Aphos enable bit. The Aphos responded indicating that this light is enabled

Controlling the strobe function on the light*

The commands below can be used to enable / disable and configure the strobe operation on the light. Refer to the Command reference section for a list of all commands, their format and the expected reply.

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,s_enable 1*ZZZZ>	None	This command enables the strobe
<C,2,s_enable 0*ZZZZ>	None	This command disables the strobe
<C,2,s_pulse 15000*ZZZZ>	None	This command sets the strobe duration to 15000 usec
<C,2,s_pulse*ZZZZ>	<R,2,15000*127A>	Send the command without a parameter to read back the lights strobe duration register
<C,2,s_delay 100*ZZZZ>	None	This command sets the delay from trigger detect to light output in usec. In this case, the strobe
<C,2,s_delay*ZZZZ>	<R,2,100*176>	Send the command without a parameter to read back the lights strobe delay register

An example is illustrated showing how to set the strobe duration to 30,000usec. Note that the strobe has a max duration of 65535usec.

<C,2,s_pulse 30000*ZZZZ> → Set the pulse duration

<C,2,s_enable 1*ZZZZ> → Enable strobe mode

Note: Do not set the strobe delay register to 0. It is recommended to keep this at 100usec (100)

Setting the light level in RS485 mode

The command below shows the format of the light level packet. No checksum is sent with this packet.

*Only applicable if the light has been ordered with strobe capability

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,c_level 0 55*ZZZZ>	None	Set the light level to 55%

As above this packet is setting the level on channel 0 of the light. For the series 4 this is the only LED channel.

Setting the lights DAC scale factor

The command below will set the scale factor of the light. The scale factor is an upper limit of power the light can output. The light level command is multiplied by this factor in the light to produce the output power. The DAC scale is a percentage value. The table below illustrates this;

Light level range	DAC scale (dac_scale)	Power level applied to LEDs
0% to 100%	100%	0 to 100%
0% to 100%	75%	0 to 75%
0% to 100%	40%	0 to 40%

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,dac_scale 55*ZZZZ>	None	Set the DAC scale to 55% of max

Querying the LED temperature

The command below will return the LED temperature.

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,temp*ZZZZ>	<R,2,50.0*0204>	Light response, LED's at 50.0 Celsius

Querying the lights status

The light monitors the temperature of the LED's and will take independent action if the LEDs exceed a predefined temperature limit. If the LED's do exceed this temperature the light enters its temperature protection mode. It is possible to query the lights status register to find out what is condition is. The command is detailed below;

Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,2,status*ZZZZ>	<R,2,0,30.0*026A>	Aphos is in normal operating state and temp of lights is 30.0 Celsius
	<R,2,1,67.6*0394>	Aphos is in a temperature protection state and temp of lights is 67.6 Celsius

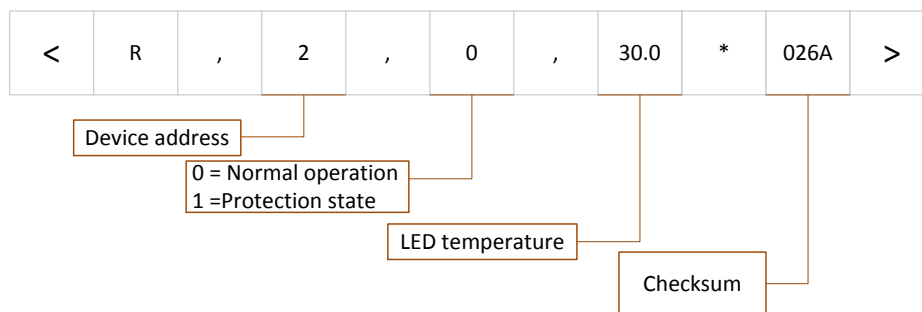


Figure 4 Status response details

Broadcasting to multiple lights

The address 255 has been reserved as a broadcast address for the Aphos lights. The purpose of this command is to allow a string of lights to be commanded at one time. An example of this would be the case where it is desired to control the level of a number of lights simultaneously. The commands below show how a number of lights can be enabled and have their levels set simultaneously

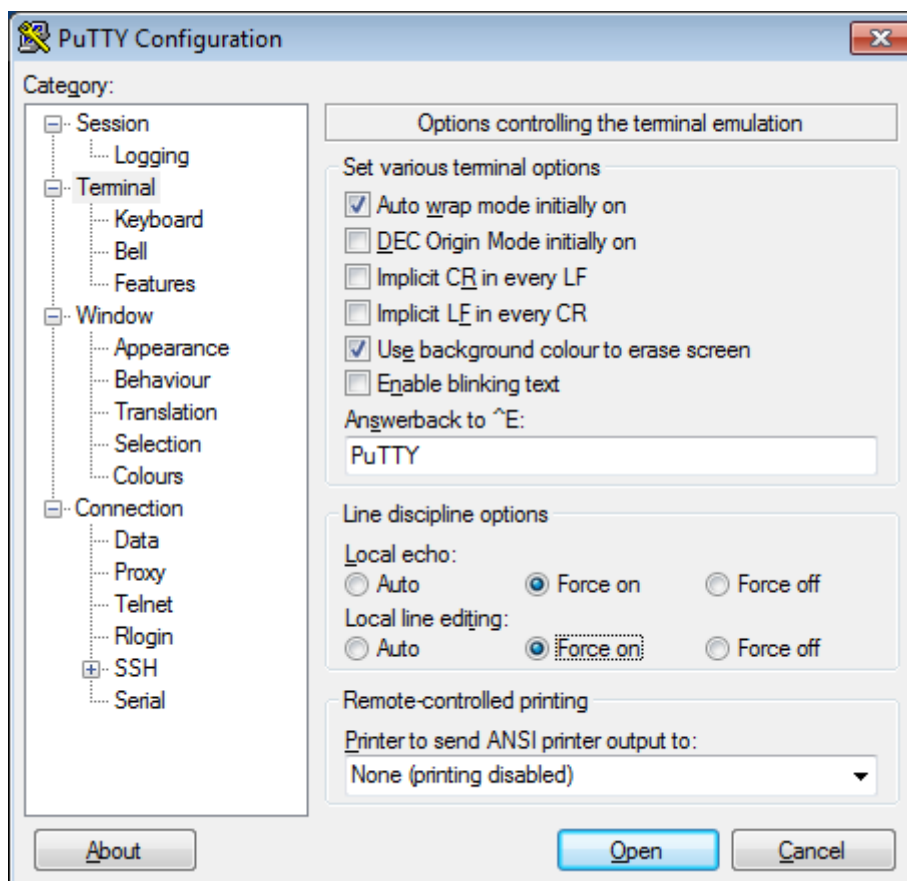
Command (Host to Aphos)	Response (Aphos to Host)	Details
<C,255,c_state 0 1*ZZZZ>	None	This command enables all lights on the RS485 network. This has no bearing on a light that is operating in triac dimming mode
<C,255,c_level 0 55*ZZZZ>	None	Set the light level to 55% on all lights connected to the RS485 interface

Note that when using the broadcast address it is not possible to get a response from a light. In summary the broadcast address can be used to 'Set' parameters in multiple lights at the same time.

A further use for the broadcast address is if a light is being debugged. See the trouble shooting section at the end of this document.

Tips on using Putty

This protocol works best if no line feed or carriage return is sent to the lights at the end of the packet. To do this the automatic line feed and carriage return insertion should be disabled in the terminal. To see the command and response in the Putty window, it is recommended to set the terminal up as shown in the screen grab below.



To send commands down to the light, it is recommended to type the commands into a text file then simply copy them one at a time to the putty window. If you copy a command from the text file, once you select the Putty terminal, you only need to right click on your mouse to send the command.

Command reference

The following commands can be used to communicate with the lights. For this table the lights RS485 address is assumed to be 2.

Command	Type	Packet	Response	Details
c_state	R/W	<C,2,c_state 0 1*ZZZZ	None	Enable light
		<C,2,c_state 0 0*ZZZZ>	None	Disable light
		<C,2,c_state 0*ZZZZ>	<R,2,1*0172>	Light response (enabled in this case)
device_mode	R/W	<C,2,device_mode 0*05F8>	None	Set light to RS485 mode
		<C,2,device_mode 1*05F8>	None	Set light to Triac mode
		<C,2,device_mode 2*05F9>	None	Set light to Ext DC control mode
		<C,2,device_mode*05A7>	<R,2,0*0437>	Light response – in this case the

				light is in RS485 mode (0)
		<C,2,device_mode*05A7>	<R,2,1*0437>	Light response – in this case the light is in triac dimming mode (1)
		<C,2,device_mode*05A8>	<R,2,2*0438>	Light response – in this case the light is in Ext DC control mode(2)
c_level	W	<C,2,c_level 0 80*ZZZZ>	None	Set the light level to 80%. Range is 0 to 100.
dac_scale	R/W	<C,2,dac_scale 100*ZZZZ>	None	This command sets the DAC scale factor to 100%
		<C,2,dac_scale*ZZZZ>	<R,2,100*016A>	Light response, DAC scale is at 100%
status	R	<C,2,status*ZZZZ>	<R,2,0,30.0*026A>	Aphos is in normal mode and temp of lights is 30.0 Celsius
light_current	R	<C,2, light_current*ZZZZ>	<R,2,3.4*0136>	This command returns the current in the LED channel. It is read only. In this example the LED current is 3.4A
s_enable	R/W	<C,2,s_enable 1*ZZZZ>	None	Enable strobe mode
		<C,2,s_enable 0*ZZZZ>	None	Disable strobe mode
		<C,2,s_enable*ZZZZ>	<R,2,1*0171>	Lights response, strobe is enabled
s_trigger	R/W	<C,2,s_trigger 1*ZZZZ>	None	Set strobe to trigger on rising edge
		<C,2,s_trigger 0*ZZZZ>	None	Set strobe to trigger on falling edge
		<C,2,s_trigger*ZZZZ>	<R,2,0*0171>	Light response, strobe is falling edge triggered
S_delay	R/W	<C,2,s_delay 100*ZZZZ>	None	Set light to wait 100usec after trigger is detected to power on the LED's
		<C,2,s_delay*ZZZZ>	<R,2,100*1234>	Light response, strobe delay is 100usec
s_pulse	R/W	<C,2,s_pulse 25000*ZZZZ>	None	Set strobe duration to 25000usec (Max duration is 65535usec)
		<C,2,s_pulse*ZZZZ>	<R,2,25000*123>	Light response, strobe duration is 25000usec
version	R	<C,2,version*ZZZZ>	<R,2,226*1234>	Light response, firmware version is 226
s_no	R	<C,2,s_no*ZZZZ>	<R,2,999*1234>	Light response, light serial number is 999

d_address	R/W	<C,2,d_address 4*ZZZZ>	None	Set lights RS485 address from 2 to 4
		<C,2,d_address*ZZZZ>	<R,2,2*0172>	Light response, RS485 address is 2
temp	R	<C,2,temp*ZZZZ>	<R,2,34.6*034C>	Light response, LED Temperature is 34.6 Celsius
tr_start	R/W	<C,2,tr_start 248*ZZZZ>	None	Set the turn on point for the light in triac dimming mode
		<C,2,tr_start *ZZZZ>	<R,2,248*12AD>	Light response, this is the current turn on point for the light in triac dimming mode
tr_stop	R/W	<C,2,tr_stop 948*ZZZZ>	None	Set the maximum point for the light in triac dimming mode
		<C,2,tr_stop *ZZZZ>	<R,2,948*12AD>	Light response, this is the current maximum point for the light in triac dimming mode
tr_burden	R/W	<C,2,tr_burden 1*0549>	None	Enable the internal 2K ohm burden resistor in the light
		<C,2,tr_burden 0*0548>	None	Disable the internal 2K ohm burden resistor in the light
		<C,2,tr_burden*04F8>	<R,2,0 *014F>	Light response – in this case burden is disabled (0)
		<C,2,tr_burden*04F8>	<R,2,0 *014F>	Light response – in this case burden is enabled (1)
tr_angle	R	<C,2,angle*033A>	<R,2,563 *0896>	Enquire from the light what the current triac position is. In this case the light responds with a position of 563

Note also that where checksum's have been included, they are for example only.

Command line examples

Setting a light for triac dimming mode with the burden resistor enabled [Note 1/](#) [Note 2.](#)

```
<C,2,device_mode 1*ZZZZ> //Set mode to Triac dimming (1)
```

<C,2,tr_burden 1*ZZZZ> //Enable the burden resistor (1)

Setting a light for RS485 mode with burden disabled, light level at 25%, LED's enabled Note 2/Note 3

<C,2,device_mode 0*ZZZZ> //Set mode to RS485

<C,2,tr_burden 0*ZZZZ> //Enable the burden resistor (0)

<C,2,c_level 0 25*ZZZZ> //Light level for channel 0 set to 25%

<C,2,c_state 0 1*ZZZZ> //Enable the LED channel –0 is channel ID, 1 is the enable

Setting a light for RS485 mode with burden disabled, light level at 65%, LED's disabled Note 2

<C,2,device_mode 0*ZZZZ> //Set mode to RS485

<C,2,tr_burden 0*ZZZZ> //Enable the burden resistor (0)

<C,2,c_level 0 65*ZZZZ> //Light level for channel 0 set to 65%

<C,2,c_state 0 0*ZZZZ> //Disable the LED channel -0 is channel ID, 0 is the disable

Setting a light to External DC control mode Note 3

<C,2,device_mode 2*ZZZZ> //Set mode to External DC mode (2)

Setting a light to External DC control mode with strobe enabled Note 4.

<C,2,device_mode 2*ZZZZ> //Set mode to External DC mode (2)

<C,2,s_enable 1*ZZZZ> //Enable the strobe mode (1)

Setting a lights DAC scale factor

<C,2,dac_scale 50*ZZZZ> //Set DAC scale to 50% of max available power

Reading a lights serial number

<C,2,s_no*ZZZZ>

Setting a lights serial number to '123'

<C,2,s_no 123*ZZZZ>

Reading a lights firmware revision

<C,2,version*ZZZZ>

Changing a lights RS485 address to 3

<C,2,d_address 3*ZZZZ> //Change address from 2 to 3

Communicating with a light that is not responding

In the case where you don't know a lights address or the light does not respond to its factory set address, there are two steps you can take to see if the issue is a firmware related one.

Query the light at the two available baud rates

If you know the lights address, but it is not responding on the RS485 interface, you can send a query to the light to return its operating baud rate. Assuming the light is set to a known address we do the following;

Set the Putty (serial terminal) to 9600 baud and send the following command

<C,1,baud_rate*ZZZZ>

If there is no response, then resend the command with the serial terminal set to 57600 baud. If the light is connected, then the response should be as follows;

<R,1,9600*ZZZZ> (for a light operating on 9600 baud at address 1)

Or

<R,1,57600*ZZZZ> (for a light operating on 57600 baud at address 1)

Use the broadcast address to configure the light

If the light still doesn't respond after checking both baud rates, it might be the case that you have the wrong address for the light. In this case you can use the broadcast address which is 255. It is important that the light under test is the only light on the network during this test.

Set the terminal to 57600 baud

<C,255,d_address 1*ZZZZ>

This command will tell the light to set its address to 1. In this example the address 1 is simply an example. Any number from 0 -254 could be used.

Next send the light a query. In this example the light is queried for its firmware version number;

<C,1,version*ZZZZ>

If the light is operating and the RS485 is not physically damaged, then it should respond with the following;

<R,1,225*0192>

The lights firmware is the number to the left of the '*' character. In this case it is 225. The checksum here '0192' is only for example, the actual one might differ.

If the light does not respond to this set of commands. Change the baud rate of the terminal from 57600 to 9600 or vice versa. These are the only two operating baud rates on a light. The default baud rate for Aphos lights is 57600.

Note 1 *Only applicable if the light has been fitted with phase dimming control circuitry*

Note 2 *Only applicable if the light has been fitted with an internal switchable Burden resistor*

Note 3 ***Only applicable if the light has been fitted with external DC control interface***

Note 4 ***Only applicable if the light has been fitted with strobe control interface***

Document Revision History

Edited by	Details	Date Edited	Revision
Fergal Brennan	First release	04-07-2013	0.0
Fergal Brennan	Added in command ref table	17-07-2013	1.0
Fergal Brennan	Added in strobe control section	31-01-2014	2.0
Fergal Brennan	Added in Triac commands and examples	08-02-2014	3.0
Fergal Brennan	Updated document for clarity. Added in section explaining the broadcast command and added in basic guide for trouble shooting an unknown light	17-02-2014	4.0
Fergal Brennan	Added in short section on using Putty with the lights	27-03-2014	5.0
Fergal Brennan	Added in section on using changing light modes Added in notes to examples	03-09-2014	6.0