



TMS LITE SDN. BHD. (Co. No. 671971V)
LED ILLUMINATION SOLUTION PARTNER



TMS-LC-15-4CH-A1
4CH Lighting Controller Unit
(Constant, Strobe and Trigger Mode)

USER MANUAL

Rev 1.12 May 2019

Contents

Hardware	1
Packing List	1
Front Panel	1
General Description	2
Specification	3
Connections	4
Operation	6
Lighting Controller Mode	6
LC-15 Control Mode	7
Display Panel Control Mode	7
Setting up using Display Panel	8
External Control Mode	12
LC-15 Cascade Operation	14
Operation Description	16
External Input Signal	19
Output Signal	20
Cable Selection	21
LEDStudio DLL Documentation	22
DLL Example	29
RS232 Raw Command	35

Revision Notes

Rev	Date/Author	Comment
1.0	KW	First Release
1.1	KW	Added Cycle Strobe and some minor changes
1.2	KW	Minor changes
1.3	KW	Delay times changed to 0-999.9ms
1.4	KW	New casing appearance
1.5	KW	Timing range change to 0-99.99ms (10us step)
1.6	KW	Update DLL documentation part
1.7	KW	Minor Correction
1.8	KW	Minor words correction
1.9	KW	Minor Correction
1.10	KW	Minor Correction
1.11	KW	Added trigger delay info in specification
1.12.	KW	Minor Correction

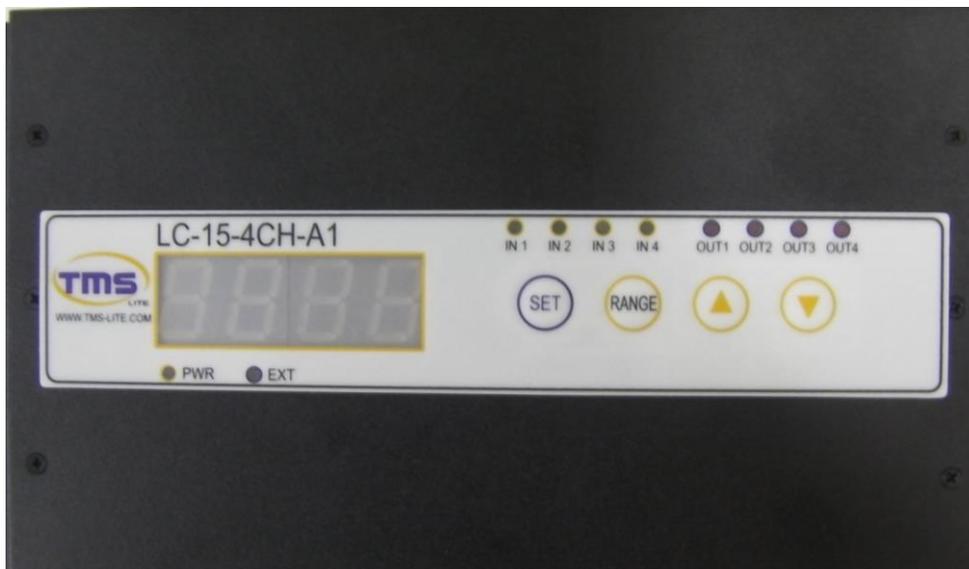
Hardware

Packing List

Please make sure that the following parts are in the packing list:

- TMS-LC-15-4CH-A1 lighting controller unit
- USB/RS232/Ethernet Cable
- Power Supply (Optional)
- 4 LED Lightings (Optional)
- 4 Ext. Cable 1.5 Meter (Optional)

Front Panel



General Description

The LC-15 LED Lighting Controller provides repeatable intensity control of LED lighting for machine vision applications.

The controller has a display panel consisting of a 4-digit seven segment display, four push buttons and LED indicators.

Power Input

- The board power connectors should be connected to power supply of voltage 7-35 VDC
- The lighting power can be supplied with 5/12/24V depend on the lighting power rating

Controller modes

4 modes of operation are provided separately for each channel:

CONSTANT MODE

- 4 Output Channel (CH1 – CH4)
- The output is continuous current
- Range from 0 – 255

STROBE MODE

- 4 Inputs (IN1 – IN4)
- Pull high/low trigger
- Lighting and Output is pulsed once per strobe
- Input signal is used to strobe the respective channel (IN1 strobe CH1, IN2 strobe CH2, etc.)
- Strobe Delay, Strobe Width and Output Delay can be set
 - * 0 – 999.9ms in 0.1ms(100us) steps [Normal version]
 - * 0 – 99.99ms in 0.01ms(10us) steps [10us version]

TRIGGER MODE

- 4 Inputs (IN1 – IN4)
- Rising/Falling edge trigger
- Input signal is used to trigger the respective channel (IN1 trigger CH1, IN2 trigger CH2, etc.)
- Output is turn on when there's an input signal, and turn off when there's no input signal

CYCLE STROBE MODE

- Lighting and Output is pulsed once, after the channel pulse ends, it will then immediately pulse the next channel which is in Cycle Mode
- Strobe Delay, Strobe Width and Output Delay are the same as in Strobe mode

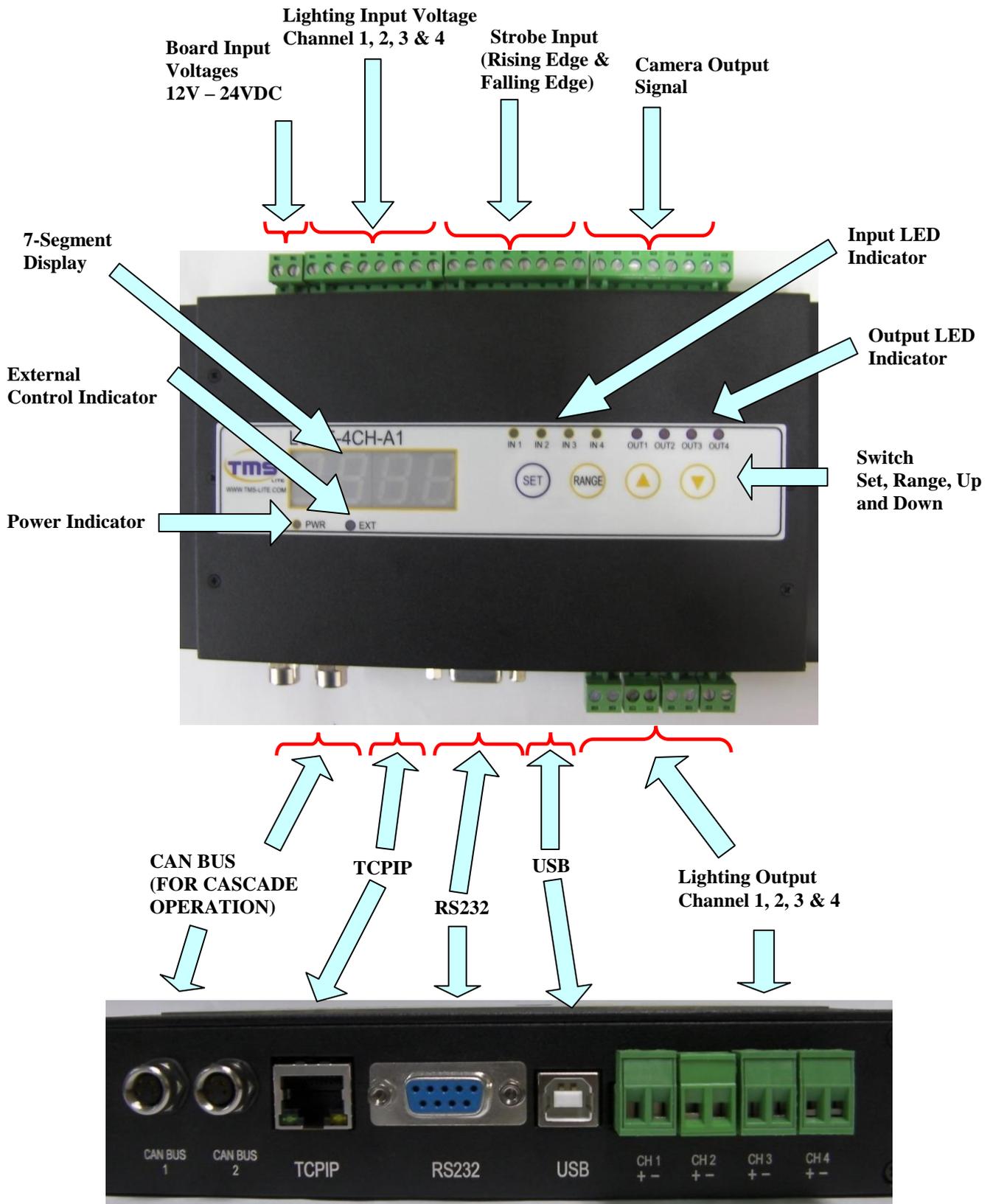
Specification

Lighting Output	Step	Digital 256 steps (0-255)
	No. of lighting output channel	4
	Voltage	5/12/24VDC
	Output current	Max 1A per channel
Input	Trigger output channel	5-24V
	No. of input channel	4
Output	No. of output channel	4
Display	4-digit 7-segment display	Colour: Red
Power supply	Power rating	7-35VDC
	Lighting power supply	5/12/24VDC
Control	Control method	1.Display panel (Buttons) 2. Com Port (RS232) 3. USB 4. TCPIP
Timing	Trigger to Output Delay	< 50us

Connections

All connections are available on screw terminal blocks or connectors.

Screw Terminal Block ID	Function
Power Input (+)	Controller power supply +
Power Input (-)	Controller power supply -
Light Input 1 (+)	Lighting CH1 power supply +
Light Input 1 (-)	Lighting CH1 power supply -
Light Input 2 (+)	Lighting CH2 power supply +
Light Input 2 (-)	Lighting CH2 power supply -
Light Input 3 (+)	Lighting CH3 power supply +
Light Input 3 (-)	Lighting CH3 power supply -
Light Input 4 (+)	Lighting CH4 power supply +
Light Input 5 (-)	Lighting CH4 power supply -
IN1+	Input 1 positive
IN1-	Input 1 negative
IN2+	Input 2 positive
IN2-	Input 2 negative
IN3+	Input 3 positive
IN3-	Input 3 negative
IN4+	Input 4 positive
IN4-	Input 4 negative
OUT1 (+)	Output 1 positive
OUT1 (-)	Output 1 negative
OUT2 (+)	Output 2 positive
OUT2 (-)	Output 2 negative
OUT3 (+)	Output 3 positive
OUT3 (-)	Output 3 negative
OUT4 (+)	Output 4 positive
OUT4 (-)	Output 4 negative
CH1+	Channel 1 positive
CH1-	Channel 1 negative
CH2+	Channel 2 positive
CH2-	Channel 2 negative
CH3+	Channel 3 positive
CH3-	Channel 3 negative
CH4+	Channel 4 positive
CH4-	Channel 4 negative



Panel Label Description

Remark:

Controller power supply GND and lighting power GND are commoned.

Operation

Lighting Controller Mode

Every lighting channel can operate in 3 different modes: Constant, Strobe and Trigger

Constant Mode

Constant mode offers continuous lighting intensity control. Brightness can be adjusted from fully OFF to fully ON in 256 steps.

Strobe Mode

An input strobe signal will turn the light ON and activate the output based on preset delay and duration. Strobing provides high current short interval triggering for applications that require high intensity and high precision.

Trigger Mode

Trigger mode provides most flexibility for variable timing light output. The light output turns ON when an input trigger signal is received and OFF when no input signal is present.

Cycle Strobe Mode

An input strobe signal will turn the light ON and activate the output continuously for channels in Cycle Strobe mode for one cycle per signal. Cycle strobing provides continuous triggering for applications that require continuous strobing and output.

LC-15 Control Mode

The LC-15 lighting controller can be controlled using either the display panel or external control (RS232/USB/TCPIP), but both controls cannot be used at the same time.

The LED labeled “EXT” is an indicator of which control is in use:

- 1) LED turned on – External control (USB/RS232/TCPIP)
- 2) LED turned off – Display Panel control

On power up, the controller is in the previously used control mode. The controller will display “- - -” which is the idle mode and the lighting controller is ready to be used.

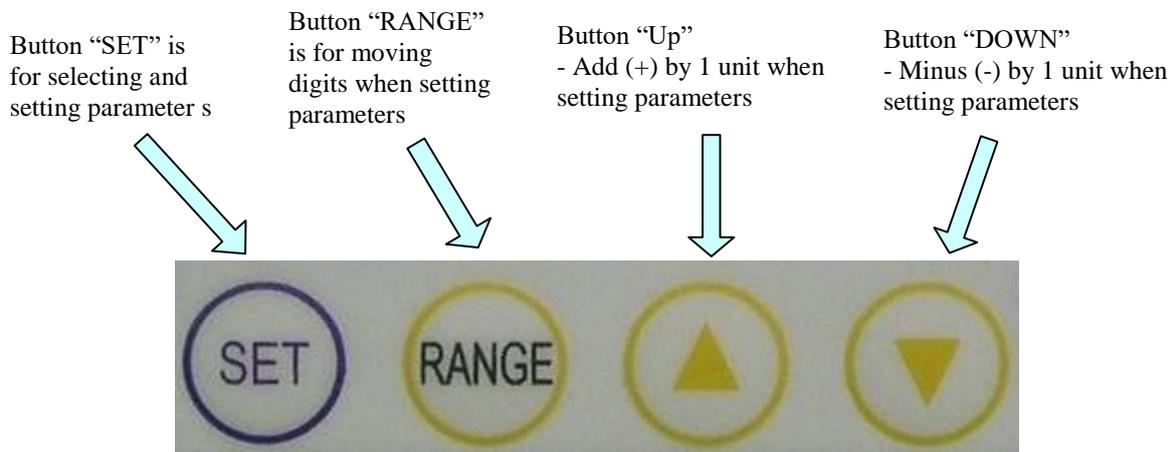
* To switch the control, during IDLE mode, hold down UP and DOWN button for about 1 second and observe the change on the LED indicator.

****REMARK****

- Stop all external input trigger signals when setting up the controller. Failure to do so may cause the controller to malfunction.

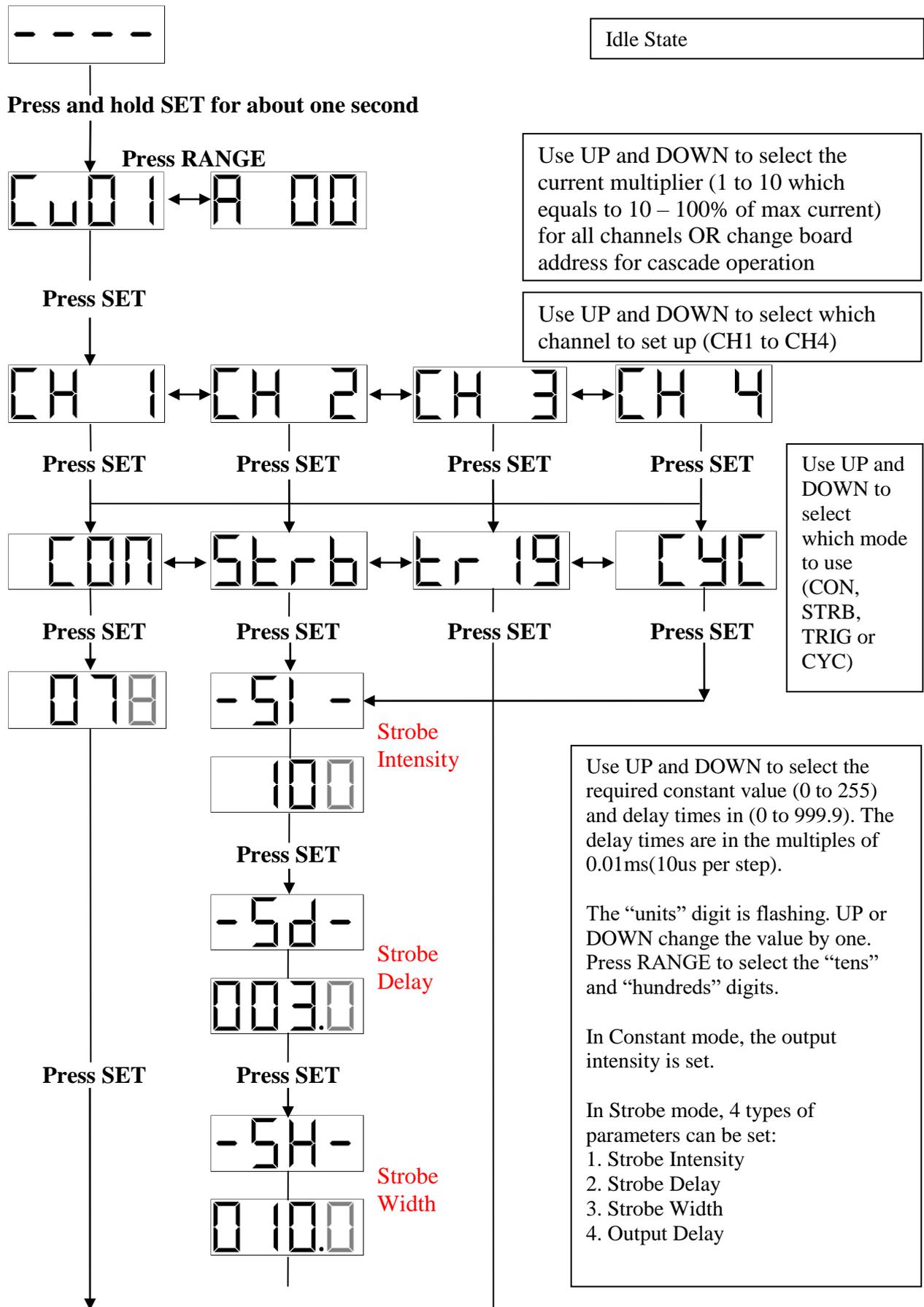
Display Panel Control Mode

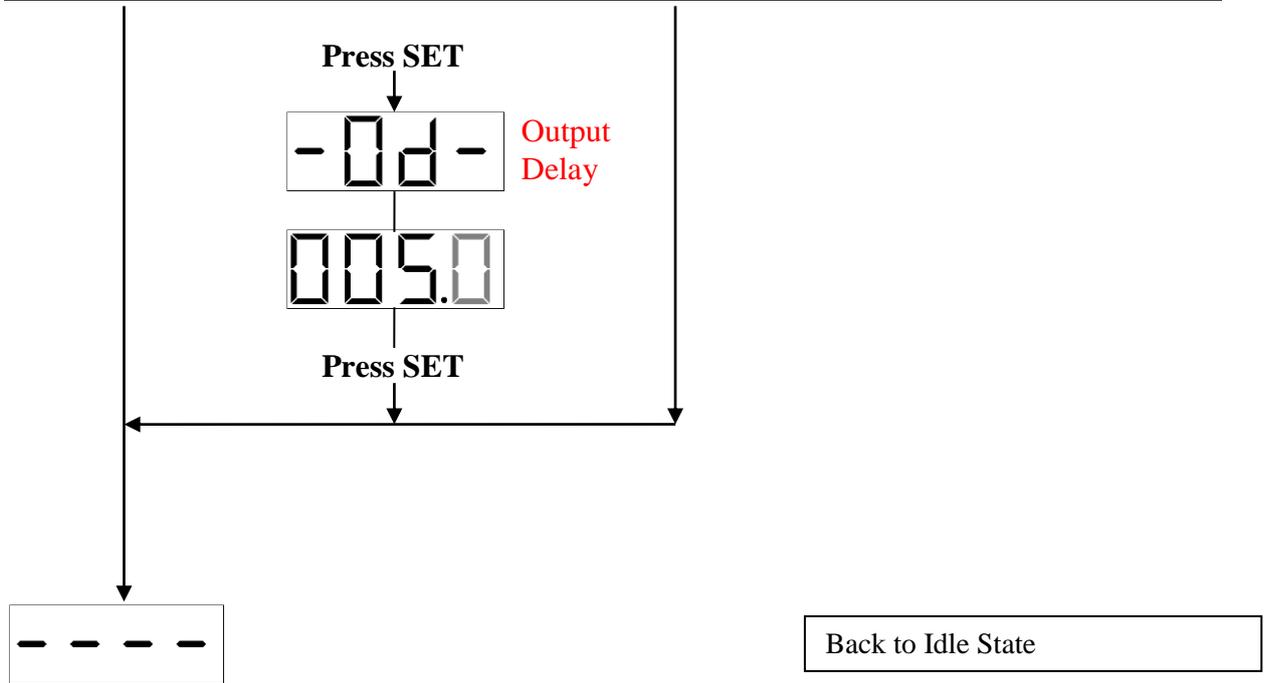
In Display Panel Control Mode, user will control the controller by using the 4 Buttons at the Front Panel (Set, Range, Up and Down) for setting Channel, Lighting intensity and trigger pulse width.



Buttons

Setting up using Display Panel

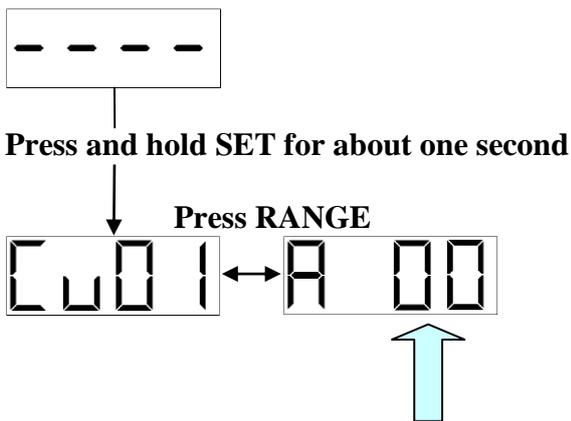




Remark:

Setting the intensity using display panel will set the intensity for both constant and strobe mode. To set different intensity for both modes, user can use the application LEDPro to set using RS232/USB/TCPIP communication.

During Strobe or Trigger mode, the function of the respective channel can be manually tested. The channel is strobe/trigger once when the RANGE button is pressed. This only works when the display is in IDLE mode.

Setting up Board Address for Cascade Operation

* Use UP and DOWN button to select board address

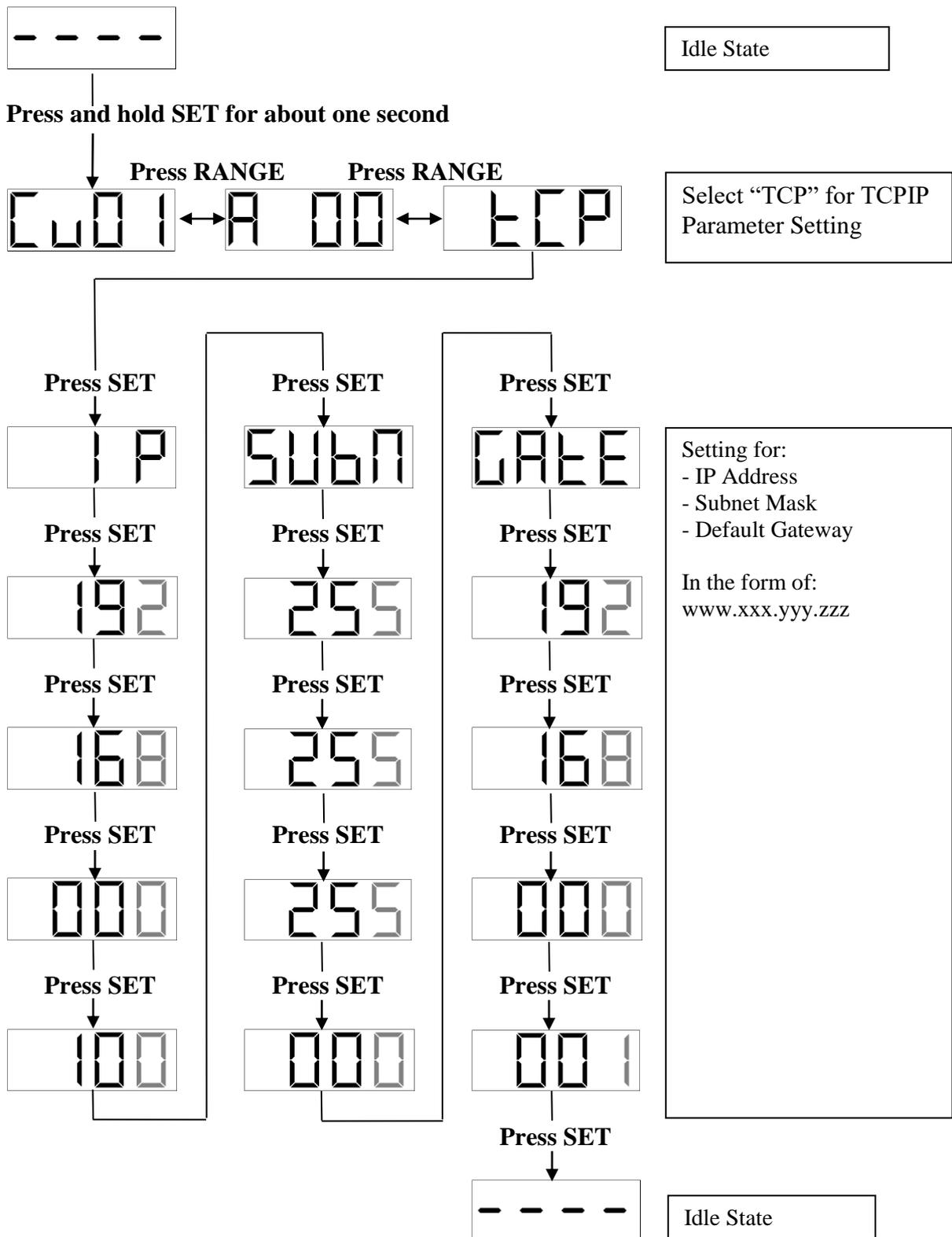
Board address must be set for controllers used in cascade operation.

Master controller is set as address 00

Slave controller can be set start from address 01

** REMARK: Address should be set to 00 if use as individual controller

Setting up TCPIP parameters



***REMARK:**

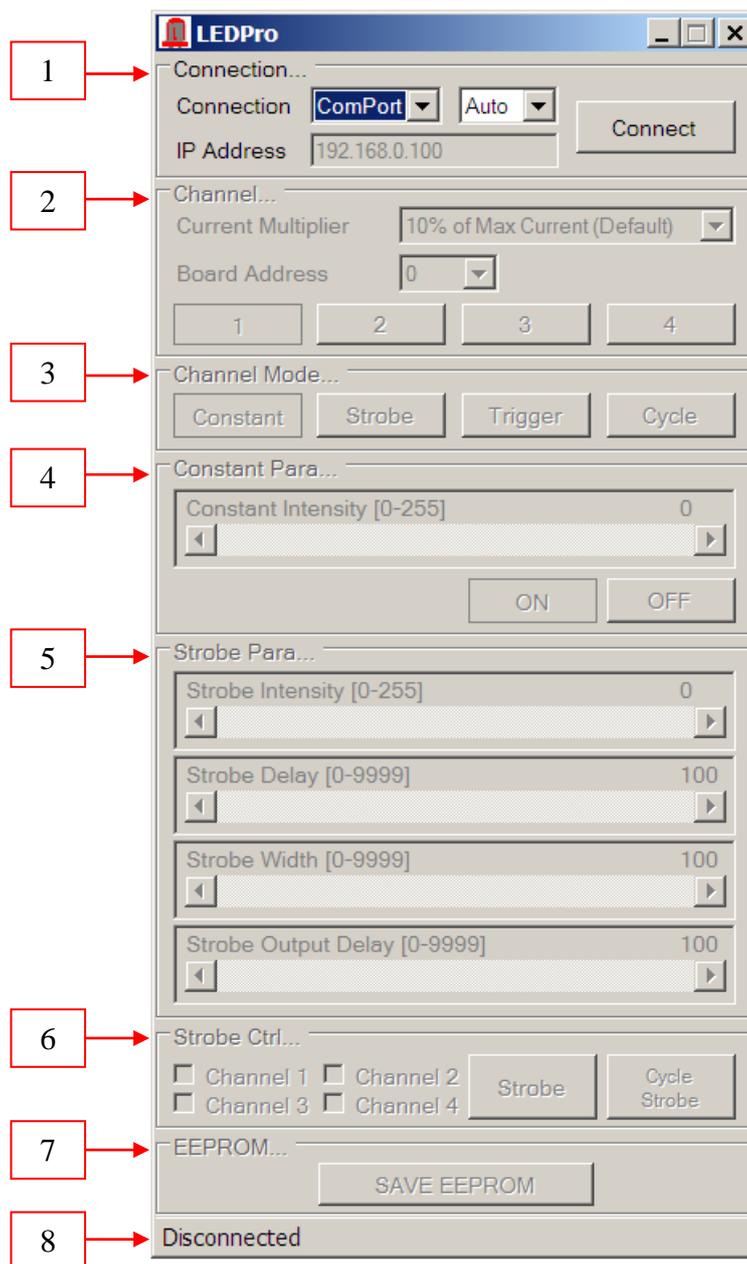
- Default IP: 192.168.0.100
- Default Subnet Mask: 255.255.255.0
- Default Gateway: 192.168.0.1
- Port Number = 80

External Control Mode

In USB/RS232/TCPIP Control Mode, the EXT LED on the front panel will be on “ON” status. Connect the RS232 Cable or USB Cable (does not need USB Driver Installation) or TCPIP Cable. The controller can be controlled using the LEDPro Software.

LEDPro

1. Unzip the LEDStudio_Pro files to C drive – C:\LEDStudio\
2. Make sure that LEDStudio.dll and RS232.dll files should be in the folder C:\LEDStudio\LEDLib
3. Run LEDPro from C:\LEDStudio\LEDPro



Description**1. Connection**

- Select the type of connection to be used - ComPort(RS232), USB or TCPIP
- Select the port number where the controller is connected (Auto or 1-30)
- Set device's IP Address to connect for TCPIP connection
- Connect/Disconnect

2. Channel

- Select current multiplier for all channels (10%, 20%, ... , 100% of max current)
- Select board address for cascade operation (0 to 3)
- Select the channel to change the parameter (Channel 1 to 4)

3. Channel Mode

- Select mode for channel
- Constant Mode, Strobe Mode, Trigger Mode, Cycle Strobe Mode

4. Constant Parameter

- Change the constant intensity
- 0 to 255

5. Strobe Parameter

- Change strobing parameters
- Strobe Intensity (0 – 255)
- Strobe Delay (0 – 99.99ms)
- Strobe Width (0 – 99.99ms)
- Strobe Output Delay (0 – 99.99ms)

6. Strobe Control

- Strobe
- Select the channel to strobe
- Channel 1 to 4
- Strobe the selected channel(s)
- Cycle Strobe – strobe once for all channels in Cycle Strobe mode continuously

7. EEPROM

- Save parameters into the EEPROM

8. Connection Status

- Show the connection status

LC-15 Cascade Operation

The LC-15 Lighting Controller can be used in cascade mode when there are more than 4 lighting output channel to be controlled at the same time.

The number of LC-15 controller can be connected in cascade is up to 16 where the address of the controller is pre-set using the display panel (00 to 15).

To use LC-15 in cascade,

- The address of the master controller must be set to 0 and communicate with the PC using RS232, USB or TCPIP
- The slave controller address can be set from 1 to 15 and is connected to the master controller using CAN BUS

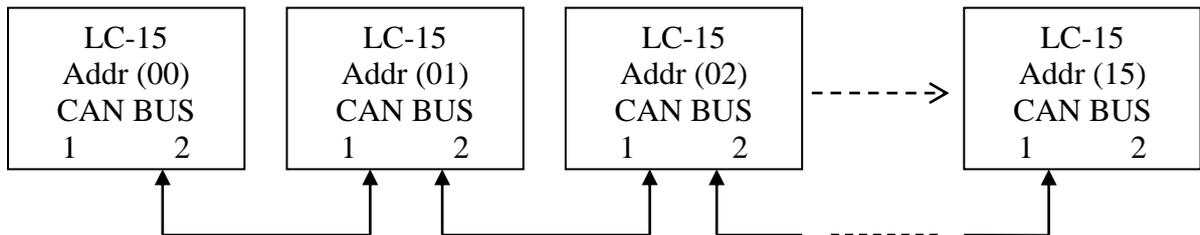
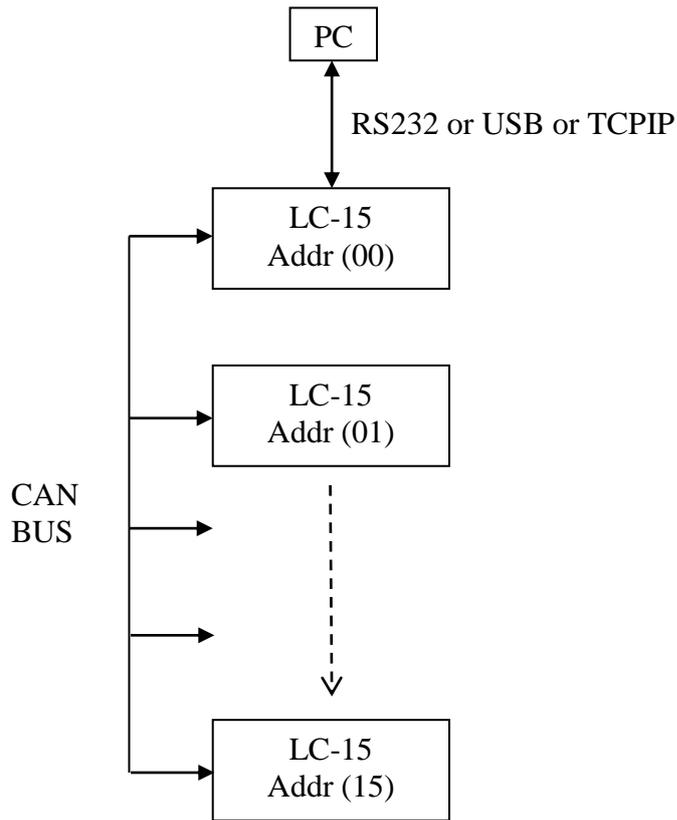
Connection:

- Connect one side of the CAN Bus cable to any of the CAN BUS socket of the main LC-15 controller (address 00) and the other side to CAN BUS Socket of other slave controller (address 01 – 15)
- To control more controllers, just connect them in loop



CAN Bus Connector Socket

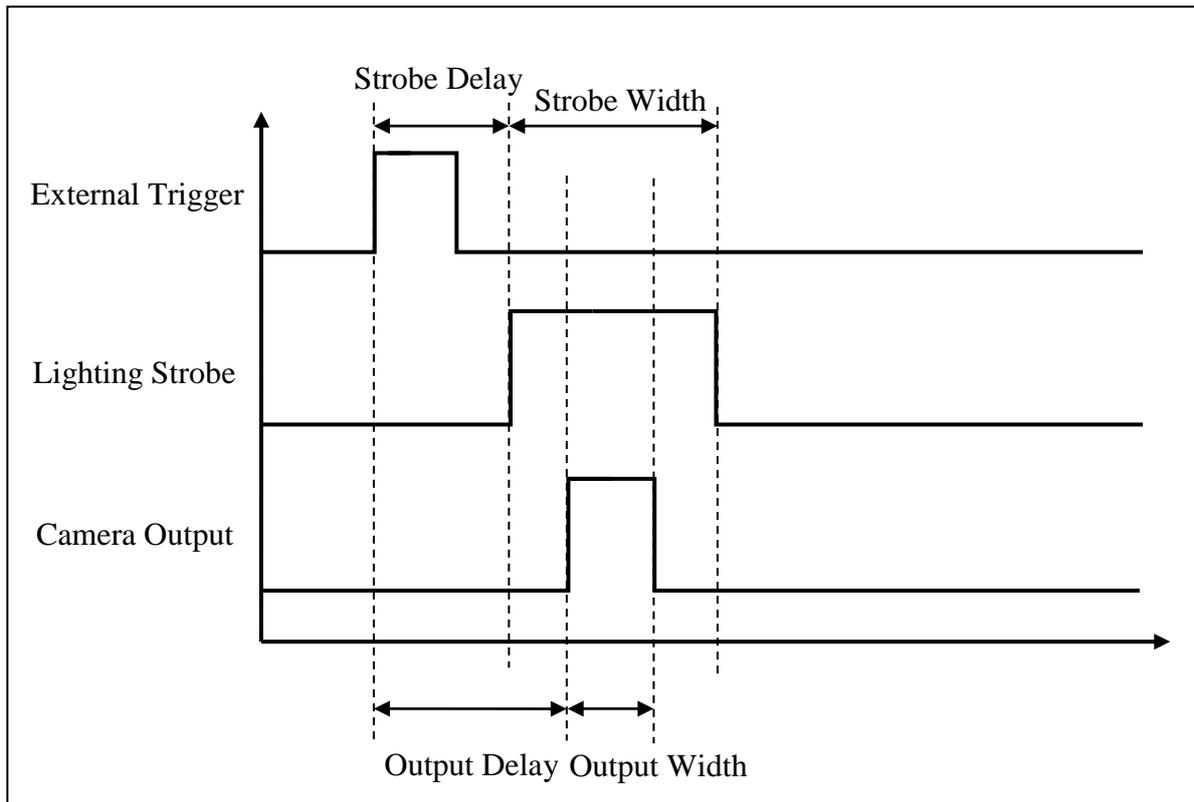
CAN BUS multi-drop connection:



Operation Description

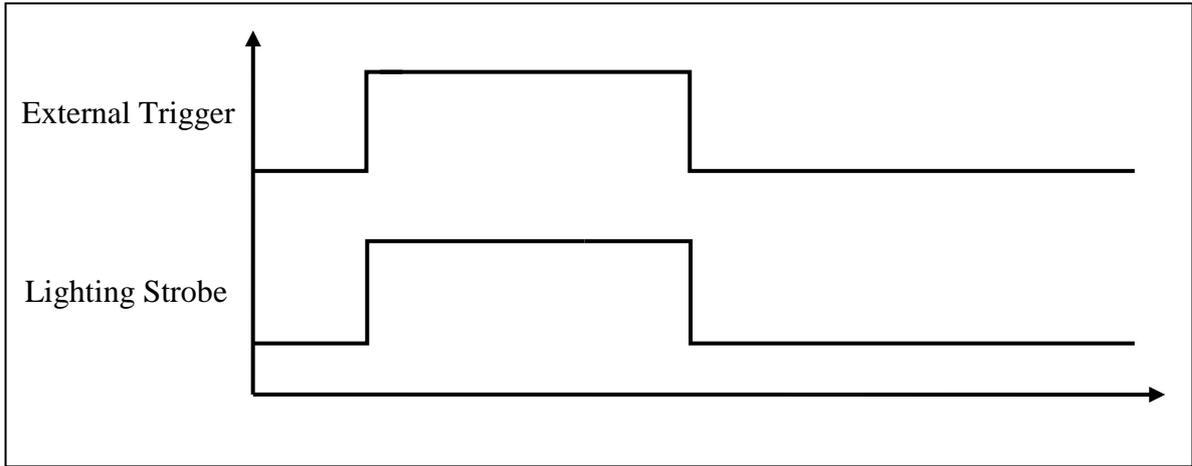
Timing diagram

1) Strobe Mode

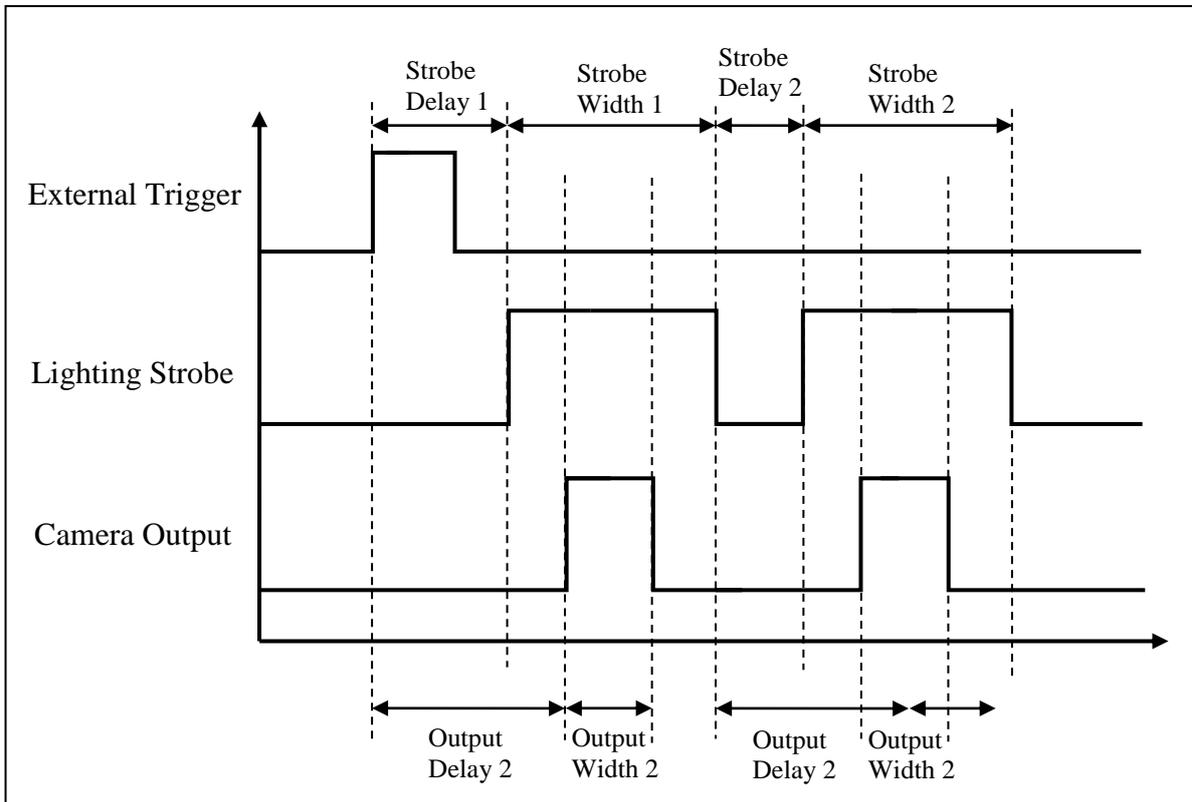
**Remark:**

The Output Width is fixed as 10ms, user can modify the other 3 parameters (Strobe Delay, Strobe Width and Output Delay)

2) Trigger Mode



3) Cycle Strobe Mode



Remark:

Using External control (RS232/USB/TCPIP) to strobe the lighting output may have delays due to the time used during communication.

Constant mode intensity to current table

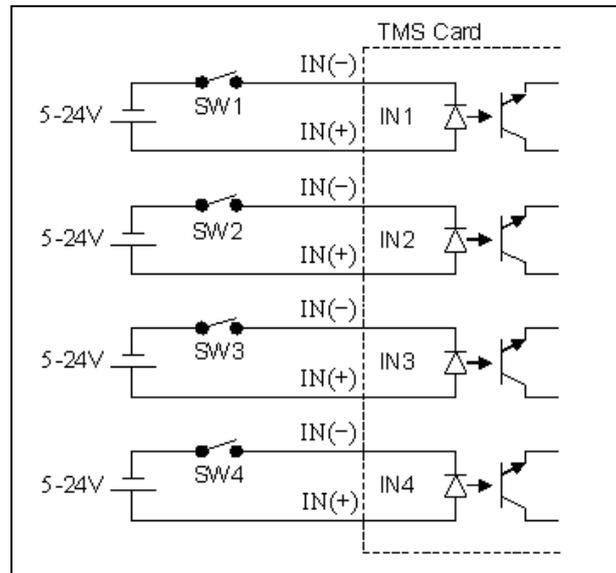
Intensity	Current (mA) (Approximate)									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
0	0	0	0	0	0	0	0	0	0	0
10	4	8	12	16	20	24	27	31	35	39
20	8	16	24	31	39	47	55	63	71	78
30	12	24	35	47	59	71	82	94	106	118
40	16	31	47	63	78	94	110	125	141	157
50	20	39	59	78	98	118	137	157	176	196
60	24	47	71	94	118	141	165	188	212	235
70	27	55	82	110	137	165	192	220	247	275
80	31	63	94	125	157	188	220	251	282	314
90	35	71	106	141	176	212	247	282	318	353
100	39	78	118	157	196	235	275	314	353	392
110	43	86	129	173	216	259	302	345	388	431
120	47	94	141	188	235	282	329	376	424	471
130	51	102	153	204	255	306	357	408	459	510
140	55	110	165	220	275	329	384	439	494	549
150	59	118	176	235	294	353	412	471	529	588
160	63	125	188	251	314	376	439	502	565	627
170	67	133	200	267	333	400	467	533	600	667
180	71	141	212	282	353	424	494	565	635	706
190	75	149	224	298	373	447	522	596	671	745
200	78	157	235	314	392	471	549	627	706	784
210	82	165	247	329	412	494	576	659	741	824
220	86	173	259	345	431	518	604	690	776	863
230	90	180	271	361	451	541	631	722	812	902
240	94	188	282	376	471	565	659	753	847	941
250	98	196	294	392	490	588	686	784	882	980
255	100	200	300	400	500	600	700	800	900	1000

External Input Signal

External input signals are optional and can be used to turn on the lightings for a period based on the signals (trigger mode). The input signals for a particular channel has no effect if the channel is operating in Constant Mod or when the previous strobe command is still running.

IN1, IN2, IN3 and IN4 are input signals for Channel 1, Channel 2, Channel 3 and Channel 4 respectively. IN(+) is the common positive input and should be connected to 5 - 24V DC and IN(-) is the common negative input and should be connected to GND as shown in the following figure.

The trigger signal will be acknowledged by the controller on the falling edge or rising edge of the trigger input signal.

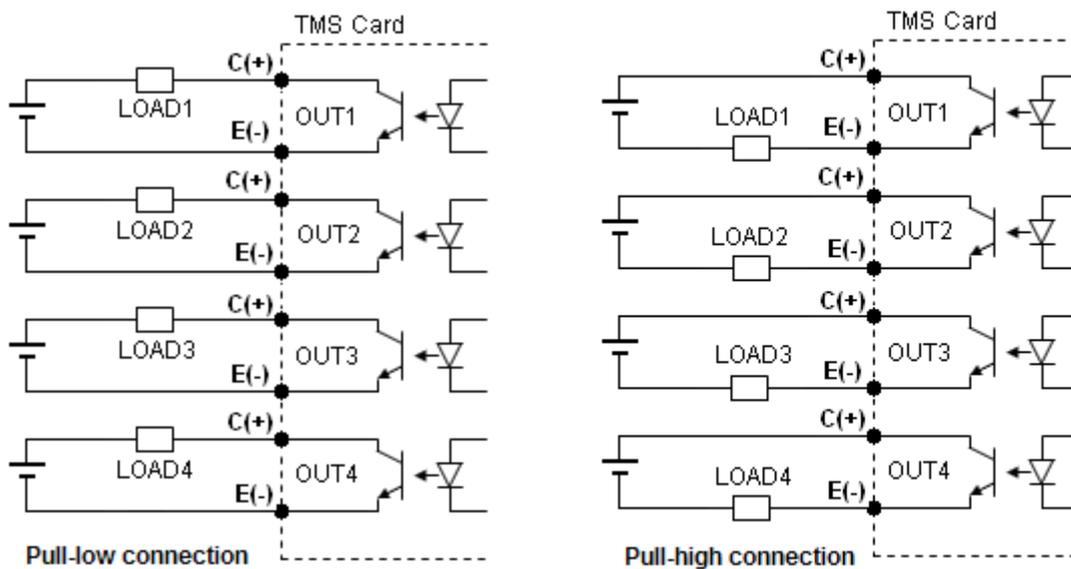


Output Signal

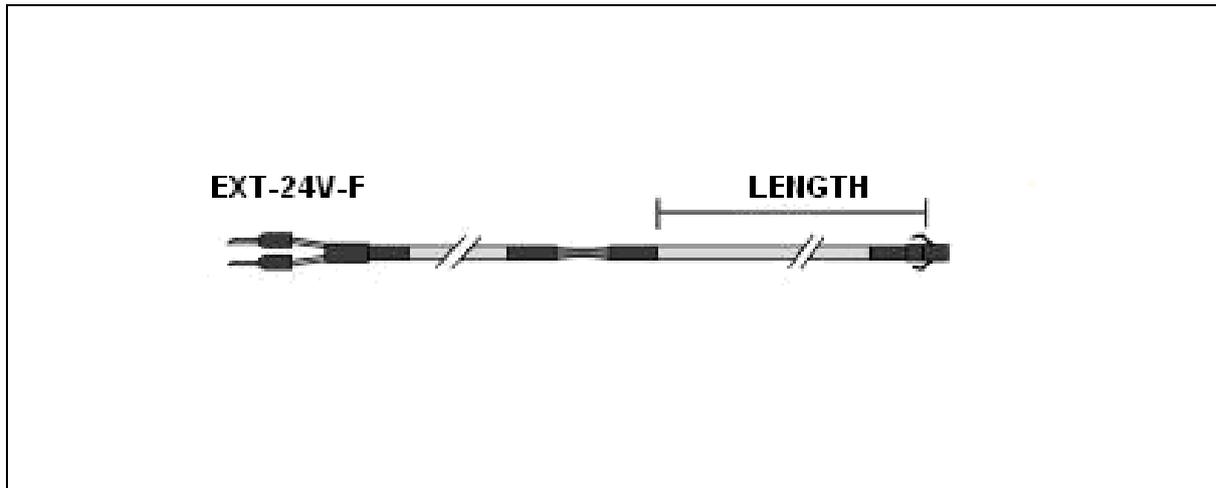
The Output signals are optional and are used to trigger external devices such as camera. The Output signals are only available in Strobe Mode.

The output pulse is triggered after a delay time defined by Strobe Output Delay.

OUT1, OUT2, OUT3, OUT4 are output signal for Channel 1, Channel 2, Channel 3 and Channel 4 respectively. C(+) is Collector positive and E(-) is Emitter negative for the output. The output connections (Pull High or Pull Low Signal) are shown in the following figures.

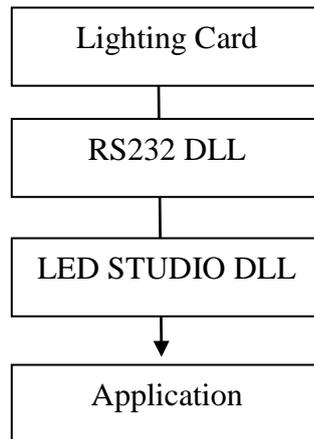


Cable Selection



MODEL	POWER	LENGTH	APPLICATIONS
EXT-24V-F	24V	1M 2M 3M 4M 5M	<ul style="list-style-type: none"> Used to connect 24V light to USB or RS232-4CH-P1 / KL-4000 / STB-1CH or 2CH-P1 / ANG-1CH-P1 / LC Series

LEDStudio DLL Documentation



DLL Location Path

1. RS232 DLL must be located at C:\LEDStudio\LEDLib\RS232.dll
2. LEDStudio DLL must be located at C:\LEDStudio\LEDLib\LEDStudio.dll

Function List

1. function LE_ComportConnect(var ComPort: Byte): Byte; stdcall;
2. function LE_ComportDisconnect(Comport: Byte): Byte; stdcall;
3. function LE_USBConnect(var ProductName: PChar): Byte; stdcall;
4. function LE_USBDISCONNECT: Byte; stdcall;
5. function LE_SetMultiplier(sMul: Byte): Byte; stdcall;
6. function LE_SetCHMode(sCH, sMode:Byte): Byte; stdcall;
7. function LE_SetConstInt(sInt: Byte): Byte; stdcall;
8. function LE_SetCONSTOnOff(sOn: Byte): Byte; stdcall;
9. function LE_SetStrobeInt(sInt: Byte): Byte; stdcall;
10. function LE_SetStrobeDelay(sDelay: Word): byte; stdcall;
11. function LE_SetStrobeWidth(sWidth: Word): Byte; stdcall;
12. function LE_SetStrobeOutDelay(sOutDelay: Word): Byte; stdcall;
13. function LE_STROBE(CH1, CH2, CH3, CH4: Boolean): Byte; stdcall;
14. function LE_GETErrMsg(ErrCode: Byte):Pchar ; stdcall;
15. function LE_EEPROM: Byte; stdcall;
16. function LE_SelectAdd(sAdd:Byte): Byte; stdcall;
17. function LE_TCPIPConnect(sIP: PChar): Byte; stdcall;
18. function LE_TCPIPDISCONNECT: Byte; stdcall;

- 1. function LE_ComportConnect(var ComPort: Byte): Byte; stdcall;**
Connect the lighting device via serial com port defined by 'Comport'
If Comport = 0, LEDStudio will auto detect and connect with Lighting device and will return the comport number connected
Parameter:-
ComPort: Byte (by reference)
- Comport number
Return:-
Error Code: Byte

- 2. function LE_ComportDisConnect(Comport: Byte): Byte; stdcall;**
Disconnect the serial com port with lighting device
Parameter:-
Comport: Byte (by value)
- Comport number
Return:-
Error Code: Byte

- 3. function LE_USBConnect(var ProductName: PChar): Byte; stdcall;**
Connect the lighting device via USB port defined.
LEDStudio will auto connect USB port with Lighting device and will return product name of the lighting device
Parameter:-
ProductName: String (by reference)
- Product Name
Return:-
Error Code: Byte

- 4. function LE_USBDDisconnect: Byte; stdcall;**
Disconnect the lighting device from USB Port
Parameter:-
None
Return:-
Error Code: Byte

5. function LE_SetMultiplier(sMul: Byte): Byte; stdcall;

Set the current multiplier of lighting device

Parameter:-

sMul: Byte (by value)

- Current Multiplier

Range = 1 – 10

1 = 10% of Max Current (Default)

2 = 20% of Max Current

3 = 30% of Max Current

9 = 90% of Max Current

10 = 100% of Max Current

Return:-

Error Code: Byte

6. function LE_SetCHMode(sCH, sMode:Byte): Byte; stdcall;

Set the mode of Channel defined by 'sCH' and 'sMode'.

Parameter:-

sCH: Byte (by value)

- Channel number

Range = 1 – 4

sMode: Byte (by value)

- Channel mode

Range =

Mode = 0 => Constant

Mode = 1 => Strobe

Mode = 2 => Trigger

Mode = 4 => Cycle Strobe

Return:-

Error Code: Byte

7. function LE_SetConstInt(sInt: Byte): Byte; stdcall;

Set the Constant Intensity defined by 'sInt' for selected channel

Parameter:-

sInt: Byte (by value)

- Constant Intensity

Range = 0-255

Return:-

Error Code: Byte

- 8. function LE_SetCONSTOnOff(sOn: Byte): Byte; stdcall;**
Set ON or OFF for output of lighting device for selected channel
Parameter:-
sOn: Byte (by value)
- Constant Output ON/OFF
Range:
On = 1
Off = 0
Return:-
Error Code: Byte
- 9. function LE_SetStrobeInt(sInt: Byte): Byte; stdcall;**
Set the Strobe Intensity defined by 'sInt' for selected channel
Parameter:-
sInt: Byte (by value)
- Strobe Intensity
Range = 0-255
Return:-
Error Code: Byte
- 10. function LE_SetStrobeDelay(sDelay: Word): Byte; stdcall;**
Set the Strobe Delay defined by 'sDelay' for selected channel
Parameter:-
sDelay: Word (by value)
- Strobe Delay
Range = 0-9999
(Normal version = 999.9ms; 10us version = 99.99ms)
Return:-
Error Code: Byte
- 11. function LE_SetStrobeWidth(sWidth: Word): Byte; stdcall;**
Set the Strobe Pulse Width defined by 'sWidth' for selected channel.
Parameter:-
sWidth: Word (by value)
- Strobe Width
Range = 0-9999
(Normal version = 999.9ms; 10us version = 99.99ms)
Return:-
Error Code: Byte

12. function LE_SetStrobeOutDelay(sOutDelay: Word): Byte; stdcall;

Set the Strobe Out Delay defined by 'sOutDelay' for selected channel.

Parameter:-

sOutDelay: Word (by value)

- Strobe Output Delay

Range = 0-9999

(Normal version = 999.9ms; 10us version = 99.99ms)

Return:-

Error Code: Byte

13. function LE_STROBE(CH1, CH2, CH3, CH4: Boolean): Byte; stdcall;

To Trigger the Strobe defined by CH1,CH2,CH3,CH4 with boolean value

In LEDBasic, use this function to TRIGGER

Parameter:-

CH1, CH2, CH3, CH4: Boolean (by value)

- Channel to Strobe

Range:

1 = Trigger

0 = No trigger

Return:-

Error Code: Byte

14. function LE_GETErrMsg(ErrCode: Byte):Pchar ; stdcall;

Get error message

Parameter:-

ErrCode: Byte (by value)

- Error Code retrieved from called function

Return:-

Error Message: String

15. function LE_EEPROM: Byte; stdcall;

To save lighting parameters into the EEPROM

Parameter:-

None

Return:-

Error Code: Byte

16. function LE_SelectAdd(sAdd:Byte): Byte; stdcall;

Select controller address

Parameter:-

sAdd: Byte (by value)

- Controller address

Range = 0-15

Return:-

Error Code: Byte

17. function LE_TCPIPConnect(sIP: PChar): Byte; stdcall;

Connect the controller with TCPIP

Parameter:-

sIP: String (by value)

- Controller IP Address, ie: "192.168.0.100"

Return:-

Error Code: Byte

18. function LE_TCPIPDisconnect: Byte; stdcall;

Disconnect the controller from TCPIP

Parameter:-

None

Return:-

Error Code: Byte

Error Code

Every function will return an error code in byte value.

If return value = 0, no error excuted. User also can get error description via **function LE_GETErrMsg(ErrCode: Byte):Pchar ;**

some pascal example:

```
var
```

```
    ErrCode: Byte;
```

```
begin
```

```
    ErrCode:= LE_SetStrobeInt(255);
```

```
    If ErrCode <> 0 then ShowMessage(LE_GetErrMsg(ErrCode));
```

```
End;
```

Error Code List

Error Code	Description	
0	Success No Error Excute	
100	'Err100 - GET_VERSION_ERROR'	Ensure all integration hardware, cable and lighting device are in proper condition. And also in the correct data range that has been defined. Kindly contact the supplier if the error codes occur.
105	'Err105 - COMPORT_CONNECT_ERROR'	
106	'Err106 - NO_COM_DEVICE_DETECTED'	
107	'Err107 - CONNECT_SELECTED_COM_FAIL'	
110	'Err110 - COMPORT_DISCONNECT_ERROR'	
115	'Err115 - USB_CONNECT_ERROR'	
116	'Err116 - NO_USB_DEVICE_DETECTED'	
120	'Err120 - USB_DISCONNECT_ERROR'	
122	'Err122 - TCPIP_CONNECT_ERROR'	
123	'Err123 - NO_TCPIP_DEVICE_DETECTED'	
124	'Err123 - TCPIP_DISCONNECT_ERROR'	
130	'Err130 - SET_CHMODE_ERROR';	
131	'Err131 - INVAILID_CH_NUMBER'	
132	'Err132 - INVAILID_MODE_NUMBER'	
135	'Err135 - SET_MULTUPIER_ERROR'	
140	'Err140 - SET_CONST_INT_ERROR'	
145	'Err145 - SET_CONST_ONOFF_ERROR'	
150	'Err150 - SET_STROBE_INT_ERROR'	
155	'Err155 - SET_STROBE_DELAY_ERROR'	
160	'Err160 - SET_STROBE_WIDTH_ERROR'	
165	'Err165 - SET_STROBE_ODELAY_ERROR'	
170	'Err170 - LE_STROBE_ERROR'	
200	'Err200 - SEND_MSG_ERROR'	
201	'Err201 - WRITE_MSG_ERROR'	
202	'Err202 - READ_MSG_ERROR'	
203	'Err203 - READ_MSG_CHECKSUM_ERROR'	
242	'Err242 - COMMAND_ERROR'	lighting device received invalid command
248	'Err248 - DATA_ERROR'	lighting device received invalid data (Data Not In Range)
255	'Err255 - COMMUNICATION_ERROR'	lighting device verify invalid checksum data

DLL Example

EXAMPLE (C#):

DLL IMPORT:

```
const string LEDStudiosDll = "C:\\LEDStudio\\LEDLib\\LEDStudio.dll";
[DllImport(LEDStudiosDll)]
public static extern byte LE_ComportConnect(ref byte ComPort);
[DllImport(LEDStudiosDll)]
public static extern byte LE_ComportDisconnect(byte Comport);
[DllImport(LEDStudiosDll)]
public static extern byte LE_USBConnect(string ProductName);
[DllImport(LEDStudiosDll)]
public static extern byte LE_USBDisconnect();
[DllImport(LEDStudiosDll)]
public static extern byte LE_SetMultiplier(byte sMul);
[DllImport(LEDStudiosDll)]
public static extern byte LE_SetCHMode(byte sCH, byte sMode);
[DllImport(LEDStudiosDll)]
public static extern byte LE_SetConstInt(byte sInt);
```

CONNECT:

```
byte portnum = 1; // COM1
string productname;
byte err_code;

if (COM_Connect) // COMPORT
{
    if (connected == false)
        err_code = LE_ComportConnect(ref portnum);
    else
        err_code = LE_ComportDisconnect(portnum);
}
else // USB
{
    if (connected == false)
        err_code = LE_USBConnect(productname);
    else
        err_code = LE_USBDisconnect();
}
```

ERROR CHECK:

```
if (err_code != 0)
```

```
MessageBox.Show(LE_GETErrMsg(err_code), "Error");
```

SET MULTIPLIER:

```
byte multiplier = 5; // 50%  
err_code = LE_SetMultiplier(multiplier);
```

SET CHANNEL & MODE:

```
byte channel = 1; // Channel 1  
byte channel_mode = 0; // Constant mode  
err_code = LE_SetCHMode(channel, channel_mode);
```

CONSTANT:

```
byte const_int = 100; // Intensity 100  
err_code = LE_SetConstInt(const_int);
```

```
byte on_off = 1; // On  
err_code = LE_SetCONSTOnOff (on_off);
```

STROBE/TRIGGER:

```
UInt16 strobe_width = 1000; // 1ms strobe width  
err_code = LE_SetStrobeWidth(strobe_width);
```

```
err_code = LE_STROBE(true, false, false, false) // strobe channel 1
```

AUTO STROBE:

```
UInt16 auto_uptime = 250; // 250ms up time  
UInt16 auto_dntime = 400; // 400ms down time  
err_code = LE_SetAutoStrobeUpTime(auto_uptime)  
err_code = LE_SetAutoStrobeDnTime(auto_dntime)
```

```
err_code = LE_AUTOSTROBE(true, false, false, false) // turn on channel 1 auto strobe (set  
false = turn off)
```

SAVE PARAMTERS IN EEPROM:

```
err_code = LE_EEPROM();
```

EXAMPLE (VC++):**DLL IMPORT:**

```

HINSTANCE lib = LoadLibraryA("C:\\LEDStudio\\LEDLib\\LEDStudio.dll");
typedef byte (__stdcall * LE_ComportConnect)(int& ComPort);
LE_ComportConnect ComportConnect =
reinterpret_cast<LE_ComportConnect>(GetProcAddress(lib, "LE_ComportConnect"));
typedef byte (__stdcall * LE_ComportDisConnect)(byte ComPort);
LE_ComportDisConnect ComportDisConnect =
reinterpret_cast<LE_ComportDisConnect>(GetProcAddress(lib,
"LE_ComportDisConnect"));
typedef byte (__stdcall * LE_USBConnect)(char* ProductName);
LE_USBConnect USBConnect = reinterpret_cast<LE_USBConnect>(GetProcAddress(lib,
"LE_USBConnect"));
typedef byte (__stdcall * LE_USBDDisconnect)();
LE_USBDDisconnect USBDisconnect =
reinterpret_cast<LE_USBDDisconnect>(GetProcAddress(lib, "LE_USBDDisconnect"));
typedef byte (__stdcall * LE_SetMultiplier)(byte sMul);
LE_SetMultiplier SetMultiplier = reinterpret_cast<LE_SetMultiplier>(GetProcAddress(lib,
"LE_SetMultiplier"));
typedef byte (__stdcall * LE_SetCHMode)(int sCH, int sMode);
LE_SetCHMode SetCHMode = reinterpret_cast<LE_SetCHMode>(GetProcAddress(lib,
"LE_SetCHMode"));
typedef byte (__stdcall * LE_SetConstInt)(int sInt);
LE_SetConstInt SetConstInt = reinterpret_cast<LE_SetConstInt>(GetProcAddress(lib,
"LE_SetConstInt"));

```

CONNECT:

```

int cp = 1; // port number
int& comport = cp; // variable by reference
String ^productname = "";

if (COM_Connect) // COMPORT
{
    if (connected == false)
        err_code = ComportConnect(comport);
    else
        err_code = ComportDisConnect(portnum);
}
else // USB
{
    if (connected == false)
        err_code = USBConnect(productname);
    else
        err_code = USBDDisconnect();
}

```

ERROR CHECK:

```
String ^err_msg;  
if (err_code != 0) err_msg = GETErrMsg(err_code);
```

SET MULTIPLIER:

```
byte multiplier = 5;  
err_code = SetMultiplier(multiplier);
```

SET CHANNEL & MODE:

```
int channel = 1; // CH1  
int mode = 0; // constant mode  
err_code = SetCHMode(channel, mode);
```

CONSTANT:

```
byte intensity = 100;  
err_code = SetConstInt(intensity);  
byte on_off = 1; // On  
err_code = SetCONSTOnOff (on_off);
```

STROBE/TRIGGER:

```
short strobe_width = 1000; // 1ms strobe width  
err_code = SetStrobeWidth(strobe_width);  
  
err_code = STROBE(true, false, false, false); // strobe channel 1
```

AUTO STROBE:

```
short auto_uptime = 250; // 250ms up time  
short auto_dntime = 400; // 400ms down time  
err_code = SetAutoStrobeUpTime(auto_uptime);  
err_code = SetAutoStrobeDnTime(auto_dntime);  
  
err_code = AUTOSTROBE(true, false, false, false); // turn on channel 1 auto strobe (set false  
= turn off)
```

SAVE PARAMTERS IN EEPROM:

```
err_code = EEPROM();
```

EXAMPLE (VB.NET):**DLL IMPORT:**

```

Const LEDStudio As String = "C:\LEDStudio\LEDLib\LEDStudio.dll"
<DllImport(LEDStudio)> _
Public Function LE_ComportConnect(ByRef ComPort As Byte) As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_ComportDisconnect(ByVal Comport As Byte) As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_USBConnect(ByRef ProductName As String) As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_USBDisconnect() As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_SetCHMode(ByVal sCH As Byte, ByVal sMode As Byte) As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_SetMultiplier(ByVal sMul As Byte) As Byte
End Function
<DllImport(LEDStudio)> _
Public Function LE_SetConstInt(ByVal sInt As Byte) As Byte
End Function

```

CONNECT:

```

Dim comport As Byte = 1 'COM1
Dim productname As String = ""

If (COM_Connect) Then 'COMPORT
    If (connected == false)
        err_code = LE_ComportConnect (comport)
    Else
        err_code = LE_ComportDisconnect (comport)
    End If
Else 'USB
    If (connected == false)
        err_code = LE_USBConnect (productname)
    Else
        err_code = LE_USBDisconnect ()
    End If
End If

```

ERROR CHECK:

```
Dim err_msg As String
If Not err_code = 0 Then
    err_msg = LE_GETErrMsg(err_code)
```

SET MULTIPLIER:

```
Dim multiplier As Byte = 5 '50% of max current
err_code = LE_SetMultiplier(multiplier)
```

SET CHANNEL & MODE:

```
Dim channel As Integer = 1 'CH1
Dim mode As Integer = 0 'constant mode
err_code = LE_SetCHMode(channel, mode)
```

CONSTANT:

```
Dim intensity as Byte = 100
err_code = LE_SetConstInt(intensity)
```

```
Dim on_off As Byte = 1 'On
err_code = LE_SetCONSTOnOff (on_off)
```

STROBE/TRIGGER:

```
Dim strobe_width as Short = 1000 '1ms strobe width
err_code = LE_SetStrobeWidth(strobe_width)
err_code = LE_STROBE(true, false, false, false) 'strobe channel 1
```

AUTO STROBE:

```
Dim auto_uptime As Short = 250 '250ms up time
Dim auto_dntime As Short = 400 '400ms down time
err_code = LE_SetAutoStrobeUpTime(auto_uptime)
err_code = LE_SetAutoStrobeDnTime(auto_dntime)
err_code = LE_AUTOSTROBE(true, false, false, false) 'turn on channel 1 auto strobe (set
false = turn off)
```

SAVE PARAMTERS IN EEPROM:

```
err_code = LE_EEPROM()
```

RS232 Raw Command

COM Port Setting

Baud Rate = 19200

Data Bits = 8

Parity = None

Stop Bits = 1

Message Format

Issuing command (PC -> Controller)

Packet length (N+4)	1 Byte
Address	1 Byte
Command	1 Byte
Data (N bytes)	N Byte
Checksum	1 Byte

Status response (Controller -> PC)

Packet length (N+4)	1 Byte
Address	1 Byte
Communication Status	1 Byte
Response data (N bytes)	N Byte
Checksum	1 Byte

*** All bytes are sent in hex value**

Address

Address: 0 to 15 (00h - 0Fh)

For cascade operation:

Master controller address: 00h

Slave address: 01h -0Fh

Checksum

Checksum is XOR of all previous bytes

Example: For a 6-byte Message, the Checksum (byte5) will be the XOR of byte0 to byte4

Command List

Command Group	Command Code	Command	Data Length	Data
System Command	00h			
	01h			
	02h			
	03h			
	0Fh	Save Param in EEPROM	0 Byte	-
Lightings Controls	10h	Set Multiplier (All channel)	1 Byte	XXh Byte 1 = Multiplier (01h - 0Ah)
	11h	Set Channel Mode	2 Byte	XX YYh Byte 1 = Channel No. (00h - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2 = Mode (00h = Constant) (01h = Strobe) (02h = Trigger) (04h = Cycle Strobe) x
	12h	On/Off lighting output	2 Byte	XX YYh Byte 1 = Channel No. (00h - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2 = On/Off (01h=ON, 00h=OFF)
	13h	Set Constant Intensity	2 Byte	XX YYh Byte 1 = Channel No. (00h - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2 = Intensity (0 - 255) = (00h - FFh)
	14h	Set Strobe Intensity	2 Byte	XX YYh Byte 1 = Channel No. (00h - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2 = Intensity (0 - 255) = (00h - FFh)
	15h	Set Strobe Delay	3 Byte	XX YYYYh Byte 1 = Channel No. (00 - 0Fh) [Bit0 = CH1, Bit1 = CH2,

				Bit2 = CH3, Bit3 = CH4] Byte 2&3 = Time (0-9999)(100us) (0000h – 270Fh)
	16h	Set Strobe Width	3 Byte	XX YYYYh Byte 1 = Channel No. (00 - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2&3 = Time (0-9999)(100us) (0000h – 270Fh)
	17h	Set Output Delay	3 Byte	XX YYYYh Byte 1 = Channel No. (00 - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4] Byte 2&3 = Time (0-9999)(100us) (0000h – 270Fh)
	19h	Strobe	1 Byte	XXh Byte 1 = Channel No. (00 - 0Fh) [Bit0 = CH1, Bit1 = CH2, Bit2 = CH3, Bit3 = CH4]
	1Fh			
Unused	F0 - FFh	Used for return message	N/A	N/A

Communication Status

Bit	Description	0	1
0	Operation command execution condition	Not executed	Executed
1	Command error	Normal	Error
2	Data error	Normal	Error
3	Communication error	Normal	Error
4 – 7	All bits are set as 1		

Example

1) Set multiplier as 10:

Tx > 05 00 10 0A 1F (5bytes)

Rx < 04 00 F1 F5 (Success)

2) Set CH1 channel mode to constant mode (0x00)

Tx > 06 00 11 01 00 16 (6bytes)

Rx < 04 00 F1 F5 (Success)

3) Set CH2 channel mode to strobe mode (0x01)

Tx > 06 00 11 02 01 14 (6bytes)

Rx < 04 00 F1 F5 (Success)

4) Set CH1 intensity = 100 (64h)

Tx > 06 00 13 01 64 70 (6bytes)

Rx < 04 00 F1 F5 (Success)

5) Set CH1 channel mode to strobe mode

Tx > 06 00 11 01 01 17 (6bytes)

Rx < 04 00 F1 F5 (Success)

6) Set CH1 Strobe Width to 5.0ms = send as 50 (32h)

Tx > 07 00 16 01 00 32 22 (7bytes)

Rx < 04 00 F1 F5 (Success)

7) Trigger CH1 (strobe)

Tx > 05 00 19 01 1D (5bytes)

Rx < 04 00 F1 F5 (Success)



MACHINE VISION SHOP

FOR ALL YOUR VISION PRODUCTS

Thank you for downloading this document from
www.machine-vision-shop.com

If you have any questions, you need help composing the right package for your application or do you want to order?

Feel free to contact us via e-mail at
sales@vision-consultancy.nl or visit our webshop.

Our vision experts are happy to help you.



Natascha Overhof



Christian Cromptoets



Machine Vision Shop

Trade name of Vision Consultancy BV

Robert Schumandomein 2
6229 ES Maastricht
The Netherlands

+31 (0) 438 522 651

sales@vision-consultancy.nl
www.machine-vision-shop.com

Scan me to visit
[machine-vision-shop](http://machine-vision-shop.com)

