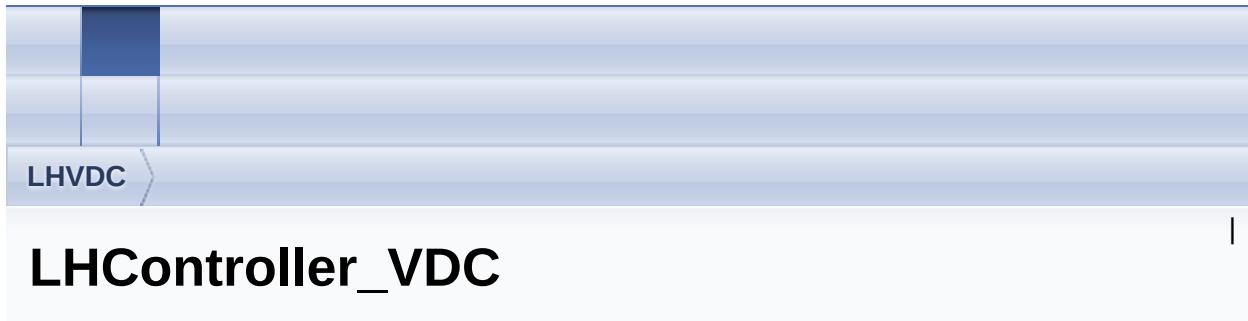


# LHController\_VDC v1.0.1

## API Instructions



```
#define LH_SUCCEED 0
```

```
...
```

```
#define LH_ERR_CREATEETHECON_FAILED 1001
```

```
...
```

```
#define LH_ERR_UNKNOWN 1002
```

```
...
```

```
#define LH_ERR_INVALIDHANDLE 1003
```

```
...
```

```
#define LH_ERR_MAXIMUM 1004
```

```
...
```

```
#define LH_ERR_INITSERIAL_FAILED 1005
```

```
...
```

```
#define LH_ERR_SEND_DATA 1006
```

```
...
```

```
#define LH_ERR_CHINDEX_OUTRANGE 1007
```

```
...
```

```
#define LH_ERR_DISCONNECTED 1008
```

```
...
```

```
#define LH_ERR_CHINDEX_LACK 1009
```

```
...
```

```
#define LH_ERR_BRIGHTNESS_LEVEL 1010
```

```
...
```

```
#define LH_ERR_TRIGGERWIDTH 1012
...
#define LH_ERR_TRIGGERMODE 1013
...
#define LH_ERR_VOLTAGE 1014
...
#define LH_ERR_DEFAULT 1015
...
#define LH_ERR_BAUDRATE 1016
...
#define LH_ERR_PARAMETER 1017
...
#define LH_ERR_TURNONCH_FAILED 1018
...
#define LH_ERR_TURNOFFCH_FAILED 1019
...
#define LH_ERR_SETBRIGHTNESS_FAILED 1020
...
#define LH_ERR_READBRIGHTNESS_FAILED 1021
...
#define LH_ERR_STROBETRIGGER_FAILED 1022
...
#define LH_ERR_SETTRIGGERWIDTH_FAILED 1023
```

```
#define LH_ERR_READTRIGGERWIDTH_FAILED 1024
```

```
...
```

```
#define LH_ERR_SEREXTERNALTRIGGER_FAILED 1025
```

```
...
```

```
#define LH_ERR_READEXTERNALTRIGGER_FAILED 1026
```

```
...
```

```
#define LH_ERR_SEROUTPUTVOLTAGE_FAILED 1027
```

```
...
```

```
#define LH_ERR_READOUTPUTVOLTAGE_FAILED 1028
```

```
...
```

```
#define LH_ERR_SETDEFAULTOUTPUT_FAILD 1029
```

```
...
```

```
#define LH_ERR_READDEFAULTOUTPUT_FAILD 1030
```

```
...
```

```
#define LH_ERR_SEDEXTERNALIOMODE_FAILD 1031
```

```
IO
```

```
...
```

```
#define LH_ERR_READEXTERNALIOMODE_FAILD 1032
```

```
IO
```

```
...
```

```
#define LH_ERR_READINPUT_FAILD 1033
```

```
IO
```

```
...
```

```
#define LH_ERR_SETOUPUT_FAILD 1034
```

```
IO
```

```
...
```

```
#define LH_ERR_READALARM_FAILD 1035
```

```
...
```

```
#define LH_ERR_CLEAREDZALARM_FAILD 1036
```

```
...
```

```
#define LH_ERR_REDTEMPERATURE_FAILD 1037
```

```
...
```

```
#define LH_ERR_REBOOT_FAILD 1038
```

```
...
```

```
#define LH_ERR_FORMAT_FAILD 1039
```

```
...
```

```
#define LH_ERR_SAVE_FAILD 1040
```

```
...
```

```
#define LH_ERR_SETCOMBAUDRATE_FAILD 1041
```

```
...
```

```
#define LH_ERR_SETSERVERIP_FAILD 1042
```

```
IP
```

```
...
```

```
#define LH_ERR_SETSERVERSUBNETMASK_FAILD 1043
```

```
...
```

```
#define LH_ERR_SETSERVERGATEWAY_FAILD 1044
```

```
...
```

```
#define LH_ERR_SETCLIENTIP_FAILD 1045
```

```
IP
```

```
...
```

```
#define LH_ERR_SETSERVERPORT_FAILD 1046
```

```
...
```

```
#define LH_ERR_SETCLIENTPORT_FAILD 1047
```

```
...
```

```
#define LH_ERR_READTCPIP_FAILD 1048  
TCP ...
```

```
#define LH_ERR_READVERSION_FAILD 1049  
...
```

```
#define LHController_Handle long
```

```
#define LHController_VDC_API long
```

---

**LHController\_VDC\_API** [\*\*LHController\\_Synchronous\\_asynchronous\*\*](#) (**LHController\_Handle** controllerHandle, int sa)

...

**LHController\_VDC\_API** [\*\*LHController\\_CreateEtheConnectionByIP\*\*](#) (char \*serverIPAddress, int serverPort, **LHController\_Handle** \*controllerHandle)

...

**LHController\_VDC\_API** [\*\*LHController\\_DestroyEtheConnection\*\*](#) (**LHController\_Handle** controllerHandle)

...

**LHController\_VDC\_API** [\*\*LHController\\_TurnonChannel\*\*](#) (**LHController\_Handle** controllerHandle, int channelIndex)

...

**LHController\_VDC\_API** [\*\*LHController\\_InitSerialPort\*\*](#) (char \*comName, int baudrate, **LHController\_Handle** \*controllerHandle)

...

**LHController\_VDC\_API** [\*\*LHController\\_ReleaseSerialPort\*\*](#) (**LHController\_Handle** controllerHandle)

...

**LHController\_VDC\_API** [\*\*LHController\\_TurnoffChannel\*\*](#) (**LHController\_Handle** controllerHandle, int channelIndex)

...

**LHController\_TurnoffTurnonChannel**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, char \*channelIndex)

...

**LHController\_SetIntensity**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int intensity)

...

**LHController\_ReadIntensity**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int \*intensity)

...

**LHController\_TurnonChannelSetIntensity**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int intensity)

...

**LHController\_AStrobeTrigger**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex)

...

**LHController\_ManyStrobeTrigger**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, char \*channelIndex)

...

**LHController\_SetTriggerWidth**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int triggerWidth)

...

**LHController\_ReadTriggerWidth**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int \*triggerWidth)

...

<b>LHController_VDC_API</b>	<b>LHController_SetExternalTrigger</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int channelMode)
	...
<b>LHController_VDC_API</b>	<b>LHController_ReadExternalTrigger</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int *channelMode)
	...
<b>LHController_VDC_API</b>	<b>LHController_TimeTrobeTrigger</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int triggerWidth)
	...
<b>LHController_VDC_API</b>	<b>LHController_SetoutputVoltage</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int voltage)
	...
<b>LHController_VDC_API</b>	<b>LHController_ReadOutputVoltage</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int *voltage)
	...
<b>LHController_VDC_API</b>	<b>LHController_SetDefaultOutput</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int channelDefault)
	...
<b>LHController_VDC_API</b>	<b>LHController_ReadDefaultOutput</b> ( <b>LHController_Handle</b> controllerHandle, int channelIndex, int *channelDefault)
	...
<b>LHController_VDC_API</b>	<b>LHController_SetExternalIOmode</b> ( <b>LHController_Handle</b> controllerHandle, int

	<inputoutput, externalmode)<br="" int=""></inputoutput,> IO ...
LHController_VDC_API	<b>LHController_ReadExternalIOmode</b> ( <b>LHController_Handle</b> controllerHandle, int inputOutput, int *externalmode) IO ...
LHController_VDC_API	<b>LHController_ReadInput</b> ( <b>LHController_Handle</b> controllerHandle, char *readInput) IO ...
LHController_VDC_API	<b>LHController_SetOutput</b> ( <b>LHController_Handle</b> controllerHandle, char *output) IO ...
LHController_VDC_API	<b>LHController_Readalarm</b> ( <b>LHController_Handle</b> controllerHandle, char *alarm) ...
LHController_VDC_API	<b>LHController_Clearedzalarm</b> ( <b>LHController_Handle</b> controllerHandle) ...
LHController_VDC_API	<b>LHController_Redtemperature</b> ( <b>LHController_Handle</b> controllerHandle, int *temperature) ...
LHController_VDC_API	<b>LHController_Reboot</b> ( <b>LHController_Handle</b> controllerHandle) ...
LHController_VDC_API	<b>LHController_Format</b> ( <b>LHController_Handle</b> controllerHandle)

LHController_VDC_API	LHController_Save ( <b>LHController_Handle</b> controllerHandle)
LHController_VDC_API	LHController_SetComBaudrate ( <b>LHController_Handle</b> controllerHandle, int baudrate)
LHController_VDC_API	LHController_SetServerIP ( <b>LHController_Handle</b> controllerHandle, char *serverip) IP ...
LHController_VDC_API	LHController_SetServerSubnetMask ( <b>LHController_Handle</b> controllerHandle, char *subnetmask)
LHController_VDC_API	LHController_SetServerGateway ( <b>LHController_Handle</b> controllerHandle, char *gateway)
LHController_VDC_API	LHController_SetClientIP ( <b>LHController_Handle</b> controllerHandle, char *clientIP) IP ...
LHController_VDC_API	LHController_SetServerPort ( <b>LHController_Handle</b> controllerHandle, int serverport)
LHController_VDC_API	LHController_SetClientPort ( <b>LHController_Handle</b> controllerHandle, int

clientport)

...

**LHController\_ReadTCPIP**

(**LHController\_Handle** controllerHandle, char \*serverIP, char \*serverSubnetMask, char \*serverGateway, char \*clientIP, int \*serverport, int \*clientport)

TCP

...

**LHController\_Readversion**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, char \*version)

...

**LHController\_Hig\_SetTriggerWidth**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int triggerWidth)

...

**LHController\_Hig\_ReadTriggerWidth**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int \*triggerWidth)

...

**LHController\_Hig\_TimeTrobeTrigger**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int channelIndex, int triggerWidth)

...

**LHController\_SetCompleteSignal**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int triggerWidth)

...

**LHController\_ReadCompleteSignal**

**LHController\_VDC\_API** (**LHController\_Handle** controllerHandle, int \*triggerWidth)

...

**LHController\_VDC\_API** **LHController\_SetDelayTime**  
**(LHController\_Handle** controllerHandle, int  
triggerWidth)

...

**LHController\_VDC\_API** **LHController\_ReadDelayTime**  
**(LHController\_Handle** controllerHandle, int  
\*triggerWidth)

...

**LHController\_VDC\_API** **LHController\_SetIntervalTime**  
**(LHController\_Handle** controllerHandle, int  
triggerWidth)

...

**LHController\_VDC\_API** **LHController\_ReadIntervalTime**  
**(LHController\_Handle** controllerHandle, int  
\*triggerWidth)

...

---

```
#define LH_ERR_BAUDRATE 1016
```

---

---

```
#define LH_ERR_BRIGHTNESS_LEVEL 1010
```

---

---

```
#define LH_ERR_CHINDEX_LACK 1009
```

---

---

```
#define LH_ERR_CHINDEX_OUTRANGE 1007
```

---

---

```
#define LH_ERR_CLEAREDZALARM_FAILD 1036
```

---

---

```
#define LH_ERR_CREATEETHECON_FAILED 1001
```

---

---

```
#define LH_ERR_DEFAULT 1015
```

---

---

```
#define LH_ERR_DISCONNECTED 1008
```

---

---

```
#define LH_ERR_FORMAT_FAILD 1039
```

---

---

```
#define LH_ERR_INITSERIAL_FAILED 1005
```

---

---

```
#define LH_ERR_INVALIDHANDLE 1003
```

---

---

```
#define LH_ERR_MAXIMUM 1004
```

---

---

```
#define LH_ERR_PARAMETER 1017
```

---

---

```
#define LH_ERR_READALARM_FAILD 1035
```

---

---

```
#define LH_ERR_READBRIGHTNESS_FAILED 1021
```

---

---

```
#define LH_ERR_READDEFAULTOUTPUT_FAILD 1030
```

---

---

```
#define LH_ERR_READEXTERNALIOMODE_FAILD 1032
```

---

IO

---

```
#define LH_ERR_READEXTERNALTRIGGER_FAILED 1026
```

---

---

```
#define LH_ERR_READINPUT_FAILD 1033
```

---

IO

---

```
#define LH_ERR_READOUTPUTVOLTAGE_FAILED 1028
```

---

---

```
#define LH_ERR_READTCPIP_FAILD 1048
```

---

TCP

---

```
#define LH_ERR_READTRIGGERWIDTH_FAILED 1024
```

---

---

```
#define LH_ERR_READVERSION_FAILD 1049
```

---

---

```
#define LH_ERR_REBOOT_FAILD 1038
```

---

---

```
#define LH_ERR_REDTEMPERATURE_FAILD 1037
```

---

---

```
#define LH_ERR_SAVE_FAILD 1040
```

---

---

```
#define LH_ERR_SEDEXTERNALIOMODE_FAILD 1031
```

---

IO

---

```
#define LH_ERR_SEND_DATA 1006
```

---

---

```
#define LH_ERR_SEREXTERNALTRIGGER_FAILED 1025
```

---

```
#define LH_ERR_SEROUTPUTVOLTAGE_FAILED 1027
```

---

```
#define LH_ERR_SETBRIGHTNESS_FAILED 1020
```

---

```
#define LH_ERR_SETCIENTIP_FAILD 1045
```

---

IP

```
#define LH_ERR_SETCIENTPORT_FAILD 1047
```

---

```
#define LH_ERR_SETCOMBAUDRATE_FAILD 1041
```

---

```
#define LH_ERR_SETDEFAULTOUTPUT_FAILD 1029
```

---

```
#define LH_ERR_SETOUTPUT_FAILD 1034
```

---

IO

```
#define LH_ERR_SETSERVERGATEWAY_FAILED 1044
```

---

```
#define LH_ERR_SETSERVERIP_FAILED 1042
```

---

IP

```
#define LH_ERR_SETSERVERPORT_FAILED 1046
```

---

```
#define LH_ERR_SETSERVERSUBNETMASK_FAILED 1043
```

---

```
#define LH_ERR_SETTRIGGERWIDTH_FAILED 1023
```

---

```
#define LH_ERR_STROBETRIGGER_FAILED 1022
```

---

```
#define LH_ERR_TRIGGRRMODE 1013
```

---

```
#define LH_ERR_TRIGGRRWIDTH 1012
```

---

---

```
#define LH_ERR_TURNOFFCH_FAILED 1019
```

---

---

```
#define LH_ERR_TURNONCH_FAILED 1018
```

---

---

```
#define LH_ERR_UNKNOWN 1002
```

---

---

```
#define LH_ERR_VOLTAGE 1014
```

---

---

```
#define LH_SUCCEED 0
```

---

---

```
#define LHController_Handle long
```

---

---

```
#define LHController_VDC_API long
```

---

---

### **LHController\_VDC\_API**

```
LHController_AStrobeTrigger ( LHController_Handle controllerHandle
                                int channelIndex
                            )
```

---

**controllerHandle**

**channelIndex** 1~8

LH\_SUCCEED

---

### **LHController\_VDC\_API**

```
LHController_Clearedzalarm ( LHController_Handle controllerHandle
```

---

**controllerHandle**

LH\_SUCCEED

## LHController\_VDC\_API

```
LHController_CreateEtheConnectionByIP ( char *
                                         int
                                         LHController_Handle *
                                         )
```

---

**serverIPAddress** IP  
172.16.83.1  
**serverPort** ,1000~9999

**controllerHandle**

LH\_SUCCEED

100

LHController\_DestroyEtheConnection

## LHController\_VDC\_API

**LHController\_DestroyEtheConnection ( LHController\_Handle controllerHandle )**

---

**controllerHandle**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_Format ( LHController\_Handle controllerHandle )**

---

**controllerHandle**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_Hig\_ReadTriggerWidth ( LHController\_Handle controllerHandle )**

```
    int  
    int *  
)  
chanr  
trigge
```

---

**controllerHandle**

**version**

LH\_SUCCEED

### **LHController\_VDC\_API**

```
LHController_Hig_SetTriggerWidth ( LHController_Handle control  
                                    int channel  
                                    int triggerV  
)  
triggerV
```

---

**controllerHandle**

**version**

LH\_SUCCEED

## LHController\_VDC\_API

```
LHController_Hig_TimeTrobeTrigger( LHController_Handle contrc  
                                int                  chann  
                                int                  trigger  
                                )
```

## controllerHandle

## version

LH SUCCEED

## LHController\_VDC\_API

```
LHController_InitSerialPort ( char * comName,
                                int baudrate,
                                LHController_Handle * controllerHandle
                            )
```

**comName** COM1  
**baudrate**

## controllerHandle

## LH\_SUCCEED

100

## LHController\_ReleaseSerialPort

## LHController\_VDC\_API

```
LHController_ManyStrobeTrigger ( LHController_Handle controller,
                                  char * channelNr
                                )
```

## controllerHandle

**channelIndex** ABCDEFGH A 1 B 2 C 3  
D 4 E 5 F 6 G 7 H 8  
0 1 "11111111","10000000"

LH SUCCEED

**LHController\_VDC\_API**

**LHController\_Readalarm ( LHController\_Handle controllerHandle,  
                                  char \*                          alarm  
                                  )**

---

**controllerHandle**

**alarm**              ABCD—— LED  0 LED  
                          ? 1 LED ? 2 ? A  
                          1 B  2 C  3 D  4

**LH\_SUCCEED**

**LHController\_VDC\_API**

**LHController\_ReadCompleteSignal ( LHController\_Handle control  
                                  int \*                          trigger  
                                  )**

---

**controllerHandle**

**version**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadDefaultOutput ( LHController\_Handle controllerHandle, int channelIndex, int \*channelDefault )**

---

**controllerHandle**

**channelIndex** 1~8

**channelDefault** “0” “1”

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadDelayTime ( LHController\_Handle controllerHandle, int channelIndex, int \*delayTime )**

```
    int * triggerWidth  
)
```

---

**controllerHandle**

**version**

LH\_SUCCEED

## LHController\_VDC\_API

```
LHController_ReadExternalIOmode ( LHController_Handle control  
                                int inputOutput  
                                int * externalmode  
)
```

---

IO

**controllerHandle**

**inputOutput** “1” I/O “2” I/O

**externalmode** X 1 Y “0” Y “1” I/O  
IO IO X 2 Y “0” IO  
Y “1” I/O IO IO

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadExternalTrigger** (**LHController\_Handle** controllerHandle, int channelIndex, int \*channelMode)

---

**controllerHandle**

**channelIndex** 1~4

**channelMode** ? 0: ? 1 ? 2 3

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadInput** (**LHController\_Handle** controllerHandle, char \*readInput)

---

IO

**controllerHandle**

**readInput**            IO IO "0" IO "1" IO  
                      1 IO 1 2 IO 3  
                      "00000"

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadIntensity** ( **LHController\_Handle** **controllerHandle**,  
                                  int                            **channelIndex**,  
                                  int \*                          **intensity**  
                                  )

---

**controllerHandle**

**channelIndex**     1~8

**intensity**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_ReadIntervalTime** (**LHController\_Handle controllerHandle**  
**int \*triggerWidth**  
)

---

**controllerHandle**

**version**

**LH\_SUCCEED**

**LHController\_VDC\_API**

**LHController\_ReadOutputVoltage** (**LHController\_Handle controllerHandle**  
**int channelIndex**  
**int \*voltage**  
)

---

**controllerHandle**

**channelIndex** 1~4

<b>voltage</b>	0 4.0V	1 5.0V	2 12.0V	3 20.0V
	4 22.0V	5 24.0V	6 26.0V	7 28.0V

LH\_SUCCEED

### **LHController\_VDC\_API**

```
LHController_ReadTCPIP ( LHController_Handle controllerHandle,  
                           char *               serverIP,  
                           char *               serverSubnetMas  
                           char *               serverGateway,  
                           char *               clientIP,  
                           int *                serverport,  
                           int *                clientport  
)
```

---

TCP

**controllerHandle**

<b>serverIP</b>	IP
<b>serverSubnetMask</b>	
<b>serverGateway</b>	
<b>clientIP</b>	IP
<b>serverport</b>	
<b>clientport</b>	

LH\_SUCCEED

### **LHController\_VDC\_API**

```
LHController_ReadTriggerWidth ( LHController_Handle controllerHandle,
                                int channelIndex,
                                int * triggerWidth
                                )
```

---

**controllerHandle**

**channelIndex** 1~8

**triggerWidth**

**LH\_SUCCEED**

### **LHController\_VDC\_API**

```
LHController_Readversion ( LHController_Handle controllerHandle,
                            char * version
                            )
```

---

**controllerHandle**

**version**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_Reboot ( LHController\_Handle controllerHandle )**

---

**controllerHandle**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_Redtemperature ( LHController\_Handle controllerHandle  
int \* temperature  
)**

---

**controllerHandle**

**temperature**

LH\_SUCCEED

[\*\*LHController\\_VDC\\_API\*\*](#)

**LHController\_ReleaseSerialPort ( LHController\_Handle controllerHandle )**

---

**controllerHandle**

LH\_SUCCEED

[\*\*LHController\\_VDC\\_API\*\*](#)

**LHController\_Save ( LHController\_Handle controllerHandle )**

---

**controllerHandle**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_SetExternalIOmode ( LHController\_Handle controllerHandle, int inputOutput, int externalmode )**

---

IO

**controllerHandle**

**inputOutput** "1" I/O "2" I/O  
**externalmode** X 1 Y "0" Y "1" I/O  
                  IO IO X 2 Y "0"  
                  IO Y "1" I/O IO IO

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_SetClientIP ( LHController\_Handle controllerHandle,**

```
        char *           clientIP  
    )
```

---

IP

controllerHandle  
clientIP IP

LH\_SUCCEED

[LHController\\_VDC\\_API](#)  
**LHController\_SetClientPort**( [LHController\\_Handle](#) controllerHandle  
 int clientport  
 )

---

controllerHandle  
clientport

LH\_SUCCEED

## **LHController\_VDC\_API**

```
LHController_SetComBaudrate ( LHController_Handle controllerH
                                int
                                )
```

---

**controllerHandle**

**baudrate**      7 “1”9600bps “2”  
                  14400bps “3”19200bps “4”38400bps  
                  “5”56000bps “6”57600bps “7”  
                  115200bps

**LH\_SUCCEED**

115200 bps

## **LHController\_VDC\_API**

```
LHController_SetCompleteSignal ( LHController_Handle controllerH
                                    int
                                    )
```

---

**controllerHandle**

**version**

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_SetDefaultOutput ( LHController\_Handle controllerH  
                                  int                            channelInd  
                                  int                            channelDef  
                                  )**

---

**controllerHandle**

**channelIndex**     1~8

**channelDefault** “0” “1”

LH\_SUCCEED

**LHController\_VDC\_API**

**LHController\_SetDelayTime ( LHController\_Handle controllerHand  
                                  int                            triggerWidth  
                                  )**

---

**controllerHandle**

**version**

LH\_SUCCEED

### [\*\*LHController\\_VDC\\_API\*\*](#)

**LHController\_SetExternalTrigger (**[\*\*LHController\\_Handle\*\*](#) controller  
                          int                            channelIn  
                          int                            channelM  
                          )

---

**controllerHandle**

**channelIndex**     1~4

**channelMode**    ? 0: ? 1 ? 2 3

LH\_SUCCEED

## **LHController\_VDC\_API**

```
LHController_SetIntensity ( LHController_Handle controllerHandle
                            int                               channelIndex,
                            int                               intensity
)
```

---

**controllerHandle**

**channelIndex**      1~8

**intensity**          000~255

LH\_SUCCEED

## **LHController\_VDC\_API**

```
LHController_SetIntervalTime ( LHController_Handle controllerHandle
                                int                               triggerWidth
)
```

---

**controllerHandle**

## version

## LH\_SUCCEED

```
LHController_SetOutput( LHController_Handle controllerHandle,  
                        char * output  
                      )
```

10

```
controllerHandle
output      "0" "1" 1 1 IO
              2 2 IO "00000"
```

LH SUCCEED

```
LHController_SetoutputVoltage ( LHController_Handle controllerH  
                                int channelInd  
                                int voltage  
                                )
```

**controllerHandle**

**channelIndex** 1~4

**voltage** V ? 0 4.0V ? 1 5.0V; ? 2  
12.0V; ? 3 20.0V; ? 4 22.0V; ?  
5 24.0V; ? 6 26.0V; ? 7 28.0V;

LH\_SUCCEED

## LHController\_VDC\_API

**LHController\_SetServerGateway** ( **LHController\_Handle controller**  
**char \* gateway**  
)

---

**controllerHandle**

**gateway**

LH\_SUCCEED

## LHController\_VDC\_API

```
LHController_SetServerIP ( LHController_Handle controllerHandle  
                           char *                  serverip  
                           )
```

---

IP

controllerHandle  
serverip IP

LH\_SUCCEED

**LHController\_VDC\_API**

```
LHController_SetServerPort ( LHController_Handle controllerHandle  
                            int                   serverport  
                            )
```

---

controllerHandle  
serverport

LH\_SUCCEED

### **LHController\_VDC\_API**

```
LHController_SetServerSubnetMask ( LHController_Handle contrc  
                                char * subne  
                                )
```

---

**controllerHandle**  
**serverip**

LH\_SUCCEED

### **LHController\_VDC\_API**

```
LHController_SetTriggerWidth ( LHController_Handle controllerHa  
                                int channelInde  
                                int triggerWidth  
                                )
```

---

**controllerHandle**  
**channelIndex** 1~8  
**triggerWidth** 0~9999 1ms



**controllerHandle**  
**channelIndex** 1~8  
**triggerWidth** ms  
0~9999

LH\_SUCCEED

#### [\*\*LHController\\_VDC\\_API\*\*](#)

**LHController\_TurnoffChannel** (**LHController\_Handle** controllerHandle,  
                                  **int**                            channelIndex)  
                                  )

---

**controllerHandle**  
**channelIndex** 1~8

LH\_SUCCEED

#### [\*\*LHController\\_VDC\\_API\*\*](#)

**LHController\_TurnoffTurnonChannel** (**LHController\_Handle** controllerHandle,  
                                  **char \***                            channelName)  
                                  )

---

**controllerHandle**

**channelIndex** ABCDEFGH A 1B 2 C 3 D  
4 E 5 F 6 G 7 H 8  
0 1 "11111111","00000000","10000000"

LH\_SUCCEED

### **LHController\_VDC\_API**

**LHController\_TurnonChannel ( LHController\_Handle controllerHandle , int channelIndex )**

---

**controllerHandle**

**channelIndex** 1~8

LH\_SUCCEED



## LHController\_VDC\_API

**LHController\_TurnonChannelSetIntensity ( LHController\_Handle c  
                                  int                          c  
                                  int                          i  
                                  )**

---

**controllerHandle**

**channelIndex**      1~8

**intensity**            000~255

**LH\_SUCCEED**

# LHController\_VDC v1.0.1

## API Instructions



:

- | -

- LH\_ERR\_BAUDRATE : **LHController\_VDC**
- LH\_ERR\_BRIGHTNESS\_LEVEL : **LHController\_VDC**
- LH\_ERR\_CHINDEX\_LACK : **LHController\_VDC**
- LH\_ERR\_CHINDEX\_OUTRANGE : **LHController\_VDC**
- LH\_ERR\_CLEAREDZALARM\_FAILED : **LHController\_VDC**
- LH\_ERR\_CREATEETHECON\_FAILED : **LHController\_VDC**
- LH\_ERR\_DEFAULT : **LHController\_VDC**
- LH\_ERR\_DISCONNECTED : **LHController\_VDC**
- LH\_ERR\_FORMAT\_FAILED : **LHController\_VDC**
- LH\_ERR\_INITSERIAL\_FAILED : **LHController\_VDC**
- LH\_ERR\_INVALIDHANDLE : **LHController\_VDC**
- LH\_ERR\_MAXIMUM : **LHController\_VDC**
- LH\_ERR\_PARAMETER : **LHController\_VDC**
- LH\_ERR\_READALARM\_FAILED : **LHController\_VDC**
- LH\_ERR\_READBRIGHTNESS\_FAILED : **LHController\_VDC**
- LH\_ERR\_READDEFAULTOUTPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_READEXTERNALIOMODE\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READEXTERNALTRIGGER\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READINPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_READOUTPUTVOLTAGE\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READTCPIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_READTRIGGERWIDTH\_FAILED : **LHController\_VDC**

- LH\_ERR\_READVERSION\_FAILED : **LHController\_VDC**
- LH\_ERR\_REBOOT\_FAILED : **LHController\_VDC**
- LH\_ERR\_REDTEMPERATURE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SAVE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SEDEXTERNALIOMODE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SEND\_DATA : **LHController\_VDC**
- LH\_ERR\_SEREXTERNALTRIGGER\_FAILED : **LHController\_VDC**
- LH\_ERR\_SEROUTPUTVOLTAGE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETBRIGHTNESS\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCLIENTIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCLIENTPORT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCOMBAUDRATE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETDEFAULTOUTPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETOUPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERGATEWAY\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERPORT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERSUBNETMASK\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETTRIGGERWIDTH\_FAILED : **LHController\_VDC**
- LH\_ERR\_STROBETRIGGER\_FAILED : **LHController\_VDC**
- LH\_ERR\_TRIGGERRMODE : **LHController\_VDC**
- LH\_ERR\_TRIGGERRWIDTH : **LHController\_VDC**
- LH\_ERR\_TURNOFFCH\_FAILED : **LHController\_VDC**
- LH\_ERR\_TURNONCH\_FAILED : **LHController\_VDC**
- LH\_ERR\_UNKNOWN : **LHController\_VDC**
- LH\_ERR\_VOLTAGE : **LHController\_VDC**
- LH\_SUCCEED : **LHController\_VDC**
- LHController\_AStrobeTrigger() : **LHController\_VDC**
- LHController\_Clearedzalarm() : **LHController\_VDC**
- LHController\_CreateEtheConnectionByIP() : **LHController\_VDC**
- LHController\_DestroyEtheConnection() : **LHController\_VDC**
- LHController\_Format() : **LHController\_VDC**
- LHController\_Handle : **LHController\_VDC**
- LHController\_Hig\_ReadTriggerWidth() : **LHController\_VDC**
- LHController\_Hig\_SetTriggerWidth() : **LHController\_VDC**
- LHController\_Hig\_TimeTrobeTrigger() : **LHController\_VDC**
- LHController\_InitSerialPort() : **LHController\_VDC**
- LHController\_ManyStrobeTrigger() : **LHController\_VDC**

- LHController\_Readalarm() : **LHController\_VDC**
- LHController\_ReadCompleteSignal() : **LHController\_VDC**
- LHController\_ReadDefaultOutput() : **LHController\_VDC**
- LHController\_ReadDelayTime() : **LHController\_VDC**
- LHController\_ReadExternalIOmode() : **LHController\_VDC**
- LHController\_ReadExternalTrigger() : **LHController\_VDC**
- LHController\_ReadInput() : **LHController\_VDC**
- LHController\_ReadIntensity() : **LHController\_VDC**
- LHController\_ReadIntervalTime() : **LHController\_VDC**
- LHController\_ReadOutputVoltage() : **LHController\_VDC**
- LHController\_ReadTCPIP() : **LHController\_VDC**
- LHController\_ReadTriggerWidth() : **LHController\_VDC**
- LHController\_Readversion() : **LHController\_VDC**
- LHController\_Reboot() : **LHController\_VDC**
- LHController\_Redtemperature() : **LHController\_VDC**
- LHController\_ReleaseSerialPort() : **LHController\_VDC**
- LHController\_Save() : **LHController\_VDC**
- LHController\_SetExternalIOmode() : **LHController\_VDC**
- LHController\_SetClientIP() : **LHController\_VDC**
- LHController\_SetClientPort() : **LHController\_VDC**
- LHController\_SetComBaudrate() : **LHController\_VDC**
- LHController\_SetCompleteSignal() : **LHController\_VDC**
- LHController\_SetDefaultOutput() : **LHController\_VDC**
- LHController\_SetDelayTime() : **LHController\_VDC**
- LHController\_SetExternalTrigger() : **LHController\_VDC**
- LHController\_SetIntensity() : **LHController\_VDC**
- LHController\_SetIntervalTime() : **LHController\_VDC**
- LHController\_SetOutput() : **LHController\_VDC**
- LHController\_SetoutputVoltage() : **LHController\_VDC**
- LHController\_SetServerGateway() : **LHController\_VDC**
- LHController\_SetServerIP() : **LHController\_VDC**
- LHController\_SetServerPort() : **LHController\_VDC**
- LHController\_SetServerSubnetMask() : **LHController\_VDC**
- LHController\_SetTriggerWidth() : **LHController\_VDC**
- LHController\_Synchronous\_asynchronous() : **LHController\_VDC**
- LHController\_TimeTrobeTrigger() : **LHController\_VDC**
- LHController\_TurnoffChannel() : **LHController\_VDC**
- LHController\_TurnoffTurnonChannel() : **LHController\_VDC**
- LHController\_TurnonChannel() : **LHController\_VDC**
- LHController\_TurnonChannelSetIntensity() : **LHController\_VDC**

- LHController\_VDC\_API : **LHController\_VDC**
- 

AuthorRex Van 2017/8/30

# LHController\_VDC v1.0.1

## API Instructions



- I -

- LHController\_AStrobeTrigger() : **LHController\_VDC**
- LHController\_Clearedzalarm() : **LHController\_VDC**
- LHController\_CreateEtheConnectionByIP() : **LHController\_VDC**
- LHController\_DestroyEtheConnection() : **LHController\_VDC**
- LHController\_Format() : **LHController\_VDC**
- LHController\_Hig\_ReadTriggerWidth() : **LHController\_VDC**
- LHController\_Hig\_SetTriggerWidth() : **LHController\_VDC**
- LHController\_Hig\_TimeTrobeTrigger() : **LHController\_VDC**
- LHController\_InitSerialPort() : **LHController\_VDC**
- LHController\_ManyStrobeTrigger() : **LHController\_VDC**
- LHController\_Readalarm() : **LHController\_VDC**
- LHController\_ReadCompleteSignal() : **LHController\_VDC**
- LHController\_ReadDefaultOutput() : **LHController\_VDC**
- LHController\_ReadDelayTime() : **LHController\_VDC**
- LHController\_ReadExternalIOmode() : **LHController\_VDC**
- LHController\_ReadExternalTrigger() : **LHController\_VDC**
- LHController\_ReadInput() : **LHController\_VDC**
- LHController\_ReadIntensity() : **LHController\_VDC**
- LHController\_ReadIntervalTime() : **LHController\_VDC**
- LHController\_ReadOutputVoltage() : **LHController\_VDC**
- LHController\_ReadTCPIP() : **LHController\_VDC**
- LHController\_ReadTriggerWidth() : **LHController\_VDC**
- LHController\_Readversion() : **LHController\_VDC**
- LHController\_Reboot() : **LHController\_VDC**
- LHController\_Redtemperature() : **LHController\_VDC**

- LHController\_ReleaseSerialPort() : **LHController\_VDC**
- LHController\_Save() : **LHController\_VDC**
- LHController\_SetExternalOmode() : **LHController\_VDC**
- LHController\_SetClientIP() : **LHController\_VDC**
- LHController\_SetClientPort() : **LHController\_VDC**
- LHController\_SetComBaudrate() : **LHController\_VDC**
- LHController\_SetCompleteSignal() : **LHController\_VDC**
- LHController\_SetDefaultOutput() : **LHController\_VDC**
- LHController\_SetDelayTime() : **LHController\_VDC**
- LHController\_SetExternalTrigger() : **LHController\_VDC**
- LHController\_SetIntensity() : **LHController\_VDC**
- LHController\_SetIntervalTime() : **LHController\_VDC**
- LHController\_SetOutput() : **LHController\_VDC**
- LHController\_SetOutputVoltage() : **LHController\_VDC**
- LHController\_SetServerGateway() : **LHController\_VDC**
- LHController\_SetServerIP() : **LHController\_VDC**
- LHController\_SetServerPort() : **LHController\_VDC**
- LHController\_SetServerSubnetMask() : **LHController\_VDC**
- LHController\_SetTriggerWidth() : **LHController\_VDC**
- LHController\_Synchronous\_asynchronous() : **LHController\_VDC**
- LHController\_TimeTrobeTrigger() : **LHController\_VDC**
- LHController\_TurnoffChannel() : **LHController\_VDC**
- LHController\_TurnoffTurnonChannel() : **LHController\_VDC**
- LHController\_TurnonChannel() : **LHController\_VDC**
- LHController\_TurnonChannelSetIntensity() : **LHController\_VDC**

# LHController\_VDC v1.0.1

## API Instructions



- I -

- LH\_ERR\_BAUDRATE : **LHController\_VDC**
- LH\_ERR\_BRIGHTNESS\_LEVEL : **LHController\_VDC**
- LH\_ERR\_CHINDEX\_LACK : **LHController\_VDC**
- LH\_ERR\_CHINDEX\_OUTRANGE : **LHController\_VDC**
- LH\_ERR\_CLEAREDZALARM\_FAILED : **LHController\_VDC**
- LH\_ERR\_CREATEETHECON\_FAILED : **LHController\_VDC**
- LH\_ERR\_DEFAULT : **LHController\_VDC**
- LH\_ERR\_DISCONNECTED : **LHController\_VDC**
- LH\_ERR\_FORMAT\_FAILED : **LHController\_VDC**
- LH\_ERR\_INITSERIAL\_FAILED : **LHController\_VDC**
- LH\_ERR\_INVALIDHANDLE : **LHController\_VDC**
- LH\_ERR\_MAXIMUM : **LHController\_VDC**
- LH\_ERR\_PARAMETER : **LHController\_VDC**
- LH\_ERR\_READALARM\_FAILED : **LHController\_VDC**
- LH\_ERR\_READBRIGHTNESS\_FAILED : **LHController\_VDC**
- LH\_ERR\_READDEFAULTOUTPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_READEXTERNALIOMODE\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READEXTERNALTRIGGER\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READINPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_READOUTPUTVOLTAGE\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_READTCPIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_READTRIGGERWIDTH\_FAILED : **LHController\_VDC**

- LH\_ERR\_READVERSION\_FAILED : **LHController\_VDC**
- LH\_ERR\_REBOOT\_FAILED : **LHController\_VDC**
- LH\_ERR\_REDTEMPERATURE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SAVE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SEDEXTERNALIOMODE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SEND\_DATA : **LHController\_VDC**
- LH\_ERR\_SEREXTERNALTRIGGER\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_SEROUTPUTVOLTAGE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETBRIGHTNESS\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCLIENTIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCLIENTPORT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETCOMBAUDRATE\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETDEFAULTOUTPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETOUPUT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERGATEWAY\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERIP\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERPORT\_FAILED : **LHController\_VDC**
- LH\_ERR\_SETSERVERSUBNETMASK\_FAILED :  
**LHController\_VDC**
- LH\_ERR\_SETTRIGGERWIDTH\_FAILED : **LHController\_VDC**
- LH\_ERR\_STROBETRIGGER\_FAILED : **LHController\_VDC**
- LH\_ERR\_TRIGGERRMODE : **LHController\_VDC**
- LH\_ERR\_TRIGGERRWIDTH : **LHController\_VDC**
- LH\_ERR\_TURNOFFCH\_FAILED : **LHController\_VDC**
- LH\_ERR\_TURNONCH\_FAILED : **LHController\_VDC**
- LH\_ERR\_UNKNOWN : **LHController\_VDC**
- LH\_ERR\_VOLTAGE : **LHController\_VDC**
- LH\_SUCCEED : **LHController\_VDC**
- LHController\_Handle : **LHController\_VDC**
- LHController\_VDC\_API : **LHController\_VDC**

# LHController\_VDC v1.0.1.1

## API Instructions



⋮

[ 1 2 ]

▼ LHVDC

LHController\_VDC

---

AuthorRex Van 2017/8/30

# LHController\_VDC v1.0.1

## API Instructions



LHVDC



---

## LHController\_VDC

---

AuthorRex Van 2017/8/30

# LHController\_VDC v1.0.1

## API Instructions

LHVDC

### LHController\_VDC

```
1 #define LH_SUCCEED  
0  
3 #define LH_ERR_CREATEETHECON_FAILED  
1001  
5 #define LH_ERR_UNKNOWN  
1002  
7 #define LH_ERR_INVALIDHANDLE  
1003  
9 #define LH_ERR_MAXIMUM  
1004  
11 #define LH_ERR_INITSERIAL_FAILED  
1005  
13 #define LH_ERR_SEND_DATA  
1006  
15 #define LH_ERR_CHINDEX_OUTRANGE  
1007  
17 #define LH_ERR_DISCONNECTED  
1008  
19 #define LH_ERR_CHINDEX_LACK  
1009  
21 #define LH_ERR_BRIGHTNESS_LEVEL1  
1010  
23 #define LH_ERR_TRIGGERWIDTH  
1012  
25 #define LH_ERR_TRIGGERMODE
```

```
1013
27 #define LH_ERR_VOLTAGE
1014
29 #define LH_ERR_DEFAULT
1015
31 #define LH_ERR_BAUDRATE
1016
33 #define LH_ERR_PARAMETER
1017
35 #define LH_ERR_TURNONCH_FAILED
1018
37 #define LH_ERR_TURNOFFCH_FAILED
1019
39 #define LH_ERR_SETBRIGHTNESS_FAILED
1020
41 #define LH_ERR_READBRIGHTNESS_FAILED
1021
43 #define LH_ERR_STROBETRIGGER_FAILED
1022
45 #define LH_ERR_SETTRIGGERWIDTH_FAILED
1023
47 #define LH_ERR_READTRIGGERWIDTH_FAILED
1024
49 #define LH_ERR_SEREXTERNALTRIGGER_FAILED
1025
51 #define LH_ERR_READEXTERNALTRIGGER_FAILED
1026
53 #define LH_ERR_SEROUTPUTVOLTAGE_FAILED
1027
55 #define LH_ERR_READOUTPUTVOLTAGE_FAILED
1028
57 #define LH_ERR_SETDEFAULTOUTPUT_FAILD
1029
59 #define LH_ERR_READDEFAULTOUTPUT_FAILD
1030
61 #define LH_ERR_SEDEXTERNALIOMODE_FAILD
1031
```

```
63 #define LH_ERR_READEXTERNALIOMODE_FAILD  
1032  
65 #define LH_ERR_READINPUT_FAILD  
1033  
67 #define LH_ERR_SETOUTPUT_FAILD  
1034  
69 #define LH_ERR_READALARM_FAILD  
1035  
71 #define LH_ERR_CLEAREDZALARM_FAILD  
1036  
73 #define LH_ERR_READTEMPERATURE_FAILD  
1037  
75 #define LH_ERR_REBOOT_FAILD  
1038  
77 #define LH_ERR_FORMAT_FAILD  
1039  
79 #define LH_ERR_SAVE_FAILD  
1040  
81 #define LH_ERR_SETCOMBAUDRATE_FAILD  
1041  
83 #define LH_ERR_SETSERVERIP_FAILD  
1042  
85 #define LH_ERR_SETSERVERSUBNETMASK_FAILD  
1043  
87 #define LH_ERR_SETSERVERGATEWAY_FAILD  
1044  
89 #define LH_ERR_SETCLIENTIP_FAILD  
1045  
91 #define LH_ERR_SETSERVERPORT_FAILD  
1046  
93 #define LH_ERR_SETCLIENTPORT_FAILD  
1047  
95 #define LH_ERR_READTCPIP_FAILD  
1048  
97 #define LH_ERR_READVERSION_FAILD  
1049  
99
```

```
100 #define LHController_Handle long
101 #define LHController_VDC_API long
102
115 LHController_VDC_API
    LHController_Synchronous_asynchronous(LHController_Handle controllerHandle, int sa);
116
117
133 LHController_VDC_API
    LHController_CreateEtheConnectionByIP(char *serverIPAddress, int serverPort,
                                             LHController_Handle *controllerHandle);
134
145 LHController_VDC_API
    LHController_DestroyEtheConnection(LHController_Handle controllerHandle);
157 LHController_VDC_API
    LHController_TurnonChannel(LHController_Handle controllerHandle, int channelIndex);
173 LHController_VDC_API
    LHController_InitSerialPort(char *comName, int baudrate, LHController_Handle *controllerHandle);
184 LHController_VDC_API
    LHController_ReleaseSerialPort(LHController_Handle controllerHandle);
195 LHController_VDC_API
    LHController_TurnoffChannel(LHController_Handle controllerHandle, int channelIndex);
207 LHController_VDC_API
    LHController_TurnoffTurnonChannel(LHController_Handle controllerHandle, char *channelIndex);
219 LHController_VDC_API
    LHController_SetIntensity(LHController_Handle controllerHandle, int channelIndex, int intensity);
232 LHController_VDC_API
```

```
    LHController_ReadIntensity(LHController_Handle
controllerHandle, int channelIndex, int
*intensity);
244 | LHController_VDC_API
    LHController_TurnonChannelSetIntensity(LHController_Handle
controllerHandle, int
channelIndex, int intensity);
255 | LHController_VDC_API
    LHController_AStrobeTrigger(LHController_Handle
controllerHandle, int channelIndex);
267 | LHController_VDC_API
    LHController_ManyStrobeTrigger(LHController_Handle
controllerHandle, char *channelIndex);
279 | LHController_VDC_API
    LHController_SetTriggerWidth(LHController_Handle
controllerHandle, int channelIndex, int
triggerWidth);
292 | LHController_VDC_API
    LHController_ReadTriggerWidth(LHController_Handle
controllerHandle, int channelIndex, int*
triggerWidth);
293 |
310 | LHController_VDC_API
    LHController_SetExternalTrigger(LHController_Handle
controllerHandle, int channelIndex, int
channelMode);
311 |
330 | LHController_VDC_API
    LHController_ReadExternalTrigger(LHController_Handle
controllerHandle, int channelIndex, int
*channelMode);
331 |
344 | LHController_VDC_API
    LHController_TimeTrobeTrigger(LHController_Handle
controllerHandle, int channelIndex, int
triggerWidth);
345 |
```

```
365 LHController_VDC_API
    LHController_SetoutputVoltage(LHController_Handle
        controllerHandle, int channelIndex, int
        voltage);
386 LHController_VDC_API
    LHController_ReadOutputVoltage(LHController_Ha
        ndle controllerHandle, int channelIndex, int
        *voltage);
387
400 LHController_VDC_API
    LHController_SetDefaultOutput(LHController_Han
        dle controllerHandle, int channelIndex, int
        channelDefault);
414 LHController_VDC_API
    LHController_ReadDefaultOutput(LHController_Ha
        ndle controllerHandle, int channelIndex, int
        *channelDefault);
429 LHController_VDC_API
    LHController_SetExternalIOMode(LHController_Ha
        ndle controllerHandle, int inputOutput, int
        externalmode);
430
445 LHController_VDC_API
    LHController_ReadExternalIOMode(LHController_H
        andle controllerHandle, int inputOutput, int
        *externalmode);
446
459 LHController_VDC_API
    LHController_ReadInput(LHController_Handle
        controllerHandle, char *readInput);
460
472 LHController_VDC_API
    LHController_SetOutput(LHController_Handle
        controllerHandle, char *output);
473
489 LHController_VDC_API
    LHController_Readalarm(LHController_Handle
```

```
    controllerHandle, char *alarm);  
490  
500 LHController_VDC_API  
    LHController_Clearedzalarm(LHController_Handle  
    controllerHandle);  
512 LHController_VDC_API  
    LHController_Redtemperature(LHController_Handle  
    controllerHandle, int *temperature);  
513  
523 LHController_VDC_API  
    LHController_Reboot(LHController_Handle  
    controllerHandle);  
524  
534 LHController_VDC_API  
    LHController_Format(LHController_Handle  
    controllerHandle);  
535  
545 LHController_VDC_API  
    LHController_Save(LHController_Handle  
    controllerHandle);  
546  
565 LHController_VDC_API  
    LHController_SetComBaudrate(LHController_Handle  
    controllerHandle, int baudrate);  
566  
577 LHController_VDC_API  
    LHController_SetServerIP(LHController_Handle  
    controllerHandle, char* serverip);  
588 LHController_VDC_API  
    LHController_SetServerSubnetMask(LHController_  
    Handle controllerHandle, char* subnetmask);  
589  
600 LHController_VDC_API  
    LHController_SetServerGateway(LHController_Han  
    dle controllerHandle, char* gateway);  
611 LHController_VDC_API  
    LHController_SetClientIP(LHController_Handle
```

```
    controllerHandle, char* clientIP);
622 | LHController_VDC_API
|     LHController_SetServerPort(LHController_Handle
|         controllerHandle, int serverport);
634 | LHController_VDC_API
|     LHController_SetClientPort(LHController_Handle
|         controllerHandle, int clientport);
651 | LHController_VDC_API
|     LHController_ReadTCPIP(LHController_Handle
|         controllerHandle,
652 |         char *serverIP,
653 |         char *serverSubnetMask,
654 |         char *serverGateway,
655 |         char *clientIP,
656 |         int *serverport,
657 |         int *clientport);
669 | LHController_VDC_API
|     LHController_Readversion(LHController_Handle
|         controllerHandle, char* version);
670 |
671 |
672 |
673 |
685 | LHController_VDC_API
|     LHController_Hig_SetTriggerWidth(LHController_
|         Handle controllerHandle, int channelIndex, int
|         triggerWidth);
686 |
687 |
699 | LHController_VDC_API
|     LHController_Hig_ReadTriggerWidth(LHController_
|         Handle controllerHandle, int channelIndex,
|         int* triggerWidth);
700 |
713 | LHController_VDC_API
|     LHController_Hig_TimeTrobeTrigger(LHController_
|         Handle controllerHandle, int channelIndex,
```

```
    int triggerWidth);  
714  
715  
727 LHController_VDC_API  
    LHController_SetCompleteSignal(LHController_Ha  
    ndle controllerHandle, int triggerWidth);  
728  
729  
741 LHController_VDC_API  
    LHController_ReadCompleteSignal(LHController_H  
    andle controllerHandle, int *triggerWidth);  
742  
743  
755 LHController_VDC_API  
    LHController_SetDelayTime(LHController_Handle  
    controllerHandle, int triggerWidth);  
756  
768 LHController_VDC_API  
    LHController_ReadDelayTime(LHController_Handle  
    controllerHandle, int *triggerWidth);  
769  
781 LHController_VDC_API  
    LHController_SetIntervalTime(LHController_Han  
    dle controllerHandle, int triggerWidth);  
782  
783  
795 LHController_VDC_API  
    LHController_ReadIntervalTime(LHController_Han  
    dle controllerHandle, int *triggerWidth);  
796  
797  
798
```