



# Contents

- Overview..... 3
- SDK Package .....4
  - 1. Sample execution file .....4
  - 2. Sample source code; .....4
  - 3. Sample description .....4
- (II) SDK directory content.....4
  - 1. CoolerMaster LED Table.xls: LED Matrix (6 row x 22 Column) .....4
  - 2. SDK Function.doc: function instructions provided .....4
  - 3. x86 directory: dynamic link libraries, Lib file and header file.....4
- Requirements.....5
  - Supported Devices .....5
  - Keyboards .....5
  - Mouse .....5
- Reference.....6
  - SDK Related Definitions .....6
    - LED Matrix SIZE .....6
    - struct KEY\_COLOR { .....6
    - struct COLOR\_MATRIX {.....7
    - enum EFF\_INDEX {.....7
  - System data related function .....8
    - TCHAR \* GetNowTime().....8
    - LONG GetNowCPUUsage().....8



DWORD GetRamUsage() .....9

float GetNowVolumePeekValue() .....9

Device operation function .....10

Void SetControlDevice(DEVICE\_INDEX devIndex) .....10

bool IsDevicePlug().....11

LAYOUT\_KEYBOARD GetDeviceLayout().....12

bool EnableLedControl(bool bEnable) .....12

bool SwitchLedEffect(EFF\_INDEX iEffectIndex) .....13

bool SetFullLedColor(BYTE r, BYTE g, BYTE b ).....13

bool SetAllLedColor(COLOR\_MATRIX colorMatrix ) .....14

bool SetLedColor(int iRow, int iColumn, BYTE r, BYTE g, BYTE b ) .....14

SDK example .....15



**Make It Yours.**

# Overview

The Cooler Master Maker Toolbox is a software development kit that gives you complete access to the code behind LED lighting on all of our peripheral products. With the controls in your hands and the ability to retrieve system data from your PC, your lighting can be programmed to change, for example, according to the music you play or the speed of your processor. Create a whole host of effects using basic C++ knowledge.

Share your profiles with other Cooler Master peripheral owners at <http://makerhub.coolermaster.com>



**Make It Yours.**

# SDK Package

## **Example Folder:**

### **(I) Example directory content**

1. Sample execution file
2. Sample source code;
3. Sample description

### **(II) SDK directory content**

1. CoolerMaster LED Table.xls: LED Matrix (6 row x 22 Column)
2. SDK Function.doc: function instructions provided
3. x86 directory: dynamic link libraries, Lib file and header file



**Make It Yours.**

## Requirements

**Windows 7 (32-bit and 64-bit);**

**Windows 8.1 (32-bit and 64-bit);**

**Windows 10 (32-bit and 64-bit).**

## Supported Devices

### Keyboards

**MasterKeys Pro L**

**MasterKeys Pro S**

**MasterKeys Pro L White**

**MasterKeys Pro M White**

**MasterKeys Pro S White**

**MasterKeys Pro M**

### Mouse

**MasterMouse Pro L**

**MasterMouse S**



Make It Yours.

# Reference

## SDK Related Definitions

LED Matrix SIZE

MAX\_LED\_ROW:6

MAX\_LED\_COLUMN:22

```
struct KEY_COLOR {
```

```
    BYTE r;
```

```
    BYTE g;
```

```
    BYTE b;
```

```
};
```



```
struct COLOR_MATRIX {
```

Description: set/store entire LED Color structure

```
KEY_COLOR KeyColor[MAX_LED_ROW][MAX_LED_COLUMN];
```

```
};
```

```
enum EFF_INDEX {
```

Description: set/store entire LED Color structure Special effects list

```
EFF_FULL_ON = 0, EFF_BREATH = 1,
```

```
EFF_BREATH_CYCLE = 2 , EFF_SINGLE = 3,
```

```
EFF_WAVE = 4, EFF_RIPPLE = 5,
```

```
EFF_CROSS = 6, EFF_RAIN = 7,
```

```
EFF_STAR = 8, EFF_SNAKE = 9,
```

```
EFF_REC = 10, EFF_MULTI_1 = 0xE0,
```

```
EFF_MULTI_2 = 0xE1, EFF_MULTI_3 =
```

```
0xE2, EFF_MULTI_4 = 0xE3, EFF_OFF = 0xFE};
```



Make It Yours.

## System data related function

TCHAR \* GetNowTime()

Description: Obtain current system time

Function name : GetNowTime

variable:

return : TCHAR : string index format is %Y %m/%d %H:%M %S

note:

LONG GetNowCPUUsage()

Description: obtain current CPU usage ratio

Function name: GetNowCPUUsage

variable :

returns : LONG : 0 ~ 100 integer

note :





Make It Yours.

DWORD GetRamUsage()

Description: Obtain current RAM usage ratio

Function name: GetRamUsage

variable :

returns : DWORD : 0 ~ 100 integer

note :

float GetNowVolumePeekValue()

Description: Obtain current volume

Function name : GetNowVolumePeekValue

variable :

returns : float : 0 ~ 1 float number

note :



## Device operation function

Void SetControlDevice(DEVICE\_INDEX devIndex)

Description: set operating device

Function name: SetControlDevic

variable : DEVICE\_INDEX: device list

DEV\_MKeys\_L,

DEV\_MKeys\_S,

DEV\_MKeys\_L\_White,

DEV\_MKeys\_M\_White,

DEV\_MMouse\_L,

DEV\_MMouse\_S,

DEV\_MKeys\_M

DEV\_MKeys\_S\_White

returns :

note :



**Make It Yours.**

bool IsDevicePlug()

Description: verify if the device is plugged in

Function name : IsDevicePlug

variable :

returns: bool : true plugged in , false not plugged in

note :



## LAYOUT\_KEYOBARD GetDeviceLayout()

Description: Obtain current device layout

Function name: GetDeviceLayout

variable :

returns: LAYOUT\_KEYOBARD List:

currently 3 LAYOUT\_UNINIT ,

LAYOUT\_US , LAYOUT\_EU

note:

## bool EnableLedControl(bool bEnable)

Description: set control over device's LED

Function name: EnableLedControl

variable : bool bEnable:

true Controlled by SW,

false Controlled by FW,

returns : bool : true Success,false Fail

note :



Make It Yours.

`bool SwitchLedEffect(EFF_INDEX iEffectIndex)`

Description: switch device current effect

Function name: SwitchLedEffect

variable : EFF\_INDEX iEffectIndex: index value of the effect

returns : bool : true Success , false Fail

note :

`bool SetFullLedColor(BYTE r, BYTE g, BYTE b )`

Description: set entire keyboard LED color

Function name : SetFullLedColor

variable : BYTE r :red, BYTE g :green, BYTE b :blue

returns : bool : true Success , false Fail

note :



`bool SetAllLedColor(COLOR_MATRIX colorMatrix )`

Description: Set Keyboard "every LED" color

Function name: SetAllLedColor

variable : COLOR\_MATRIX colorMatrix:structure,

fill up RGB value according to LED Table

returns : bool : true Success,false Fail

note :

`bool SetLedColor(int iRow, int iColumn, BYTE r, BYTE g, BYTE b )`

Description: Set single Key LED color

Function name: SetLedColor

variable: int iRow: row,

int iColumn:column BYTE r :red,

BYTE g :green, BYTE b :blue

returns : bool : true Success , false Fail

note :

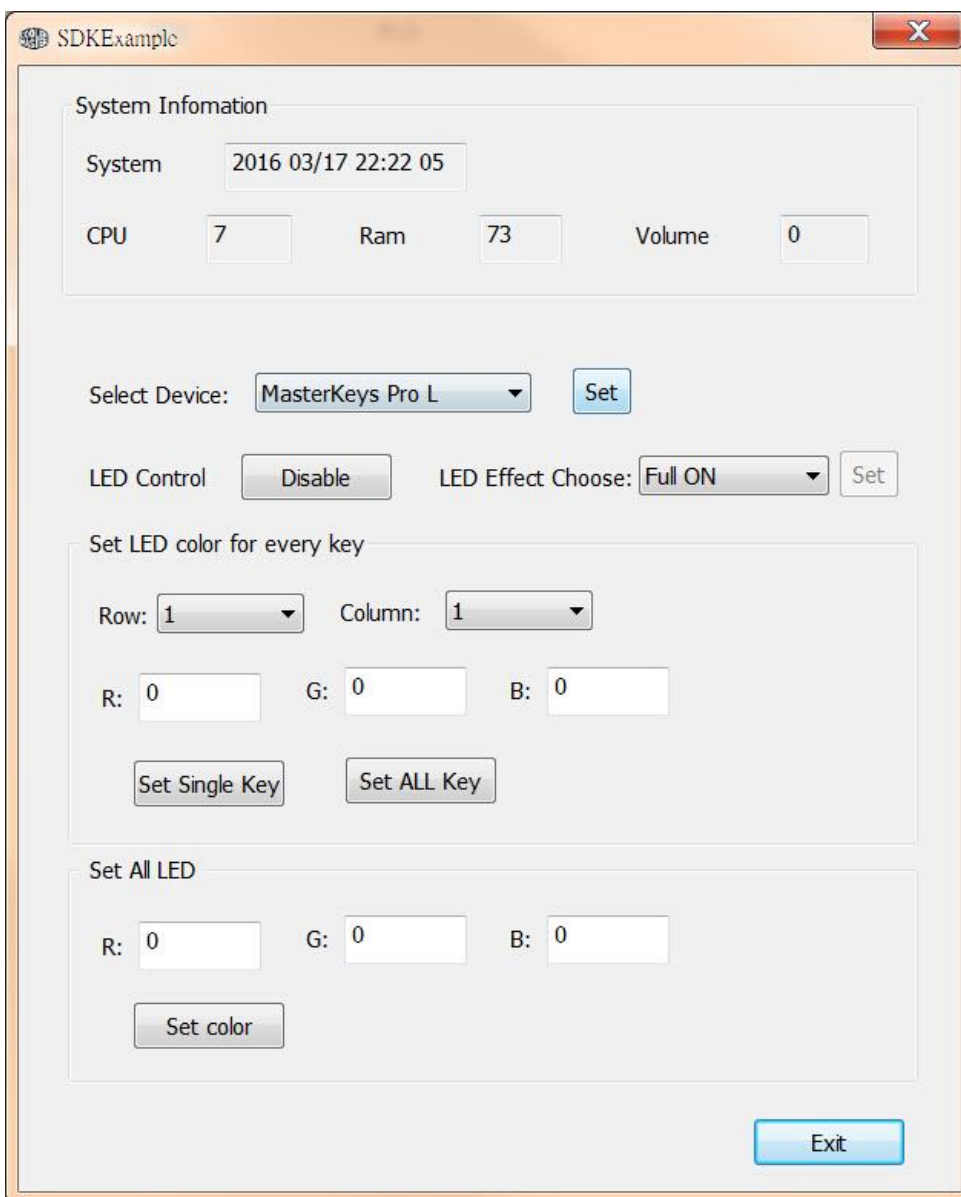


Make It Yours.

# SDK example

SDK examples illustrate

Examples UI screen





Make It Yours.

Development environment: VC ++ 2008

1. System Information: for the user's computer to fetch the local system time, CPU usage, memory usage percentage, the current playback volume percentage.
2. Select Device: to select the device that you want to control, the default option is MasterKeys Pro L.
3. LED Control: can choose from enable and disable, in the disable state can switch effects; in the enable state can setup the keyboard LED color
4. Set LED Color for every Key: is allowed to set different colors of each key, there are two ways to set up. One is to set a single Key; the other one is to set all keys on the keyboard to specified / different color. Please use the drop-down menu and select Row Column with "CoolerMaster LED Table.xls" table to determine the location specified color.
5. Set All Led: set the whole keyboard as a single color quickly.