

Monarch Pro – UNS310709 Windows API Manual - C -

[Download the library from – Monarch PRO Camera Library C](#)

TOC

Download the library from – Monarch PRO Camera Library C1

1).	Connect	2
2).	Control methods:	3
3).	LUT methods:	4
4).	Camera methods	6
5).	Other methods	7
6).	Sample Code	8
7).	Files provided:	9

1). Connect

1.1). init_controller

initialize camera module controller:

```
setup H/W's gpio control setting,  
clear LUT,  
load LUT from flash,  
set sensor's exposure time & gain to default,  
disable video streaming output,  
enable DAC power,  
write all DACs to zero.
```

1.1.1). input: ---

1.1.2). output: ---

1.1.3). example: `init_controller()`

1.2). Reset

reset camera module controller.

```
init_controller + reset few inner variables.
```

1.2.1). input: ---

1.2.2). output: ---

1.2.3). example: `Reset()`

2). Control methods:

2.1). SetExposureFps

Set fps value to camera module in order to control exposure time.

Note: $\text{fps} = \text{frames per seconds} = 1000/\text{exposure_time}$

2.1.1). input: fps (2-1000)

2.1.2). output: ---

2.1.3). example: `SetExposureFps(100)`

2.2). GetExposureTime

Get the exposure time in milliseconds.

2.2.1). input: ---

2.2.2). output: exposure time

2.2.3). example: `exp_time = GetExposureTime()`

2.3). SetGain

Set gain value to camera module.

2.3.1). input: gain value (1-10 float)

2.3.2). output: ---

2.3.3). example: `SetGain(2.6)`

2.4). GetGain

Get the gain value.

2.4.1). input: ---

2.4.2). output: gain value

2.4.3). example: `gain_value = GetGain()`

3). LUT methods:

LUTLine = ([v1, v2, v3, v4], data1, data2, band)

Example: ([2303, 4023, 1094, 2673], 10, 1, 810)

3.1). EnableLowPower

Disable/enable video-stream output from camera module.

3.1.1). input: True/False

3.1.2). output: ---

3.1.3). example: EnableLowPower(False)

3.2). SetLUTIndex

Select the current line's index to get that line data with GetLUTLine or Set the index of the last line to be performed while RunLUT is running.

3.2.1). input: line index

3.2.2). output: ---

3.2.3). example: SetLUTIndex(5)

3.3). GetLUTLine

Read the line - number given by index - in LUT sequence that store it in camera module.

Note: always use SetLUTIndex before this function.

3.3.1). input: line index, converted voltages vector (4), d1, d2, cwl :
unsigned short c1[4];
unsigned short d1;
unsigned short d2;
unsigned short cwl;

3.3.2). output: converted voltages vector (4), fps, gain, cwl. (False, if failed).

3.3.3). example: `line_data = GetLUTLine(5, c1, &d1, &d2, &cwl)`

3.4). SetLineVoltages

Set filter mode of desired band from LUT.

Lines indexes range: 0-9.

After this function, takeFrame will provide the right filtered frame.

3.4.1). input: line index

3.4.2). output: ---

3.4.3). example: SetLineVoltages(4)

3.5). CaptureLUT

Capture spectral cube.

Perform camera module LUT and store all the frames in a dedicated array.

Note: By default, this function set to run 0-9 indexes (10 bands).

For using less than 10 bands, use SetLUTIndex before this function so it will run until the line's index that has been set with SetLUTIndex is reached.

3.5.1). input: 1, unsigned short LUT_frames[10*1024*1280]={0};

3.5.2). output: array of 10-bit RAW frames

3.5.3). example: CaptureLUT(1, LUT_frames)

3.6). BuildCustomLUT

Build the current LUT to be performed by RunLUT

3.6.1). input: int idxList[10] = { 5,3,6,4,0 }; length = 5;

3.6.2). output: ---

3.6.3). example: BuildCustomLUT(idxList, length)

3.7). RestoreLUT

Restore LUT from memory to be the current LUT.

Note: needed only in case that current LUT is different than the saved LUT in camera module.

3.7.1). input: ---

3.7.2). output: ---

3.7.3). example: RestoreLUT()

4). Camera methods

4.1). camClose

Release (close) camera connection

4.1.1). input: --- , Output: ---

4.1.2). example: `camClose()`

4.2). waitKeyX

set time (in milliseconds) for frame to be displayed

Note: needed only if using `cv2.imshow()`

4.2.1). input: number

4.2.2). output: ---

4.2.3). example: `waitKeyX(10)`

4.3). takeFrame

Capture frame from camera and process it.

usage: before calling `getFrame1()` or `getFrame2()`

4.3.1). input: 0

4.3.2). output: ---

4.3.3). example: `takeFrame(0)`

4.4). getFrame0

Get the raw image from camera. note: no need for takeFrame().

4.4.1). input: unsigned short `data_raw[1024][1280]`

4.4.2). output: RAW frame of 16-bit

4.4.3). example: `frame0 = getFrame0(data_raw)`

4.5). getFrame1

Get grayscale frame of 10-bit in 16-bit package.

This is the processed raw image from camera.

usage: for raw and envt formats.

Note: user should apply `getFrame(0)` before

4.5.1). input: unsigned short `data16[1024][1280]`

4.5.2). output: grayscale frame of 10-bit

4.5.3). example: `frame16 = getFrame1(data16)`

4.6). getFrame2

Get grayscale frame of 8-bit.

This is the processed grayscale image from camera.

usage: for display and png format.

Note: user should apply `getFrame(0)` before

4.6.1). input: unsigned char `data8[1024][1280]`

4.6.2). output: grayscale frame of 8-bit

4.6.3). example: `frame8 = getFrame2(data8)`

5). Other methods

5.1). PowerLED

Toggle LED light on camera module PCB.

5.1.1). input: True/False

5.1.2). output: ---

5.1.3). example: `PowerLED(True)`

5.2). GetSerial

Get serial number of the camera module.

5.2.1). input: ---

5.2.2). output: serial number. (False, if failed)

5.2.3). example: `SN = GetSerial()`

5.3). GetTemperature

Get the temperature value in Celsius.

The temperature sensor is located on the camera module PCB.

5.3.1). input: ---

5.3.2). output: temperature. (False, if failed)

5.3.3). example: `temp = GetTemperature()`

6). Sample Code

For full sample code, please see the open code of TestUCamDLL.exe.

```
#include <iostream>
#include <stdio.h>
#include <conio.h>
#include <string>
#include <cstdio>

#include "../UCamDll/CameraCardController.h"
using namespace std;

unsigned short data_orig[1024][800];
unsigned short data_raw[1024][1280];
unsigned short data16[1024][1280];
unsigned char data8[1024][1280];

int main()
{
    CameraCardController myCamera;
    myCamera.InitCam(myCamera.m_iCamID);
    Sleep(100);
    myCamera.PowerLed(1);

    myCamera.SetExposureFps(100);
    myCamera.SetGain(1.25);

    myCamera.takeFrame(0);
    myCamera.getFrame0(&data_raw[0][0]);
    myCamera.getFrame1(&data16[0][0]);
    myCamera.getFrame2(&data8[0][0]);
}
```


7). Files provided:

7.1). UCamDll.dll

7.2). TestUCamDll.exe

7.3). Header files