ArduCAM USB Camera Shield

Application Note for MT9M001

Rev 1.0, June 2017

Table of Contents

1 Introduction

This user guide describes the detail operation of ArduCAM USB camera for MT9M001. The latest deivce driver, SDK library and examples can be downloaded from the https://github.com/ArduCAM/ArduCAM_USB_Camera_Shield.

2 Hardware Installation
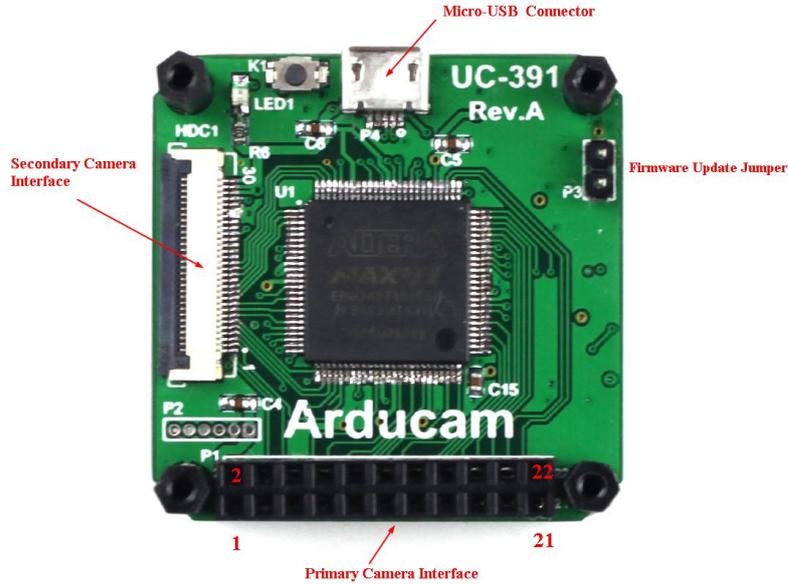
There are two different camera interface provided on the USB camera shield, but only one camera interface can be used at a time. The MT9M001 camera header board should be connected to the primary camera interface and should align the pin 1 of the camera breakout board to the USB camera shield camera connector pin 1.
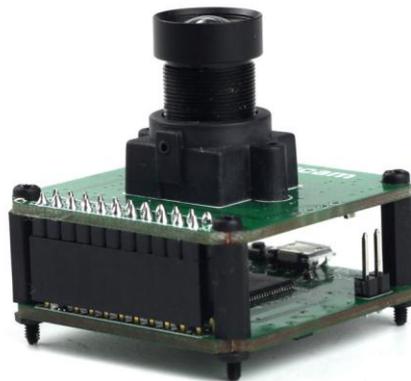
Table 1 P1 Connector Pin Definition

Pin No.	PIN NAME	TYPE	DESCRIPTION
1	VCC	POWER	3.3v Power supply
2	GND	Ground	Power ground
3	SCL	Input	Two-Wire Serial Interface Clock
4	SDA(SDATA)	Bi-directional	Two-Wire Serial Interface Data I/O
5	VS(VSYNC)	Input	Active High: Frame Valid; indicates active frame
6	HS(HREF)	Input	Active High: Line/Data Valid; indicates active pixels
7	PCLK	Input	Pixel Clock output from sensor
8	XCLK	Output	Master Clock into Sensor
9	D7	Input	Pixel Data Output 7 (MSB)
10	D6	Input	Pixel Data Output 6
11	D5	Input	Pixel Data Output 5
12	D4	Input	Pixel Data Output 4
13	D3	Input	Pixel Data Output 3
14	D2	Input	Pixel Data Output 2
15	D1	Input	Pixel Data Output 1
16	D0	Input	Pixel Data Output 0(LSB)
17		NC	
18		NC	
19		NC	
20		NC	
21		NC	
22	Trigger(EXP)	Output	External trigger output

The firmware update jumper should be left open when normal operation.

3 Run the Demo

Plug in the USB cable to the camera and the host PC USB port, and open the Windows demo software. Select the MT9M001 from the Sensor drop down list then click auto-open button.

www.ArduCAM.com
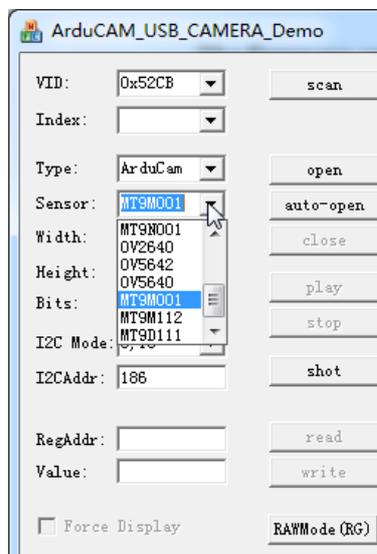
Click play button to run the camera in video mode.
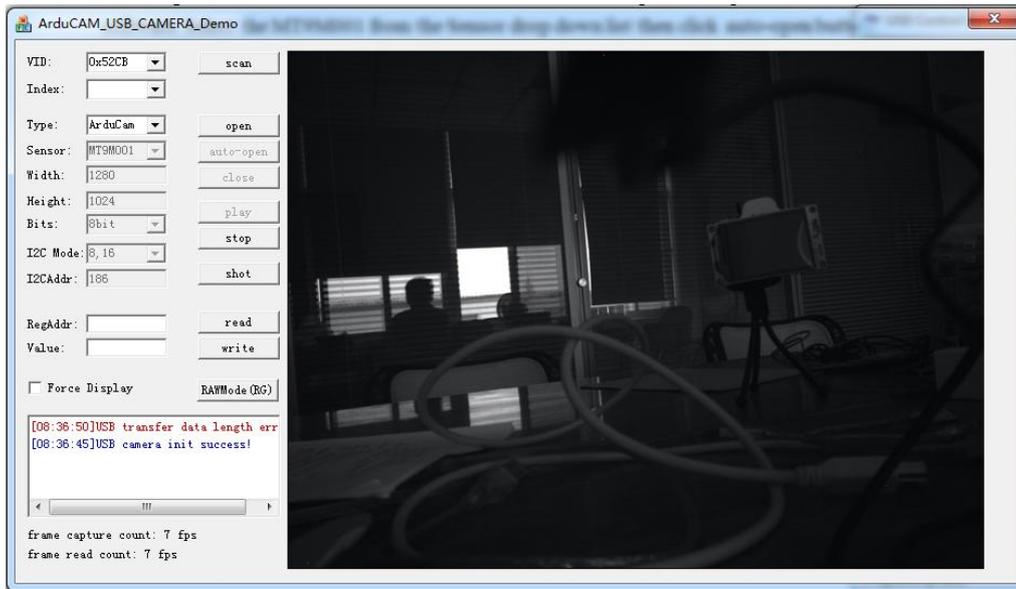
4 Tune the Sensor Registers

4.1 Identify the Sensor Version

Sensor register address 0x00 is read only, and always return the chip vision 0x8431(33841) when read it.

Register # (Hex)	Description	Data Format (Binary)	Default Value (Hex)
0x00	Chip Version	1000 0100 0001 0001	0x8431

Input the register address 0 in decimal to the RegAddr dialog box and click read button, the Value dialog box will show 33841 in decimal which is identical to 0x8431 in hex.

4.2 Adjust the Sensor Gain

The gain is individually controllable for each of the four groups of pixels that lie in odd rows and columns, even rows and columns, odd rows and even columns, and even rows and odd columns. This is shown in the register chart.

Formula for gain setting:

when Gain $\leqslant$ 8:　　Gain = (bit[6] + 1) x (bit[5-0] x 0.125)

when Gain > 8 and (bit[6] = 1 and bit[5] = 1):　　Gain = 8.0 + bit[2-0]

Since bit[6] of the gain registers are multiplicative factors for the gain settings, there are alternative ways of achieving certain gains. Some settings offer superior noise performance to others, despite the same overall gain. The following lists the recommended gain settings:

Gain	Increments	Recommended Settings
1.000 to 4.000	0.125	0x08 to 0x20
4.25 to 8.00	0.25	0x51 to 0x60
9.0 to 15.0	1.0	0x61 to 0x67

There are 5 register below control the gain settings.

Register	Bit	Description
0x2B	6:0	Even row, even column—default = 0x08 (8) = 1x gain.
0x2C	6:0	Odd row, even column—default = 0x08 (8) = 1x gain.
0x2D	6:0	Even row, odd column—default = 0x08 (8) = 1x gain.
0x2E	6:0	Odd row, odd column—default = 0x08 (8) = 1x gain.
0x35	6:0	Global gain—default = 0x08 (8) = 1x gain. This register can be used to set all four gains at once.

To simplify the demo, we will change the 0x35(53) global gain register.
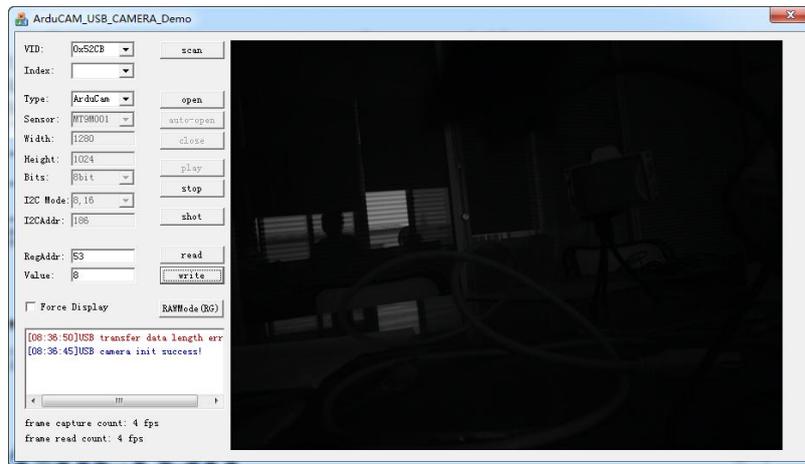
RegAddr = 53, Value = 8
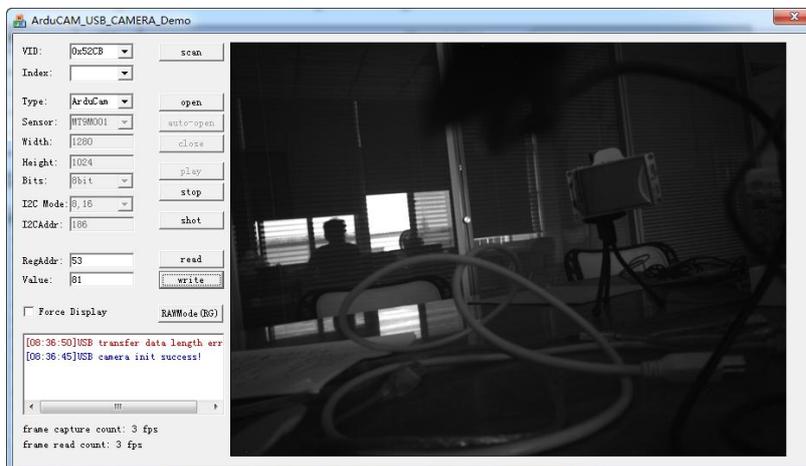
RegAddr = 53, Value = 81
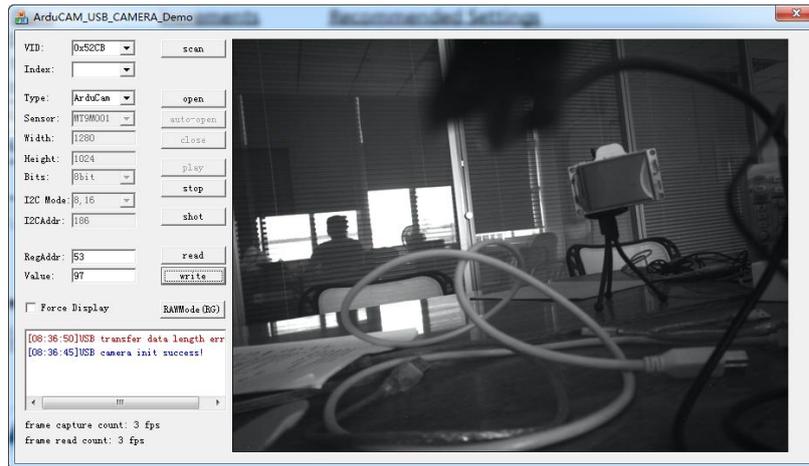
RegAddr = 53, Value = 103

4.3 Adjust the Sensor Exposure

The exposure is also called Pixel Integration Control. The MT9M001 is rolling shutter so the exposure is line basis, They do exposure line by line, so all pixels in a line are exposed at the same time, but different pixels from different line are exposed at different time.
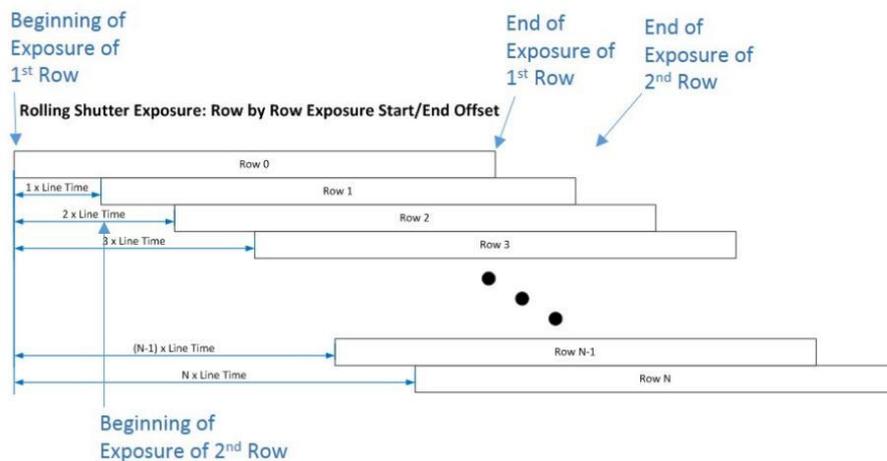
In this case, the minimum exposure time is 1 *Line Time*, and the maximum exposure time is one *Frame* time. Basically it equals to total vertical resolution times 1 *Line Time,* but sometimes there are dummy lines and blanking period added to the frame.

These registers (along with the window sizing and blanking registers) control the integration time for the pixels.

The actual total integration time (tINT) is:

tINT = Reg0x09 x row time - overhead time - reset delay, where:

Row time = ((Reg0x04 + 1) + 244 + Reg0x05 - 19) pixel clock periods

Overhead time = 180 pixel clock periods

Reset delay = 4 x Reg0x0C pixel clock periods

If the value in Reg0x0C exceeds (row time - 548)/4 pixel clock cycles, the row time will be extended by (4 x Reg0x0C - (row time - 548)) pixel clock cycles.

In this expression, the row time term, Reg0x09 x ((number of columns) + 244 + horizontal blanking register - 19), corresponds to the number of rows integrated. The overhead time (180

pixel clocks) is the overhead time between the READ cycle and the RESET cycle, and the final term is the effect of the reset delay.

Typically, the value of Reg0x09 is limited to the number of rows per frame (which includes vertical blanking rows) such that the frame rate is not affected by the integration time. If Reg0x09 is increased beyond the total number of rows per frame, the MT9M001 will add additional blanking rows as needed. A second constraint is that tINT must be adjusted to avoid banding in the image from light flicker. Under 60Hz flicker, this means tINT must be a multiple of 1/120 of a second. Under 50Hz flicker, tINT must be a multiple of 1/100 of a second.

Also for simplify we only change the 0x09 register to demonstrate how manual exposure works.

0x09	13:0	Number of rows of integration—default = 0x0419 (1049).

When global gain setting is 103, we change the 0x09 register as follows, we can get different exposure result as follows.
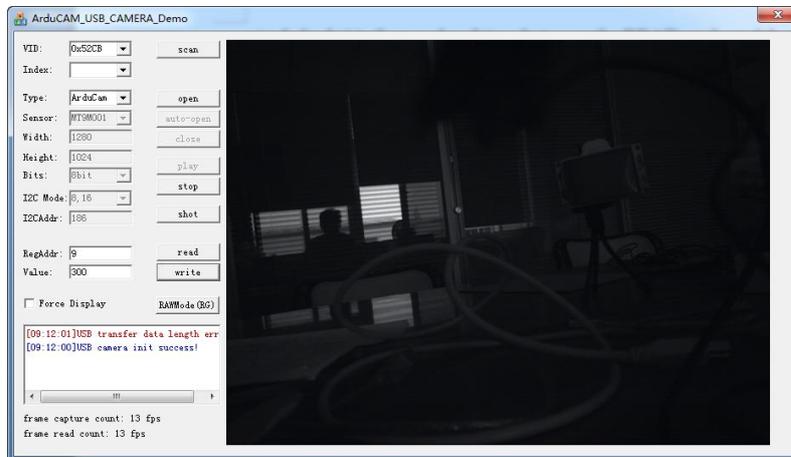
RegAddr = 9, Value = 300
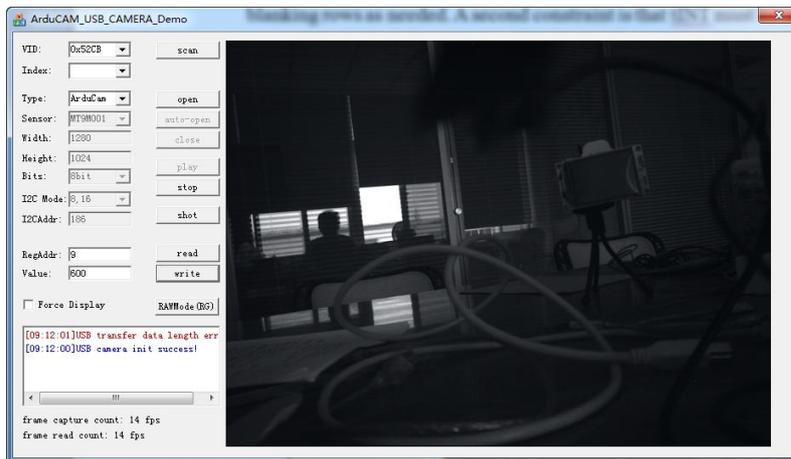
RegAddr = 9, Value = 600

RegAddr = 9, Value = 900