ArduCAM-Mini-5MP-Plus
OV5642 Camera Module

5MP SPI Camera Hardware Application Note

Rev 1.0, Oct 2016

Table of Contents

1 Introduction

This application note describes the detail hardware operation of ArduCAM-Mini-5MP-Plus OV5642 camera module. For software operation please refer to ArduCAM-Mini-5MP-Plus software application note.

2 Pin Definition

Table 1 ArduCAM-M-5MP Pin Definition

Pin No.	PIN NAME	TYPE	DESCRIPTION
1	CS	Input	SPI slave chip select input
2	MOSI	Input	SPI master output slave input
3	MISO	Output	SPI master input slave output
4	SCLK	Input	SPI serial clock
5	GND	Ground	Power ground
6	VCC	POWER	3.3V~5V Power supply
7	SDA	Bi-directional	Two-Wire Serial Interface Data I/O
8	SCL	Input	Two-Wire Serial Interface Clock

3 Typical Wiring

3.1 Single Camera Wiring

The typical connection between ArduCAM module and Arduino or etc platform is shown in the Figure 1. More typically the Figure 2 shows the wiring for Arduino UNO R3 board.

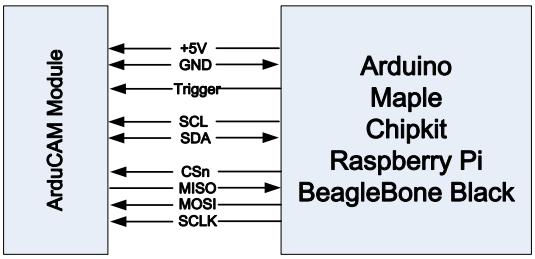

Figure 1 Typical Wiring

3.2 Multi Cameras Wiring

The multi-cameras connection between ArduCAM module and Arduino or etc platform is shown in the Figure 3. Please note that the 5MP-Plus camera uses massive power, so connecting multiple cameras you should use external power supply.

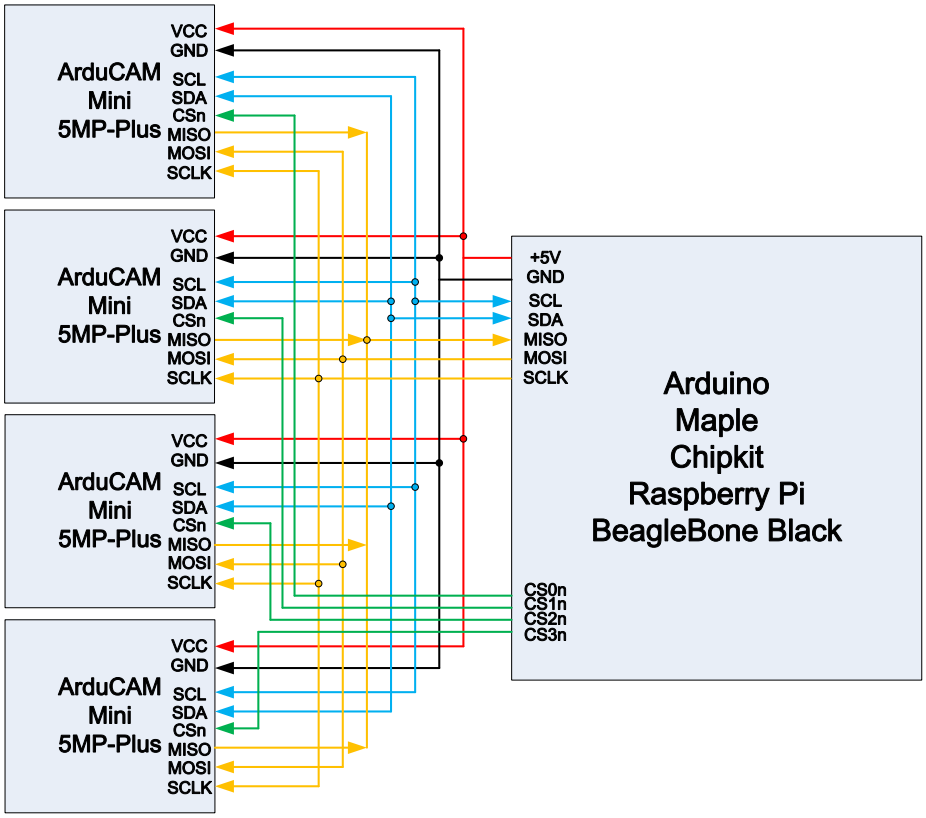

Figure 3 Multi-Cameras Wiring

4 I2C Interface

The I2C interface is directly connected to the image sensor OV5642. The OV5642 I2C slave address is 0x78 for write and 0x79 for read. User can use I2C master to read and write all the registers in the OV5642 sensor. For more information about the OV5642 register, please refer the OV5642 datasheet. The Figure 4 shows writing value 0x80 to the OV5642 register 0x3008. The Figure 5 shows reading value 0x56 from the OV5642 register 0x300A.

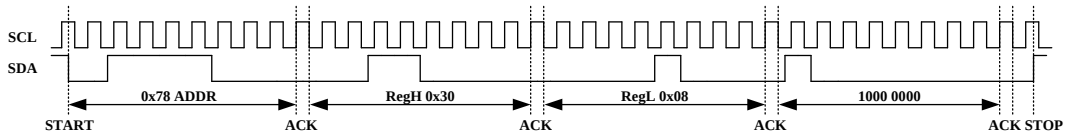

Figure 4 I2C Write Bus Timing

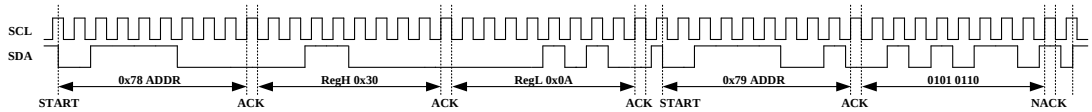

Figure 5 I2C Read Bus Timing

5 SPI Slave Interface

The ArduCAM SPI slave interface is fixed SPI mode 0 with POL = 0 and PHA = 0. The stable SCLK speed is 8MHz, care should be taken when over clock the SPI bus speed. The SPI protocol is designed with a command phase with variable data phase. The chip select signal should always keep asserted during the SPI read or write bus cycle.

The first bit[7] of the command phase is read/write byte, ‘0’ is for read and ‘1’ is for write, and the bit[6:0] is the address to be read or write in the data phase. ArduChip register table see Table 1.

6 External Trigger

External hardware trigger input can be used to start a capture manually, software part only needs to polling the "capture done" status bit before reading the image data. The Figure 6 shows the trigger pin which is marked with "T" at the back of the module, and the trigger input is active high.

Figure 6 External Trigger Input

7 ArduChip Timing Diagram

7.1 SPI Bus Write Timing

The SPI bus write timing composed of a command phase and a data phase during the assertion of the chip select signal CSn. The first 8 bits is command byte which is decoded as a register address, and the second 8 bits is data byte to be written to the ArduChip internal registers.

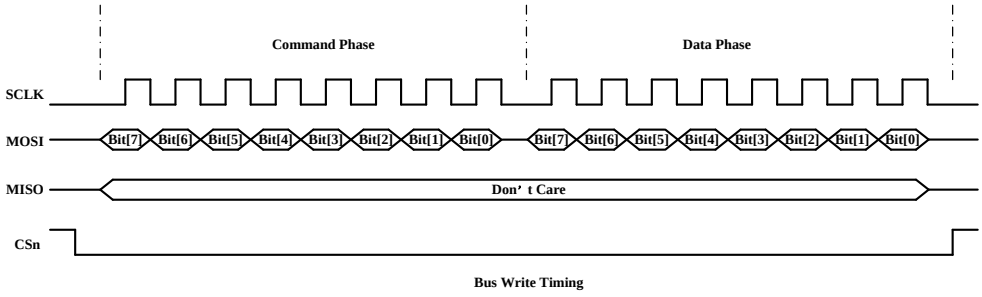

Figure 7 SPI Bus Write Timing

7.2 SPI Bus Single Read Timing

The SPI bus single read timing is for read operation of ArduChip internal registers and single FIFO read function. It is composed of a command phase and a data phase during the assertion of chip select signal CSn. The first 8 bits is command byte which is decoded as a register address, the second 8 bits is dummy byte written to the SPI bus MOSI signal, and the content read back from register is appeared on the SPI bus MISO signal.

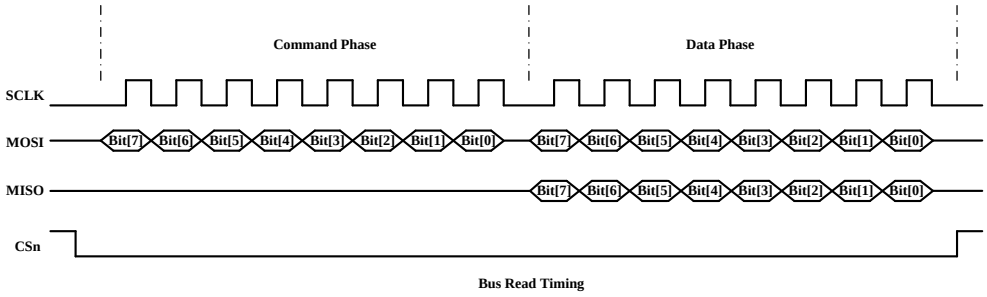

Figure 8 SPI Bus Single Read Timing

7.3 SPI Bus Burst Read Timing

The SPI bus burst read timing is only for burst FIFO read operation. It is composed of a burst read command phase and multiple data phases in order to get double throughput compared to the single FIFO read operation.

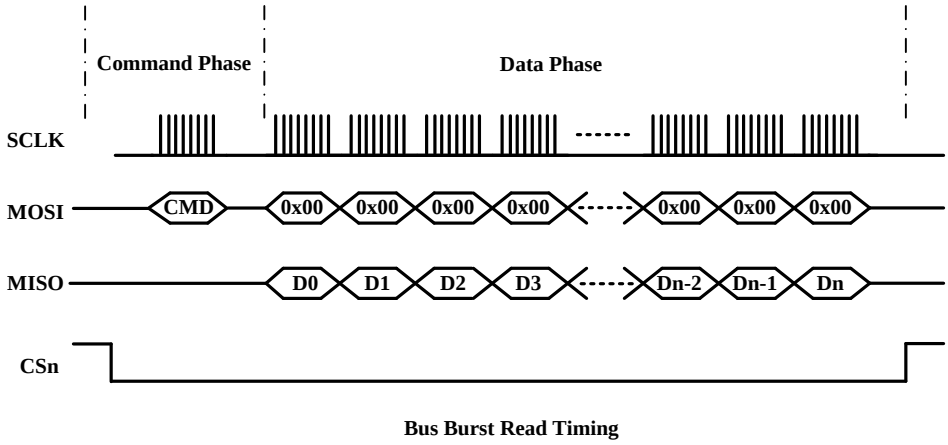

Figure 9 SPI Bus Burst Read Timing

If user want to break up the burst transaction by multiple burst read. Please note that do not use other SPI command between burst read transactions, it will cause the image data lost. Detail timing can be found from Figure 10.

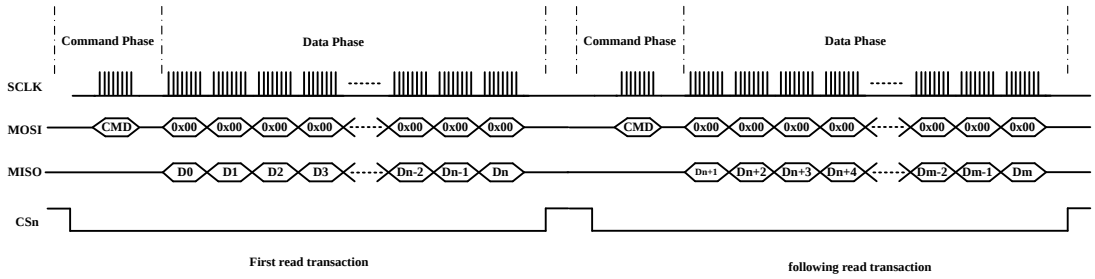

Figure 10 Multiple burst read timing diagram

8 Registers Table

Sensor and FIFO timing is controlled with a set of registers which is implemented in the ArduChip. User can send capture commands and read image data with a simple SPI slave interface. The detail description of registers' bits can be found in the software section in this document.

As mentioned earlier the first bit[7] of the command phase is read/write byte, '0' is for read and '1' is for write, and the bit[6:0] is the address to be read or write in the data phase. So user has

to combine the 8 bits address according to the read or write commands they want to issue.

Table 2 ArduChip Register Table

Register Address bit[6:0]	Register Type	Description
0x00	RW	Test Register
0x01	RW	Capture Control Register Bit[2:0]: Number of frames to be captured, the value 0~6 means taking 1~7 photos, the value 7 means continuous capture until the 8MByte frame buffer is full
0x02	RW	Reserved
0x03	RW	Sensor Interface Timing Register Bit[0]: Sensor Hsync Polarity, 0 = active high, 1 = active low Bit[1]: Sensor Vsync Polarity 0 = active high, 1 = active low Bit[3]: Sensor PCLK reverse 0 = normal, 1= reversed PCLK
0x04	RW	FIFO control Register Bit[0]: write '1' to clear FIFO write done flag Bit[1]: write '1' to start capture Bit[4]: write '1' to reset FIFO write pointer Bit[5]: write '1' to reset FIFO read pointer
0x05	RW	Reserved
0x06	RW	GPIO Write Register Bit[0]: Sensor reset IO value Bit[1]: Sensor power down IO value Bit[1]: Sensor power enable IO value
0x3B	RO	Reserved
0x3C	RO	Burst FIFO read operation
0x3D	RO	Single FIFO read operation
0x3E	RO	Reserved
0x3F	RO	Reserved
0x40	RO	ArduChip firmware version, constant value 0x62 for 5MP-Plus model Bit[7:4]: integer part of the revision number Bit[3:0]: decimal part of the revision number
0x41	RO	Bit[0]: camera vsync pin realtime status Bit[3]: camera capture done flag
0x42	RO	Camera write FIFO size[7:0]
0x43	RO	Camera write FIFO size[15:8]
0x44	RO	Camera write FIFO size[22:16]
0x45	RO	Reserved