

To develop an interface among C3088 camera and computer using AVR microcontroller

Ramandeep Kaur

Lecturer in ECE (Shaheed Bhagat Singh State Technical Campus, Ferozepur, India)

Abstract : The aim of this project is the development and construction of an interface between a CMOS camera and computer. C3088 smart camera with CMOS image sensor O6620 is chosen as the main component of this product. Main advantage of this camera module is that it has digital output and digital video port that supplies a continuous 8-16 bit range of data stream. This interface allows a user to get images from the camera, to change some of the properties of the camera as brightness, luminance, etc. Digicam passes the data to controller through I2C protocol. The interface was implemented using the Atmel AVR ATmega16 microcontroller. Servo motor is also added to the system to get panoramic images.

Keywords: AVR microcontroller, C3088 camera, computer, Image, Interface.

I. Introduction

The aim of the project is the design and construction of an interface between the CMOS camera c3088 and a computer, using the AVR microcontroller. The information flows in two ways: on the one hand there are commands from the computer to the camera to change different characteristics of it, on the other images from the camera should be sent to the computer. The communication between the computer and the AVR is through the serial port. The communication between the camera and the microcontroller is: using the I2C protocol to access to the different registers of the camera; and using an 8 bit port to read the images. In addition the camera will be connected to a TV with its analog output for debugging purpose.

The images from the camera are read following the Zoom Video Port Format implemented by the camera. The images read are in black and white because only the Y channel is read. To control all this communications the microcontroller is connected to the computer via serial port. The microcontroller read commands from the computer and performs the appropriate actions. The serial port is also used to receive the images in BMP format

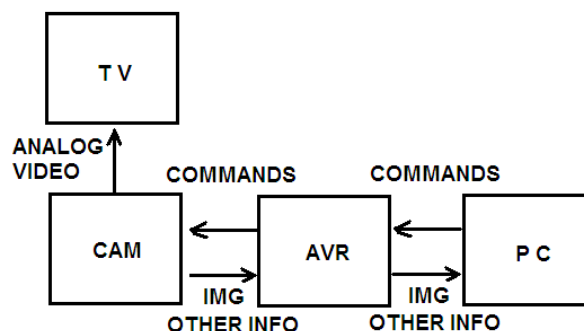


Figure 1.1 Block diagram of system

A complete schematic view of system is shown in Fig 1.2. The only voltage used is $V_{cc} = 5V$. There are three modules in system design. The first part of system in which camera module is interfaced to AVR controller is shown.

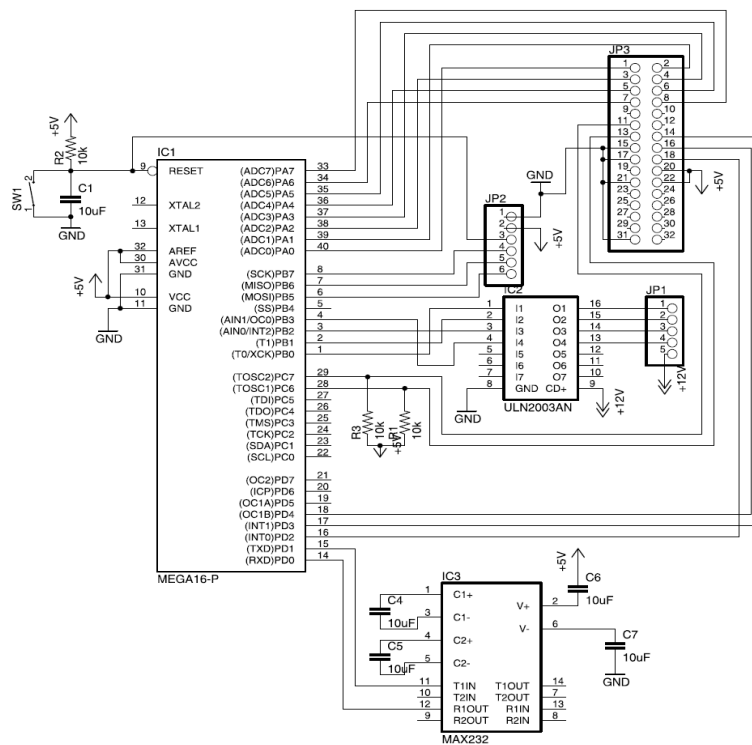


Figure 1.2 Schematic diagram

In this module color camera C3088 sensor with digital output, is used to capture the image. This is interfaced with AVR controller Atmega16. PC sends commands to camera through AVR using serial port, AVR controls the camera using I²C interface to capture the image. Then camera sends image to AVR in bmp format which is directly send to PC by AVR controller. Interfacing of servo can also be seen in Fig 1.2. It is connected to pin PB0-PB-3, because this pin is connected to the comparator of the counter that is used to give the position of the servo. Vcc and ground are also present in the connector of the servo. The C3088 camera is explained in next section.

II. The C3088 Camera

The C3088 is the camera used in the project and it is the main part of it that is the reason that a short explanation is made. It is a color camera module with digital output [1]. It uses a CMOS image sensor OV6620 from Omnivision. C3088 color camera module is shown in Fig 2.1.



Figure 2.1 C3088 camera module

It has a digital video port that supplies a continuous 8/16 bit-wide image data stream. All the camera functions, such as exposure, gamma, gain, white balance, windowing, can be changed through I2C interface by writing in some registers. The video output can be expressed in different formats, and with different type of channels (RGB, UVY). The format used in the project is the simplest one: Zoom Video Port Format. In this format 8 bits represent the intensity (Y channel) of one pixel, the other 8 bits represent the U and V channels, but are not used. The information is sent continuously and is synchronized by the HREF, VSYNC and PCLK signals as showed in the next graph in Fig 2.2. The VSYNC signal indicates new frame, the HREF signal indicates when the information is valid and makes the horizontal synchronization, and PCLK is the clock signal (the data is valid in the rising edge). The period of PCLK can be changed by writing in the registers of the

camera. This will allow reading images from the camera directly with the microcontroller without the use of additional hardware. The registers accessed by the I2C bus are 80 and as it was said above allow changing different properties of the camera. In this project they are used to change the period of PCLK, to read the size of the image, to make a mirror of the image, and to reset the camera. It is important here to make a small clarification about the I2C protocol in the camera. The datasheet of the C3088 camera says that the bus used is I2C, but nothing says about the registers and the addresses that should be used. So it is necessary to read the datasheet of the OV6620 chip. There you can find that the protocol used for the communication is not I2C, it is SCCB [2]. If you read the specifications of this protocol you can see that it is for some modes the same as I2C protocol. The ID address of the camera is 0xC0 for writing and 0xC1 for reading, as is said in the datasheet of the OV6620 [3].

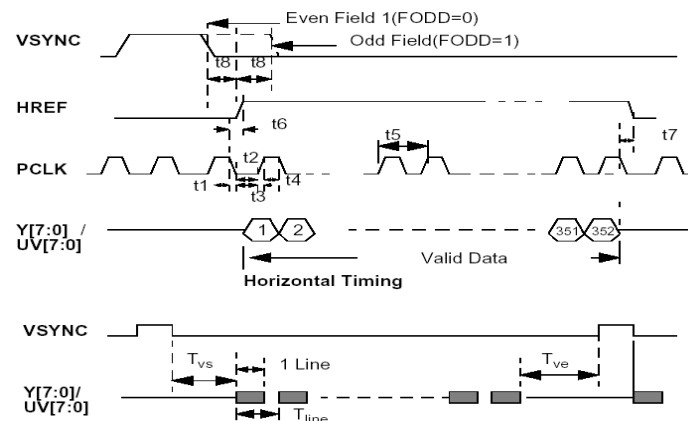


Figure 2.2 Output signals from the C3088

Features

- 101,376 pixel, CIF/QCIF format
- Small size: 40 x 28 mm
- Lens: f=4.9mm (Optional)
- 8/16 bit video data: CCIR601, CCIR656, ZV port
- Read out - progressive
- Data format -YCrCb 4:2:2, GRB 4:2:2, RGB
- I2C interface
- Wide dynamic range, anti blooming, zero smearing
- Electronic exposure / Gain / White balance control
- Image enhancement - brightness, contrast, gamma,
- Saturation, sharpness, window, etc
- Internal / external synchronization scheme
- Frame exposure / line exposure option
- Single 5V operation
- Low power consumption (<100mW)
- Monochrome composite video signal output(50 Hz)

Pin Description

- 1~8 Y0~Y7 Digital output Y Bus
- 9 PWDN Power down mode
- 10 RST Reset
- 11 SDA I2C Serial data
- 12 FODD Odd Field flag
- 13 SCL I2C Serial clock input
- 14 HREF Horizontal window reference output
- 15 AGND Analog Ground
- 16 VSYN Vertical Sync output
- 17 AGND Analog Ground
- 18 PCLK Pixel clock output
- 19 EXCLK External Clock input (remove crystal)
- 20 VCC Power Supply 5VDC

- 21 AGND Analog Ground
- 22 VCC Power Supply 5VDC
- 23-30 UV0-UV7 Digital output UV bus
- 31 GND Common ground
- 32 VTO Video Analog Output (75Ω monochrome)

2.1 C3088 Camera Connections

The camera operation voltage is as the rest of the circuits 5V, and pins 20 and 22 that are VCC are connected to it. Pin 31 is GND and pins 21 and 15 are AGND (analog ground) and all of them are connected to the common ground. The pins PWDN and RST are connected to ground, so all the resets of the camera are performed by software. The bus Y0-Y7 of the camera is connected to the port A (PA0-PA7). PCLK is connected to PD2, HREF to PD3 and VSYNC to PD4. The bus UV is not connected because it is not used. SDA and SCL (I2C bus) from the camera are connected to PC7 and PC6 in the microcontroller, because these pins are connected to the TWI hardware that is used to implement the I2C protocol.

2.2 Getting Image from the Camera

The Image from the camera is read taking in account the diagram of Fig 2.2. Because the initial frequency of PCLK is 17.73 MHz and the AVR is not fast enough to read each pixel at this frequency two solutions could be taken:

- Use additional hardware to read and store the image.
- Decrease the frequency of PCLK by writing in the register 0x11.

This last solution was the one taken. The frequency taken to read the image depends on the way we read the image. If we read the image by horizontal lines we need to put the lowest frequency allowed that it is: 69,25KHz. This let us to read one line at the same time that is stored in the memory of the AVR. The other mode we read a vertical line of the image in each frame. In this case a higher frequency of 260 KHz is used, but we need to read as many frames as vertical lines have an image to get a complete one. In the case of the horizontal lines reading the resulting image is too bright, and that is the reason why the vertical reading is used, even if we need to read as many frames as vertical lines. The horizontal reading is used to read one horizontal line and make a little image process of it. The selection of this frequency was made experimentally trying to use the highest as possible frequency. When we want to send an image to the computer the headers and the palette are sent to the computer and then we proceed to read the image from the camera. We read from the first frame the first column and send pixel by pixel with the serial port to the computer, after that the second column, and so on until we are done with the whole image.

III. Results

Receiving the Image at Desktop

The best way to see the results obtained is to show how the system works. Once everything is connected and the terminal program in the computer is running we press the reset button and receive in the screen the next message:

```
....Control Camera Program....  
RAMANDEEP KAUR  
Type HELP and return for help
```

Also on the TV the image from the camera can be seen. If we press HELP we receive the next message:

```
ACK  
HELP MENU-Commands:  
RR arg1  
WR arg1 arg2  
READALL  
RESET  
MIRRORON  
MIRROROFF  
PHOTO  
TESTBMP
```

This menu shows the different commands implemented in the system. These commands are explained below:

- RR arg1 reads the register arg1 of the camera. arg1 can be expressed in hexadecimal or decimal format. It prints the result in the form: "Register = XX Value = XX".

- WR arg1 arg2 writes the value arg2 in the register arg1 of the camera. As in the case of RR, arg1 and arg2 can be expressed in hexadecimal or decimal format.
- READALL reads all the registers of the camera and prints them in the same way as RR does.
- RESET resets all the registers of the camera to the preset values.
- MIRRORON makes image to be seen as if it were seen through a mirror, this is the left part of the image is seen in the right part and vice versa. This change can be seen both in the TV and in the images sent to the computer.
- MIRROROFF makes the image to be seen without the mirror property.
- PHOTO sends a BMP file to the computer containing the image seen through the camera. To read the image with the microcontroller the frequency of PCLK is reduced, and for each frame only one column is read. This has as result that an image takes about one minute to be taken and sent at the same time to the computer. This makes that only static images can be taken. The resolution of the image is the resolution of the camera (352x244 pixels). We can see in Fig 3.1 photo captured by DigiCAM. To receive the image in the terminal of the computer we should put it in receiving file mode to store it in a file.
- TESTBMP sends a BMP generated by software image to the computer. This image is generated in the microcontroller with few control loops. It can be seen in Fig 3.2.



Figure 3.1 Photo taken with the camera

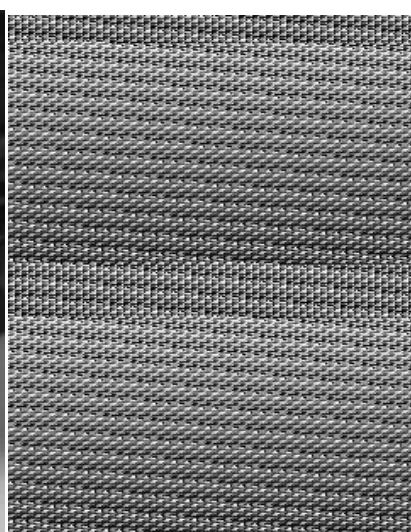


Figure 3.2 Image generated with the AVR

IV. Conclusion

All the proposed tasks of the project were done. The communication with the computer was done, also the I2C communication with the camera. Images can be read and sent to the computer in real time, with the only disadvantage that it takes a lot of time. The servo is also used to get panoramic images. As a future enhancement the system can be made wireless (mobile) and fully automatic by giving the entire control to the MCU. The limitation of the system developed is that it takes large time for processing the images captured by camera. So someone has to think for the solution to fasten the response time.

References

- [1]. Datasheet of the C3088 Camera <http://www.roboter-teile.de/datasheets/C3088.pdf>
- [2]. Datasheet of the SCCB bus http://www.ovt.com/pdfs/ds_note.pdf
- [3]. Digital Camera Interface <http://www.robozes.com/inaki/dproject>
- [4]. Web site of the I2C bus <http://www.semiconductors.philips.com/buses/i2c/>
- [5]. Web site if Microsoft with information about HyperTerminal program <http://www.microsoft.com>
- [6]. Web site of the Real Term terminal <http://realterm.sourceforge.net/>
- [7]. Datasheet of the PCF8584 <http://www.semiconductors.philips.com/pip/PCF8584P.html>
- [8]. Web page with information about the BMP file <http://www.progzon.free.fr/graphisme/formats/bmp/bmp.html>
- [9]. Ramandeep Kaur and Balwinder Singh, "Design and development of sapling Monitoring System", International Journal of Computer Science, Engineering and Application(IJCSEA) Vol. 3, No. 5, October2013.
- [10]. Tan Kejun; Luan Xiuzhen; Meng Xian Yao, "Design of Pan-Camera Decoder in Digital Video Monitoring System Based on Embedded Microcontroller," Electronic Measurement and Instruments, 2007 ICEMI '07. 8th International Conference on, vol., no., pp.3-183-22, Aug. 16 2007-July 18 2007
- [11]. Hassan, H.; Bakar, R.A.; Mokhtar, A.T.F., "Face recognition based on auto-switching magnetic door lock system using microcontroller," System Engineering and Technology (ICSET), 2012 International Conference on, vol., no., pp.1,6, 11-12 Sept. 2012.