# Hardware Implementaion of Image Acquisition system using FPGA & ARM

Ms. S. R. Lad
Department of Technology
Shivaji University
Kolhapur, India
*ssonalad@gmail.com*

Mr. P.C. Bhaskar
Department of Technology
Shivaji University
Kolhapur, India
*pxbhaskar@yahoo.co.in*

***Abstract*—** In this paper, image data acquisition system has been introduced. In this task, a system of high-speed image data acquisition based on ARM and FPGA is designed according to the needs of actual system in image data transmission, which can be used in data monitoring and surveillance systems. The choice of ARM is a 32-bit embedded RISC microprocessor architecture, which has a rich instruction set and programming flexibility. FPGA has a great advantage in the speed and parallel computing, suitable for real-time requirements of image processing. The interface between camera module and FPGA as well as interface between FPGA and ARM is done using UART. Image from ARM is transmitted to PC using Ethernet.

***Keywords-*** *Camera module, FPGA, ARM, Ethernet, UART*

*****

## I. INTRODUCTION

Image acquisition in image processing can be broadly defined as the action of retrieving an image from some source, usually a hardware-based source, so it can be passed through whatever processes need to occur afterward. Performing image acquisition in image processing is always the first step in the workflow sequence because, without an image, no processing is possible. The image that is acquired is completely unprocessed and is the result of whatever hardware was used to generate it, which can be very important in some fields to have a consistent baseline from which to work. One of the ultimate goals of this process is to have a source of input that operates within such controlled and measured guidelines that the same image can, if necessary, be nearly perfectly reproduced under the same conditions so anomalous factors are easier to locate and eliminate. In this paper a new image acquisition system based on Field programmable gate array (FPGA) and Advanced RISC Machines (ARM) technologies has been developed in order to realise the continuous data acquisition and real-time data transmission.

## I. HARDWARE DESIGN

### A. System Design

Image acquisition system can be divided into three main blocks:

1. Data acquisition: Essentially comprising the image sensor
2. Processor: Processes information to the specific application.
3. Communication interface: the interface through which various components of the system send or receive the data.

The image acquisition system requires a sensor which is basically an analog to digital converter to capture an image. If we want to do some processing on captured image we need a processor. Here we have selected FPGA which has an advantage of speed and parallel computing, suitable for real time requirement of image processing. To transmit the collected data from FPGA to PC we have selected ARM.

In our system, the camera module captures the image after sending commands from the PC. After capturing an image the image is sent to PC through FPGA and ARM. The system uses UART for the communication between camera module & FPGA as well as between FPGA & ARM. The communication between ARM & PC is done through Ethernet.

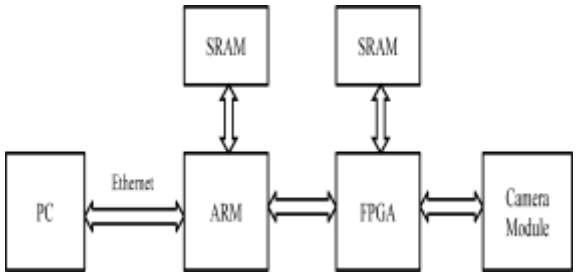


Fig.1 Block diagram of Image Acquisition System

### B. Image Acquisition

An image sensor or imaging sensor is a sensor that detects and conveys the information that constitutes an image. It does so by converting the variable attenuation of light waves (as they pass through or reflect off objects) into signals, small bursts of current that convey the information. The waves can be light or other electromagnetic radiation. Image sensors are used in electronic imaging of both analog and digital type, which include digital cameras, camera modules, medical imaging equipment, night vision equipment such as thermal imaging devices, radar, sonar. As technology changes, digital imaging tends to replace analog imaging.
The image sensor consists of picture elements, also called pixels, which register the amount of light that falls on them. They convert the received amount of light into a

corresponding number of electrons. The stronger the light, the more electrons are generated. The electrons are converted into voltage and then transformed into numbers by means of an A/D-converter. The signal constituted by the numbers is processed by electronic circuits inside the camera. There are 2 types of image sensors commonly used to digitally acquire an image, CCD and CMOS. While both have similar image quality, their core functionality and other features greatly differ.

For capturing an image TTL serial camera module has been used. The module has a few features built in, such as the ability to change the brightness/saturation/hue of images, auto-contrast and auto-brightness adjustment, and motion detection.

The serial port commands request the module to freeze the video and then download a JPEG color image. The maximum image size it can take is 640x480 pixels. And it is sensitive to infrared light, which alters the color rendition somewhat.

Specifications of camera module are:
Module size: 32mm x 32mm
Image sensor: CMOS 1/4 inch
CMOS Pixels: 0.3M
Output format: Standard JPEG/M-JPEG
Baud rate: Default 38400
Operating voltage: DC +5V
Communication: 3.3V TTL (Three wires TX, RX, GND)
User commands are sent using a simple serial protocol that can instruct the camera to capture images & send the jpeg image to host. Image file is a bunch of bytes. These bytes are sent one by one through the serial port. Then save these bytes received using UART in FPGA SRAM. All these bytes together form the jpeg file.

UART (Universal Asynchronous Receiver/Transmitter):

The Universal Asynchronous Receiver/Transmitter (UART) performs serial-to-parallel conversions on data received from a peripheral device and parallel-to-serial conversion on data received from the CPU. UART format consists of a start bit (0), data, parity bit and a stop bit(1). The start bit marks the beginning of a new word. When detected, the receiver synchronizes with the new data stream. Next follows the data bits (7 or 8 bit). LSB is sent first. The parity bit is added to make the number of 1's even (even parity) or odd (odd parity).This bit can be used by the receiver to check for transmission errors. The stop bit marks the end of transmission. The receiver checks to make sure it is '1'. Stop bit separates one word from the start bit of the next word.

**LPC1768-Xplorer**

LPC1768-Xplorer is a breakout board for the NXP LPC1768 ARM Cortex-M3 microcontroller. The LPC1768 operates at up to 100 MHz. Here LPC 1768 is configured as embedded erver. Its features are:

- 512KB of internal Flash
- 64KB RAM
- Ethernet MAC
- USB Device/Host/OTG interface
- 8-channel general-purpose DMA controller
- Four UARTs
- SPI interface
- Three I2C-bus interfaces
- 2-input plus 2-output I2S bus interface
- 8-channel 12-bit ADC
- 10-bit DAC
- Ultra-low-power Real-Time Clock (RTC) with separate battery supply
- Up to 70 general-purpose I/O pins.

Fig.2 LPC1768

The Ethernet block of LPC 1768 contains a full featured 10 Mbps or 100 Mbps Ethernet MAC (**Media Access Controller**) designed to provide optimized performance through the use of DMA hardware acceleration. Features include a generous suite of control registers, half or full duplex operation, flow control, control frames, hardware acceleration for transmit retry, receive packet filtering and wake-up on LAN activity. Automatic frame transmission and reception with Scatter-Gather DMA off-loads many operations from the CPU.

LPC1768 has 4-UARTs numbering 0-3, similarly the pins are also named as RXD0-RXD3 and TXD0-TXD3.As the LPC1768 pins are multiplexed for multiple functionalities, first they have to be configured as UART pins.

**Nexys 3 FPGA Board**

The Nexys 3 is a complete, ready-to-use digital circuit development platform based on the Xilinx Spartan-6 LX16 FPGA. The Spartan-6 is optimized for high performance logic, and offers more than 50% higher capacity, higher performance, and more resources as compared to the Nexys 2's Spartan-3 500E FPGA. Its features are:

- Xilinx Spartan-6 LX16 FPGA in a 324-pin BGA package
- 16Mbyte Cellular RAM (x16)
- 16Mbytes SPI (quad mode) PCM non-volatile memory
- 16Mbytes parallel PCM non-volatile memory
- 10/100 Ethernet PHY
- On-board USB2 port for programming & data xfer
- USB-UART and USB-HID port (for mouse/keyboard)
- 8-bit VGA port
- 100MHz CMOS oscillator
- 72 I/Os routed to expansion connectors
- GPIO includes 8 LEDs, 5 buttons,8 slide switches and 4-digit seven-segment display

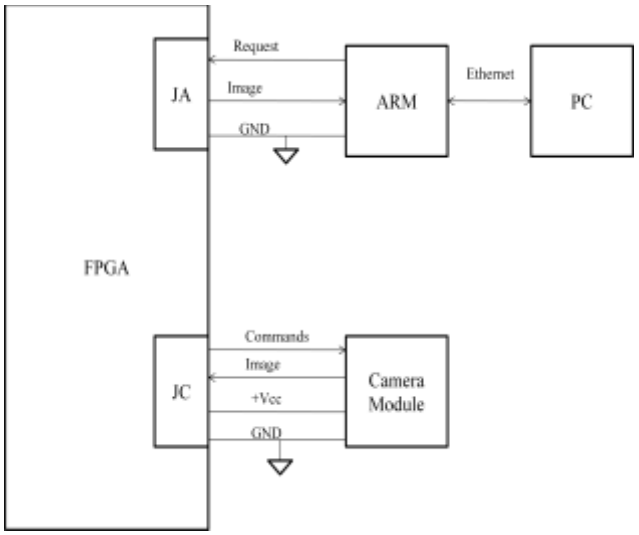


Fig.3 Communication between system devices

There are total 6 pins on camera module out of which 4 pins have been used for capturing image. Four pins are transmit, receive, +VCC and ground. Various commands used for capturing an image are: Reset command, Photo taking command, Reading the length of the photo taken command and Reading the data of photo taken command, stop photo taking command. These commands are stored in the form of look up table in command buffer unit. Command sequencer is used for selection of command. Data count unit stores the size of the data. To properly handle the UART transfer, we implemented a UART stack. The stack includes a receiver & FSM. The receiver interface consists of RX and TX lines for serial receive and transfer, respectively. The RX line is most important as this has the actual data. The receiver samples this line at the UART sampling clock frequency of 14 MHz divided by 24. It looks for an 8-bit UART transfer. The FSM receives bytes from the receiver one at a time. The protocol looks for a START byte, and then data followed by stop bit and buffers it into a shift register. Here, en_16_x_baud signal is used as clock enable within UART macros. This signal should be applied to the macro at a rate 16 times faster than the desired bit rate. In our system on board clock is 100MHz and required baud rate is 38400bps. Therefore,

en_16_x_baud= 38400*16=614400 Hz.

UART will automatically transmit any bits that are present in the buffer at the baud rate 'en_16_x_baud' (and the clock frequency) until the FIFO buffer is empty. It will take around 0.32seconds to transfer the captured image.

Image is received automatically and then stored in the receiver FIFO buffer. The FIFO is fully synchronous with the clock (clk). When active High (1), the 'buffer_data_present' flag is used to signify to the application that there is at least one bit ready to be read which will be present on the 'data_out' port. The application should capture this data and then provide an active High (1) level for one rising edge of the clock to the 'buffer_read' control input. The receiver FIFO will then drive the 'buffer_data_present' flag Low (0) to indicate that the FIFO is now empty.

Image stored in the SRAM of FPGA is transferred to ARM over UART when it requests for the data.

Here LPC1768 is implemented as a dynamic http server. An http server is a piece of software that understands URLs & HTTP. Whenever a browser needs a file hosted on a web server, the browser requests the file via http. When the request reaches the correct web server the http server sends the requested data back. Embedded C language has been used for the software implementation of the embedded web server.

Steps for Configuring UART0 of LPC1768:

Below are the steps for configuring the UART0.

1. Configure the GPIO pin for UART0 function using PINSEL register. It will configure the P0.2 pin as UART0 Tx and P0.3 pin as UART0 Rx to enable the serial communication over these pins.
2. Configure the FCR for enabling the FIFO and Reset both the Rx/Tx FIFO.
3. Configure LCR for 8-data bits, 1 Stop bit, Disable Parity and Enable DLAB.
4. Get the PCLK from PCLKSELx register 7-6 bits. We have PCLK=16250000 Hz
5. Calculate the DLM, DLL values for required baudrate from PCLK. Baud rate= 16250000/16=1Mbps
6. Update the DLM, DLL with the calculated values.
7. Finally clear DLAB to disable the access to DLM, DLL.

After this the UART will be ready to Transmit/Receive Data at the specified baudrate.

The image received by ARM is transferred to PC over Ethernet. Here ARM LPC1768 is implemented as embedded web server. The TCP/IP layered model of IEEE 802.3 standard is implemented in embedded C language.

An http request is sent from PC to ARM. If the IP address is correct and matches to that of the server a TCP/IP connection is established. Then ARM sends data request signal to FPGA in order to capture an image. After image has been captured the same image is stored in SRAM of FPGA. Then it is transferred to ARM. The image data is temporarily stored in the RAM of ARM at the same time it is transferred to PC.

## RESULTS

Fig. 4 Captured Image

## CONCLUSION

This system gives more flexibility of ARM and the parallel processing of FPGA, to design the high-speed image data acquisition board. The commands are sent from PC to capture the image. The image is captured by camera module & sent to PC over Ethernet from ARM. Batch script is used for running multiple commands in sequence. It takes approximately 5 seconds for the image to be available on PC after sending the request. It is applicable to a variety of fields such as surveillance, industrial control, medical, instrument etc

## REFERENCES

[1] Wan XiaoFeng, Zeng Pu. "A design of data acquisition of high speed CCD camera", 2009
[2] Smita Salgaonkar, Mrs. J.D. Bhosale, Mrs. Padmaja Bangade, "Embedded web server based on ARM cortex for DAC system", International Journal of Scientific & Engineering Research, Volume 5, Issue 7, July-2014 ISSN 2229-5518
[3] www.wisegeek.com
[4] Suma G S, "FPGA based high speed data acquisition system with ethernet interface", International Journal of Advances in Science Engineering and Technology, ISSN: 2321-9009
[5] Hui Xiaowei, Shen Qinglei, Miao Changyun "Acquisition board design of high speed image data based on ARM & FPGA", , 201O International Conference On Computer Design And Appliations (ICCDA 2010)
[6] User Manual: LPC1768-Xplorer
[7] OV0706 TTL camera datasheet
[8] www.xilinx.com
[9]Wang Fei, Wu Zhijie, Chen Hong, Xi Yi2, " High-Speed Data Acquisition System Based on FPGA/SoPC," The Tenth International Conference on Electronic Measurement & Instruments, 2011, pp.21-27 [8].
[10]Anju P. Raju, "High Speed Data Acquisition System with Ethernet Interface", IOSR Journal of Electronics and Communication Engineering (IOSRJECE) ISSN : 2278-2834 Volume 2, Issue 5 (Sep-Oct 2012), PP 12-17
[11] Niturkar Priyanka, Prof. V.D.Shinde, "Design and development of ARM9 based embedded web server", Int. Journal of Engineering Research and Applications, ISSN: 2248-9622, Vol. 5, Issue 8, (Part - 5) August 2015, pp.50-53