International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:268
Vol.2, No.11, November 2013 DOI:10.15693/ijaist/2013.v2i11.31-35

Sobel Edge Detection Using FPGA

AbhijeetKhandale
M.Tech Student
Dept. of ECE
BMS College of Engineering, Bangalore
INDIA

H R Bhagyalakshmi
Associate professor
Dept. of ECE
BMS College of Engineering, Bangalore
INDIA

***Abstract*—A comprehensive method of implementation of the Sobel edge detection algorithm for low power consumption and fast performanceusing FPGA has been presented in this paper. The image pixel values are sent to the FPFA from PC through serial communication. MATLAB GUI is used to interface PC and FPGA. The image pixels are converted to binary format.**

***IndexTerms*—FPGA,MATLAB, Image processing, RS232, VHDL, Verilog.**

I. INTRODUCTION

Field Programmable Gate Array (FPGA) is the part of current reconfigurable computing technology, which is an ideal alternative for image and video processing.Field Programmable Gate Array (FPGA) technology became a viable target for implementation of real time algorithms suited to video processing application. FPGAs generally consist of a system with logic blocks such as look up tables, gates, flip-flops and some memory blocks all placed in the vast array of interconnects. The FPGA can be reconfigured to a particular logic circuit using hardware description language like VHDL or Verilog. The FPGA architecture allows large variety of logic designs for real time application.

In digital image processing,the edge is referred as drastic change of the pixel values or it is also the boundary between two different colors or within same color with different shades. The edge detection method is to determine the variance among the pixels by applying matrix operator with different sizes.There are various methods to implement edge detection such as first order (gradient method) and second order derivatives. The gradient method includesRoberts's operator, Prewitt operator and Sobel operator [1].

MATLAB GUI is used to communicate with the FPGA board.The Image is converted into binary format and then using RS232 interface the pixel data has been sent to the board.

This paper gives a better idea to implement the edge detection on FPGA board.As the data is in binary format the hardware required is less and processing is fast.Because of binary data one don't need to implement square root algorithm in Verilog or VHDL language.In the section II, we describe the Sobel operator; RS232 communication protocol and MATLAB GUI implementation along with flow-chart to implement the HDL code. The MATLAB GUI interface, simulation results and FPGA board statistics are presented in Section III. Conclusion is drawn in section IV.

II. DESIGN REQUIREMENT AND IMPLEMENTATION

A. Sobel Edge Detection

Sobel edge detection has two masks, one mask detects the horizontal edges and other mask detects the vertical edges. The mask that finds horizontal edge is equivalent to vertical gradient and the mask that finds vertical edge is equivalent to horizontal gradient [2].

A(1,1)	A(1,2)	A(1,3)	…	A(1,n)
A(2,1)	A(2,2)	A(2,3)		
A(3,1)	A(3,2)	A(3,3)		
⋮	⋮	⋮		⋮
A(m,1)			…	A(m,n)

Fig.1. Arrangement of pixels for an image

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Fig. 2.Sobel Operator

By applying these two masks on an image the gradient along the horizontal and vertical direction can be computed at different location in the image. The gradient of an image $f(x,y)$at location (x,y) is defined as a vector.

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (1)$$

Computation of the magnitude of the gradient is calculated as follows.

$$\nabla f = mag(\nabla f) = [G_x^2 + G_y^2]^{1/2} \quad (2)$$

B. RS232

The RS-232 serial communication protocol is a standard protocol used in asynchronous serial communication [3].

In asynchronous serial communication the data is transmitted without any clock signal to the receiver. Instead, special bits like start bit stop bit and parity bits are sent along with the data bits to synchronize transmitter and receiver. When data has to send using asynchronous transmission a start bit is added at the beginning of the data and then data bits along with parity bit and stop bit is added. Here parity bit is optional. In ideal condition both Tx and Rx lines are held high. The length of data bits that can be sent are 5, 6, 7or 8 bits. The start bit is '0' and the end bit might be 1, 1.5 or 2 bits in length with '1' value.

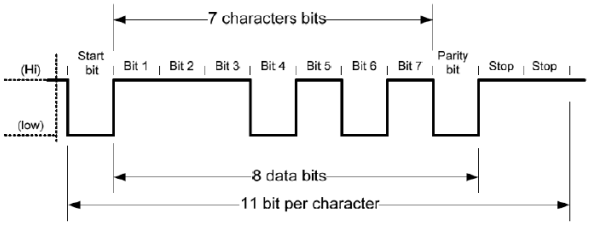

Fig. 3. RS232 asynchronous communication data waveform.

C. Baud Rate Calculation

Baud rate is a measurement of transmission speed in asynchronous communication; it represents the number of bits that are actually being sent over the serial link

Spartan 3E starter Kit Operating Frequency= 50 MHz

Baud Rate = 115200

Count1 = 50 MHz / 115200 ~= 434

For receiver the sampling is required so the Count is get modified for receiver

Count2 = 50 MHz / (115200 * 16) ~= 27

The flow chart to implement and generate the baud rate is given below.Here two counter are initiated to count the values shown in count1 and count2. After reaching each value the counter will be reset and start counting again. For each time counters reaching the terminal value , the baud clcok and sampled baud clock are set '1'.

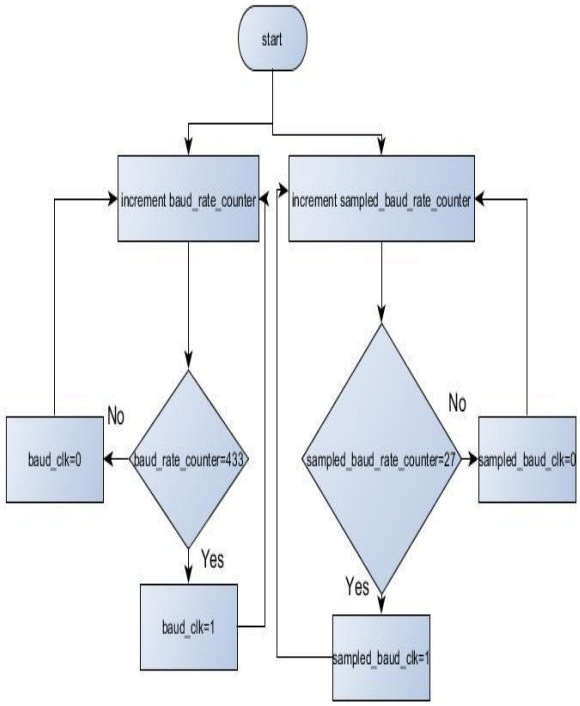

Fig. 4. Flow chart of Baud rate signal generation.

D. Implement RS232 Prototype using HDL

The receiver block of the RS232 serial communication will run on the sampled baud clock while transmitter block will run on the normal baud clock signal.The transmitter can also run on the sampled baud clock but as switching will be more in sampled baud clock to reduce the dynamic power consumption we will run transmitter block on normal baud clock signal.

The sampling can be done with 8, 16 or 32 samples.Here we are doing it with 16 samples.The sampling is required because of the uncertainty of the start bit arrival, as it is a asynchronous communication. The flow chart for receiver and transmitter is shown below.In the receiver the counter2 and counter 3 need to be synchronized. As we are taking 16 samples the data will be captured at 8^{th} sample , so as soon as counter2 reaches 8^{th} sample the data will be received and after 16^{th} sample counter3 will be incremented. Counter3 has a length equivalent to the data length.

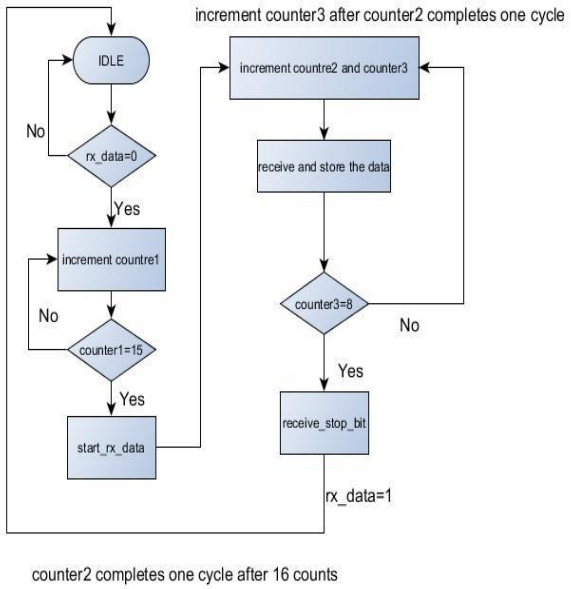

Fig.5. Flow chart for RS232 receiver.

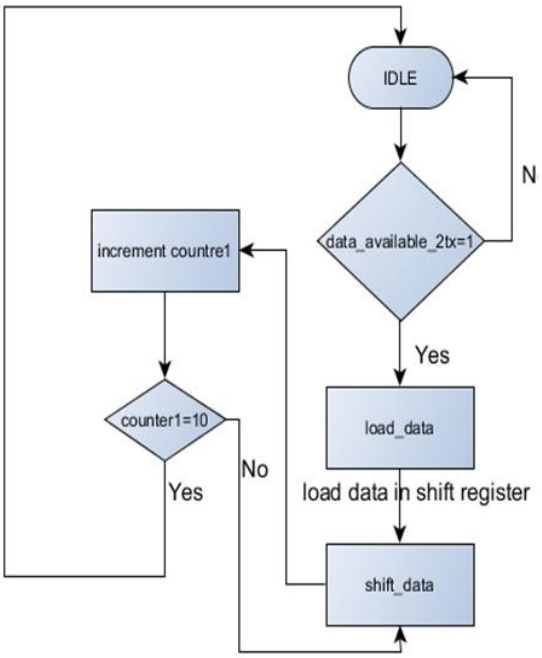

Fig. 6. Flow chart for RS232 transmitter

E. MATLAB GUI Implementation

MATLAB has a very good Graphics User Interface development environment tool to develop reliable and fast user interface. The binary image conversion is done easily with MATLAB in build functions. Also serial communication prototype is implemented using MATLAB to communicate with FPGA board via RS232.

The MATLAB graphics use interface window is as shown in below

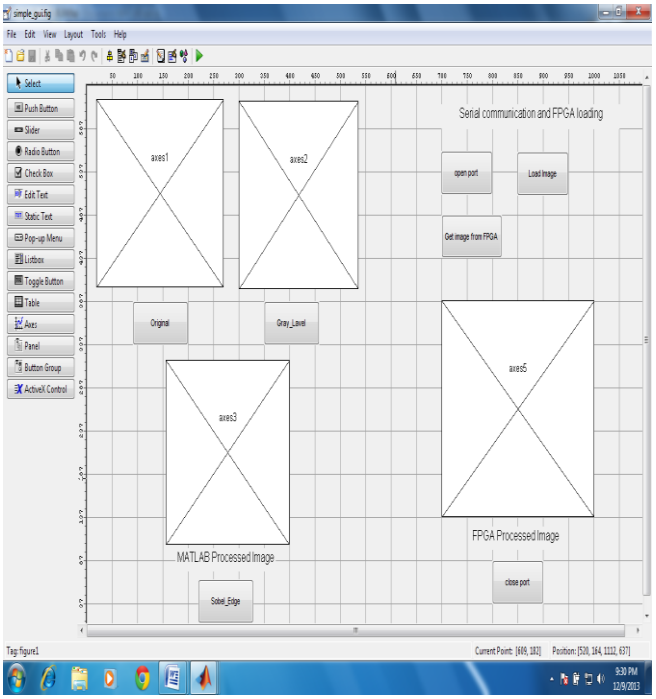

Fig.7. MATLAB GUI

F. Creating MATLAB Serial Port

Serial_port= **serial** (‘com3’) //port creation

Setting Parameter of the Port

1) **Set** (serial_port, ‘BaudRate’, 115200)
2) **Set(**serial_port, ‘InputBufferSize’, totalpixels**)**
3) **Set** (serial_port, ‘OutputBufferSize’, totalpixels)

Writing and reading to/from the port

Fopen(serial_port)	//opens the port
Fwrite(serial_port, [0,12,4,5])	//writing binary data
A= **fread**(serial_port, n)	//reading binary data
	//n indicates no. of data

Closing serial port
Delete (serial_port)
Clear serial_port

G. Sobel Edge Detection

Here we operate the horizontal and vertical operators on binary values and after computing the combined gradient we normalised to binary again so we don’t need to compute the square root of it [4]. The flow chart is as given below.

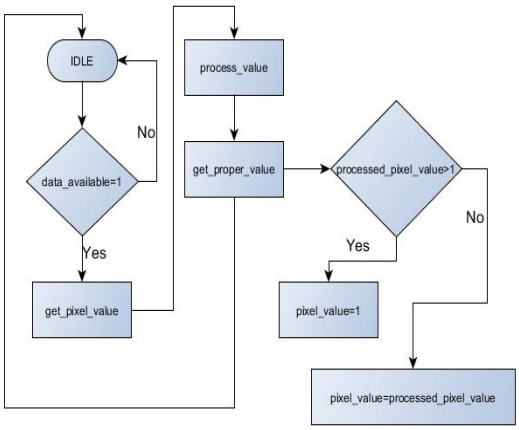

Fig.8. Simulation results for first part

III. Simulation and Implementation Result

Here we have used 256x256 pixel images, which have 65536 no. of pixels values. While synthesizing the HDL code, we have used internal FPGA RAM instead of on board memory, so the no. of LUTs used gone up to 90% from the initial 20% value.

The simulation has been divided into three parts. These parts are receiving data, processing and obtaining Sobel edge and finally transmitting the data.

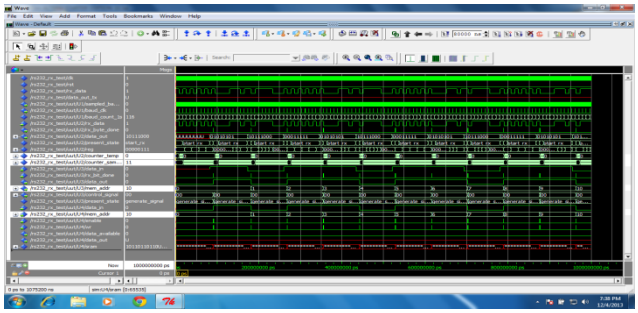

Fig.9. Simulation results for first part

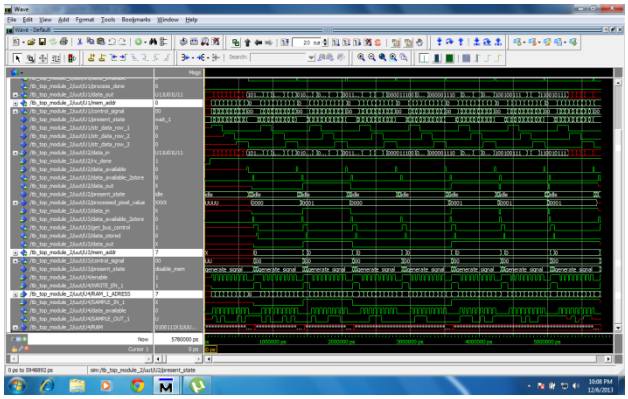

Fig.10. Simulation results for second part

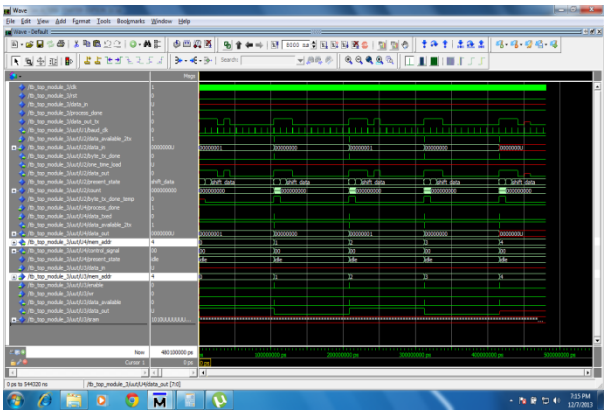

Fig.11. Simulation results for third part

A. Device utilization summary

TABLE I. Device:3s500efg320-4 utilization

Number of Slices	4177 out of 4656	89%
Number of Slice Flip-Flop	316 out of 9312	3%
Number of 4 input LUTs	8022 out of 9312	86%
Number used as logic	3926	
Number used as RAMs	4096	
Number of IOs	4	
Number of bonded IOs	4 out of 232	1%
Number of GCLKs	2 out of 24	8%

IV. Conclusion

Converting Image to binary values will increase the efficiency of the system. Image processing applications required large memories, due to this memory control logic become vital in the image processing application. RS232 serial communication is simple to implement but the transfer speed is very less as compared to other communication techniques. The hardware description languages are very useful to implement the algorithms in behavioral models.

The parallel architecture for the algorithms can be implemented using different methods to increase the efficiency and to decrease the computational time. The on

boards capture and conversion of image is useful to eliminate the PC and FPGA board interfacing.

.

ACKNOWLEDGMENT

This research is supported by the BMS College of Engineering, Bangalore. The authors wish to thank BMS college of Engineering for supporting this work by encouraging and supplying thenecessary tools.

REFERENCES

[1] Mr. Manoj K. Vairalkar, Prof. S.U. Nimbhorkar, "Edge detection of images using sobeloperaotr," International Journal of Emerging Technology and Advance Engineering, Vol.2, January 2012.

[2] Han Xiaoru, GaoYudong, "Design and Implementation of the Universal RS232-GPIB Interface," Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on.

[3] I. Yasri, N.H. Hamid, V.V. Yap, "Performance Analysis of FPGA Based Sobel Edge Detection Operator," IEEE2008 International Conference on Electronic Design, Dec. 2008.