IJECBS

Frequency: Bi-Annual

ISSN (Online): 2230 - 8849

International Journal of Enterprise Computing and Business Systems (Online)

IJECBS India

Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases

http://www.ijecbs.com

Vol. 2 Issue 2 July 2012

Matrix Manipulation Using High Computing Field Programmable Gate Arrays

[1]*Mr.Rounak R. Gupta*, [2]*Prof. Atul S. Joshi*

Department of Electronics and Telecommunication Engineering,
Sipna College of Engineering and Technology,
S.G.B.Amravati University
Amravati,444603, India

Department of Electronics and Telecommunication Engineering,
Sipna College of Engineering and Technology,
S.G.B.Amravati University
Amravati,444603, India

ABSTRACT

Matrix manipulation is very important in many types of applications. The suitability of reconfigurable hardware devices, in the form of field programmable gate arrays (FPGAs), is investigated as a low-cost solution for implementing two matrix manipulation operations. This paper present the design and implementation of matrix operations like addition, subtraction and multiplication using VHDL design approach, where the performances of programming language have been presented Solutions for processing different matrix operations.

Advanced RISC Machine (ARM) is the Implementation result of the RISC microprocessor architecture. RISC introduces simpler hardware processor architecture because fixed length Instruction format with the opcode in the same bit position requires less decoding, allowing any register to be used in any context simplifies the compiler design and Complex addressing is performed through sequences of arithmetic and/or load store operations.

Keywords: Coprocessor, FPGA, Matrix Manipulation, RISC, VHDL.

1. INTRODUCTION:

Shift registers are a type of sequential logic circuit, mainly for storage of digital data. They are a group of flip –flops connected in a chain so that the output from one flip -flop becomes the input of the next flip -flop. Most of the registers possess no characteristic internal sequence of states. All flip -flop is driven by a common clock, and all are set or reset simultaneously.

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

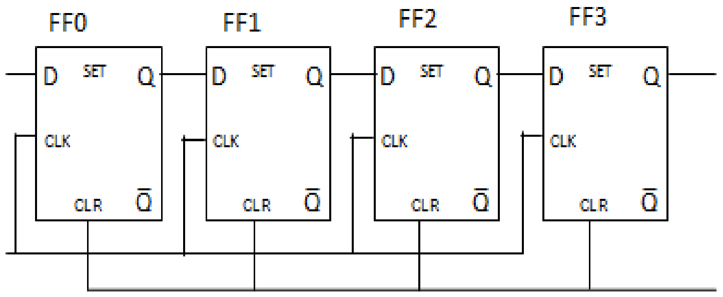


Fig.1 4- bit shift register

A basic four-bit shift register can be constructed using four D flip -flops, as shown in Figure The operation of the circuit is as follows.
1.The register is first cleared, forcing all four outputs to zero.
2. The input data is then applied sequentially to the D input of the first flip -flop on the left (FF0).
3.During each clock pulse, one bit is transmitted from left to right.
4. Assume a data word to be 1001.
5. The least significant bit of the data has to be shifted through the register from FF0 to FF3.

In order to get the data out of the register, they must be shifted out serially. This can be done destructively or non- destructively. For destructive readout, the original data is lost and at the end of the read cycle, all flip -flops are reset to zero.

FF0	FF1	FF2	FF3	
0	0	0	0	1001

The data is loaded to the register when the control line is HIGH (ie WRITE). The data can be shifted out of the register when the control line is LOW (ie READ).

CLEAR	FF0	FF1	FF2	FF3
1001	0	0	0	0

WRITE:

FF0	FF1	FF2	FF3	
1	0	0	1	0000

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

READ:

FF0	FF1	FF2	FF3	
1	0	0	1	1001

Now as shown in fig. 2 if we have desire for multiplication operation it should be perform by shift register in the multiple of 2.

Let take an simple example using 0001.initially the value in the shift register is 1. After the first left shift, value in the shift register must be 2. After second left shift, this value should be 4 and finally after third shift it will give 8. So by using shift register we can perform multiplication operation in the multiple of 2 only.

In this paper we designed a system that gives simple and faster matrix manipulation operations.

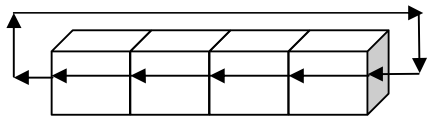


Fig.2 Rotate Left Operation

2. System Building Blocks:

The system can be best suitably planned around high performance programmable gate arrays. Fig. shows the proposed system build around Field Programmable Gate Arrays (FPGA).

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

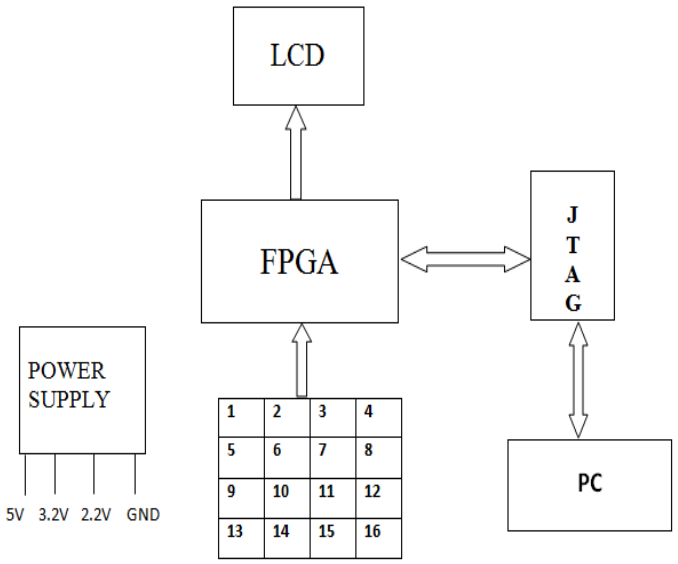


Fig.3 Basic Building Blocks

Out of all other intelligent devices, programmable gate arrays are excellent choice because of startling features like 16-bit LUT RAM, In-System Programming (ISP), Boundary scan and Read back ability. Out of available different versions and capacity FPGAs, XC2S200 is the best choice.

2.1 4x4 Hex Keypad:

It Hex keypad is the matrix arrangement of 4x4 press to on switches. With 4x4 arrangements it becomes 16 possible values that can be encoded and designer can encode the values as per their convenience. For entering the values for matrix manipulation it is required to values to be input from the 4x4 matrix.

2.2 Hex Keypad Scanner:

For the Matrix manipulation using high speed programmable gate arrays, the sixteen combinations are encoded into Hex Values i.e. values from ‘0’ to ‘9’ and characters from ‘A’ to ‘F’.

For encoding the values from the 4x4 hex keypad matrix, keypad scanning routine needs to be executed repeatedly, keypad can be scan by enabling only one row **(or column)** at a time and checking the individual column **(or row).** For each of the column in enabled row different values can be encoded. Fig. 4 shows the matrix like structure of the Hex-keypad. Same process can be repeated for remaining rows.

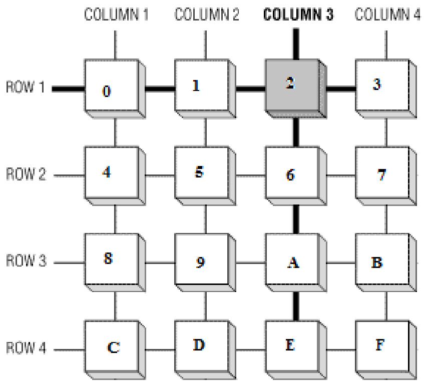


Fig. 4 Hex Keypad structure

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

3. Proposed Flow Diagram:

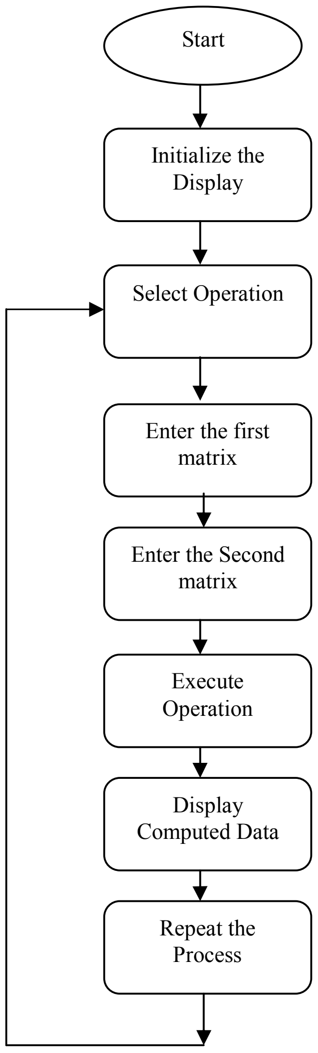


Fig.5 Flow Chart

The process of matrix manipulation can be initiated by initializing the LCD display unit to following mode:

a. 8-bit Interface Mode,
b. Double Character Line Mode with 5x7 Dot Matrix

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
IJECBS
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

c. Display Curser On and
d. Increment next character display address automatically.

For initializing the LCD unit to the above mode data pattern of 38H, 0CH and 06H can be input to the control register with Register Select pin as Low. The above pattern can be loaded into the RS Register by each trailing edge of the pulse on Enable (E) pin. After such initialization the LCD unit is ready for the data displaying.

The data once available inside the FPGA, it can be operated to get the desired operation by operating it using suitable algorithm for matrix manipulation.

Algorithm can be designed to operate the Hex data directly or data can be first converted to decimal values and then operate on the converted data. It is also possible to convert the incoming data into binary format or octal format and then operate on the new data.

After multiplication the result is displayed on the LCD. For any of the converted data manipulation algorithm data needs to be converted into ASCI value for displaying on the LCD unit. As the size of the matrix is large i.e. having atleast 4x4, which requires four rows to be displayed and the size of the LCD is of 2 rows. Hence to display total 4 rows, shift register can be designed internally and the top of the register bank can be used to display the complete 4 rows. This technique can be used to display N number of rows by using only 2 - Row LCD display.

The operation result to be display is depending upon the control register LSB 2 bits as given in the following table:

BIT1	BIT0	OPERATION
0	0	NO OPERATION
0	1	MULTIPLICATION
1	0	ADDITION
1	1	SUBTRACTION

First all the coefficients of the both matrices are to be updated and then control register data should be updated else we may get wrong result.

4. Observation:

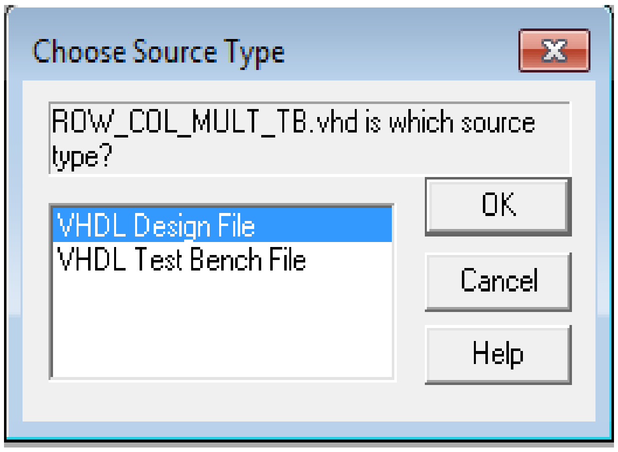
```
entity MATRIX_MULT is
GENERIC (
B: INTEGER :=8;                    --NUMBER OF BITS
W: INTEGER :=7;                    --NUMBER OF ADDRESS BITS
C: INTEGER :=16                              --BUS WIDTH
          );
     Port (
     clock : in  STD_LOGIC;
     reset : in  STD_LOGIC;
     mult_en : in  STD_LOGIC;
```

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

```
            a11 : in  signed ( B-1 downto 0);
end MATRIX_MULT;
        COMPONENT ROW_COL_MULT
            GENERIC (
            B: INTEGER;
                            --NUMBER OF BITS
            W: INTEGER;
                            --NUMBER OF ADDRESS BITS
            C: INTEGER
                            --BUS WIDTH
                    );
            PORT(
                            clock : IN  std_logic;
                            reset : IN  std_logic;
                          mult_en: IN  std_logic;
                    a_r1 : IN  signed ( B-1 downto 0);
                    a_r2 : IN  signed ( B-1 downto 0);
                    a_r3 : IN  signed ( B-1 downto 0);
                    a_r4 : IN  signed ( B-1 downto 0);
                    b_c1 : IN  signed ( B-1 downto 0);
                    b_c2 : IN  signed ( B-1 downto 0);
                    b_c3 : IN  signed ( B-1 downto 0);
                    b_c4 : IN  signed ( B-1 downto 0);
                    out1 : OUT signed ( C-1 downto 0)
                                    );
            END COMPONENT;
                          generic map ( B, W, C)
                                PORT MAP (
                               clock=> clock,
                               reset=> reset,
                            mult_en=> mult_en,
                              a_r1 => a11_sig,
                              a_r2 => a12_sig,
                              a_r3 => a13_sig,
                              a_r4 => a14_sig,
                              b_c1 => b11_sig,
                             b_c2    => b21_sig,
                             b_c3    => b31_sig,
                             b_c4    => b41_sig,
                               out1     => c11
                                    );

        element_c12: ROW_COL_MULT;
```

IJECBS
Frequency: Bi-Annual
ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

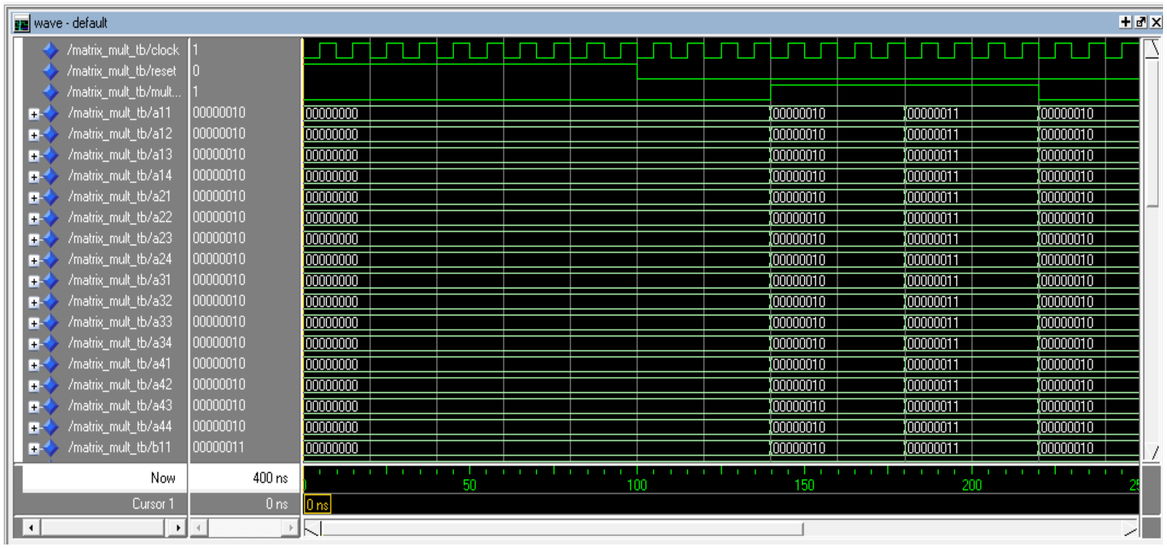


Choose Source Type

ROW_COL_MULT_TB.vhd is which source type?

VHDL Design File
VHDL Test Bench File

Waveform 1



```
/matrix_mult_tb/clock    1
/matrix_mult_tb/reset    0
/matrix_mult_tb/mult...  1
/matrix_mult_tb/a11      00000010
/matrix_mult_tb/a12      00000010
/matrix_mult_tb/a13      00000010
/matrix_mult_tb/a14      00000010
/matrix_mult_tb/a21      00000010
/matrix_mult_tb/a22      00000010
/matrix_mult_tb/a23      00000010
/matrix_mult_tb/a24      00000010
/matrix_mult_tb/a31      00000010
/matrix_mult_tb/a32      00000010
/matrix_mult_tb/a33      00000010
/matrix_mult_tb/a34      00000010
/matrix_mult_tb/a41      00000010
/matrix_mult_tb/a42      00000010
/matrix_mult_tb/a43      00000010
/matrix_mult_tb/a44      00000010
/matrix_mult_tb/b11      00000011
Now      400 ns
Cursor 1   0 ns
```

Waveform 2

5. Acknowledgment:

Thanks to department of Electronics and Telecommunication of SIPNA college of engineering and technology, Amravati.

6. Conclusion:

Considering the market segment, different manufacturers supports different processors or co-processors that can be used to compute the complex algorithms. As compared to that the present system is more users friendly and demonstrates each and every step of execution of the complex algorithm.

System is designed around high computing, high performance gate arrays that are extensively used for integrating parallel architecture and furnishes outstanding performances when operated at frequencies in the range of and above 100 MHz. This system can be used for lab purposes for digital image processing and analysis.

7. References:

[1] "Digiital Logic and Computer Design", by Morris Mano: Advanced Digital Design fundamentals and Issues.
[2] http://www.itu.dk/courses/ISOM/E2005/ARMv6Architecture.pdf
[3] http://www.arm.com/products/processors/index.php
[4] http://www.graficaobscura.com/matrix/index.html
[5] www.xilinx.com
[6] http://www.sciencedaily.com/releases/2008/02/080208162238.htm
[7] http://www.sciencedaily.com/releases/2008/02/080208162238.htm

[8] http://books.google.com/books?id=C9UxoH5ehwYC&printsec=frontcover&source=gbsv2_summary_r&cad=0#v=onepage&q&f=false
[9] http://jnci.oxfordjournals.org/content/93/20/1563
[10] S. Vassiliadis, S. D. Cotofana, and P. Stathis. Vector isa extension for sprase matrix multiplication. In Proceedings of EuroPar'99 Parallel Processing, pages 708–715, September 1999.
[11] S. Vassiliadis, S. D. Cotofana, and P. Stathis. Bbcs based sparse matrix-vector multiplication: initial evaluation. In Proc. 16th IMACS World Congress on Scientific Computation, Applied Mathematics and Simulation, pages 1–6, August 2000.
[12] S. Vassiliadis, S. D. Cotofana, and P. Stathis. Block based compression storage expected performance. In Proc. 14th Int. Conf. on High Performance Computing Systems and Applications (HPCS 2000), June 2000.

8. Author Information:

Mr. Rounak R. Gupta is currently working as a lecturer in Electronics and Telecommunication Engineering, Sipna College of Engineering, Amravati (India), since 2011. He is also pursuing his M.E. in Digital Electronics from the same institute. His areas of interest are Digital Image Processing, VLSI Design and Digital Signal Processing.

Prof. Atul S. Joshi is currently working as Professor in Electronics and Telecommunication Engineering Department, Sipna College of Engineering, Amravati (India) since inception. He is also perusing his Ph.D. in the faculty of Electronics Engineering From SGB Amravati

ISSN (Online): 2230 - 8849
International Journal of Enterprise Computing and Business Systems (Online)
IJECBS India
IJECBS
Frequency: Bi-Annual
Specialized, Refereed and Indexed Journal in International Scientific and Corporate Databases
http://www.ijecbs.com
Vol. 2 Issue 2 July 2012

University, Amravati (India). His areas of interest are Communication networks, Electronics Circuits and Communication Engineering.