ABSTRACT
There are a number of physical features that have driven the ARM processor design. First, Portable embedded system require some form of battery power. The ARM processor has been specifically designed to be small to reduced power consumption and extended battery operation-essential for applications such as mobile phones and personal digital assistants (PDAs). It is used for 32-bit embedded processor cores. Many of the top semiconductor companies around the world produce products based around the ARM processor. The main idea of this paper is to make such a 32 bit operated RISC machine using FPGA. This will help designer to increase the core density and to make the less power consumption device for the 32 bit ARM coprocessor. In this paper 32*32 matrix coprocessor is developed which is used for different applications like image processing, satellite communication etc.

KEYWORDS ARM, FPGA, RISC, DSP

I. INTRODUCTION
ARM has incorporated hardware debug technology within the processors so that software Engineers can view what is happening while the processor is executing code .with greater Visibility, software engineers can resolve issues faster, which has a direct effect on the time to market and reduces overall development costs. The ARM core is not a pure RISC architecture because of the constraints of its primary Application-the embedded system. In some sense, the strength of the ARM core is that it does not take the RISC concept too far. In today’s systems the key is not raw processor speed but total effective system performance and power consumption. The ARM instruction set differs from the pure RISC definition in several ways that make the ARM instruction set suitable for embedded applications: Variable cycle execution for certain instruction-not every ARM instruction executes in a single cycle. For example, load-store-multiple instruction varies in the number of execution cycles depending upon the number of registers being transferred. The Transfer can occur on sequential memory addresses, which increases performance since sequential memory accesses are often faster than random accesses. Code density is also improved since multiple register transfers are common operations at the start and end of functions. Inline barrel shifter leading to more complex instructions- the inline barrel shifter is a hardware component that preprocesses one of the input register before it is used by an instruction. This expands the capability of many instructions to improve core performance and code density. Thumb 16-bit instruction set-ARM enhanced the processor core by adding a second 16-bit instruction set called thumb that permits the ARM core to execute either 16- or 32-bit instructions. The 16-bit instruction improve code density by about 30% over 32-bit fixed-length instructions. Conditional execution is only executed when a specific condition has been satisfied. This feature improves performance and code density by reducing branch Instructions. Enhanced instructions- the enhanced digital signal processor (DSP) instructions were added to the standard ARM instruction set to support fast 16*16-bit multiplier operations and saturation. These instructions allow a faster-performing ARM processor in some cases to replace the traditional combinations of a processor plus a DSP. The ARM processor controls the embedded device. Different versions of the ARM processor are available a core (the execution engine that processes instructions and manipulates data) plus the surrounding components that interface it with a bus. These components can include memory management and caches. Controllers coordinates important functional block of the system. Two commonly found controllers are interrupt and memory controllers.

II. PROPOSED SYSTEM ARCHITECTURE AND DESIGN
VHDL does not constrain the user to one style of description. VHDL allows designs to be described using any methodology - top down, bottom up or middle out! VHDL can be used to describe hardware at the gate level or in a more abstract way. Successful high level design requires a language, a tool set and a suitable methodology. VHDL is the language; you choose the tools, and the methodology. Below shows the hardware architecture and working of the project.

Fig 1. Hardware Block diagram of the system
A. Functional architecture of XC 2S100 PQ 208:
Field Programmable Gate Arrays is a flexible, programmable architecture that consist of Configurable Logic Blocks (CLBs), Programmable IO Blocks, Delayed Lock Loops (DLLs), and In-system Programmable Capability etc. CLBs are important blocks in the programmable gate arrays. It is the unit that is configured by the application program created by the designer. Generally the CLBs are configured using some Hardware Description Languages (HDLs) like Very High Speed Integrated Circuit Hardware Description Language (VHDL) and Verilog. Basic element of the CLB is the Look-Up Table (LUT). The Spartan®-II Field-Programmable Gate Array family gives users high performance, abundant logic resources, and a rich feature set, all at an exceptionally low price. The six-member family offers densities ranging from 15,000 to 200,000 system gates. System performance is supported up to 200 MHz. Features include block RAM (to 56K bits), distributed RAM (to 75,264 bits), 16 selectable I/O standards, and four DLLs. Fast, predictable interconnect means that successive design iterations continue to meet timing requirements. The Spartan-II family is a superior alternative to mask-programmed ASICs. The FPGA avoids the initial cost, lengthy development cycles, and inherent risk of conventional ASICs. Also, FPGA programmability permits design upgrades in the field with no hardware replacement necessary (impossible with ASICs).

B. Hex Keypad and Hex Keypad Scanner:
Hex Keypad is the means using which input variables can be applied to the matrix. It is simply arrangement of Micro switches in a matrix form after decoding of which hexadecimal values are obtained. Hex keypad scanner is the chip which decodes the key pressed and generates the distinct binary value output for each key press. Along with the detection of key pressed and generating the distinct code for individual code, hex keypad is also serves to select the particular operation to be performed on the input data stream.

C. Display Screen:
Display screen is basically 16x2 character LCD display, which will display the input data being applied, mode of operation and finally the manipulated data. Display screen provides the means through which user can confirm the activities of the design with one which is expected. Instead of character LCD Display, Graphic LCD can also be applied which will be the better solution to display the matrix elements, but using graphic LCD will unnecessary increase the system cost, along with the hardware and software complexity.

D. Clock Unit:
Clock generating unit is the frequency source that provides the stable frequency source to work and synchronize the different modules of the system sub-blocks. Crystal as a frequency source followed by the clock driver circuit can be used for successful generation and distribution of the clock. Clock distribution is an important factor in the VLSI based application development. If this distribution of clock is not proper then it causes drawbacks such clock skew, Metastability and total system performance hampers greatly. To avoid such drawbacks clock driver unit is employed which takes care for better performance.

E. Display Section:
16x2 character Liquid Crystal Display is used to display the number of visitors visited to the arena under surveillance. It is 16x2 meaning that 16 characters at a time and in two columns. With this it is possible to display total 32 characters at a time.

F. The overall hardware prerequisites are as follow:
- Full wave Bridge rectifier
- Clock circuit for 4 MHz
- Supply 9 volt, 1 amp
- Filtering by 1 capacitor 470 microfarad, 50 volts
- 7805 regulator IC
- LM 317 for generating 2.2 v, LM 1117 for generating 3.3 v.
- 16*2 LCD, 4*4 hex keypad
- XC2s100 Pq208
- Pin header flash(XCF01S) and JTAG cable

G. Software requirement:
- Eagle software based layout for the hardware and the pin descriptions
- Xilinx 10.1 ISE simulator

III. OBSERVATION AND SYNTHESIS REPORT
The 32*32 bit matrix coprocessor is performing the different operations of matrix. This Spartan XC2s100pq208-5 chip has to deal with 1024 bits for matrix operations like matrix transpose, matrix addition, matrix subtraction and matrix multiplication. The RTL view and different result are shown after the simulation and synthesis as below.

<table>
<thead>
<tr>
<th>Selected Device</th>
<th>2S100pq208-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slices</td>
<td>2813 out of 1200 234%</td>
</tr>
<tr>
<td>Number of Slice Flip Flops</td>
<td>928 out of 2400 38%</td>
</tr>
<tr>
<td>Number of 4 input LUTs</td>
<td>5151 out of 2400 214%</td>
</tr>
<tr>
<td>Number used as logic</td>
<td>4223</td>
</tr>
<tr>
<td>Number used as Shift registers</td>
<td>928</td>
</tr>
<tr>
<td>Number of IOs</td>
<td>3074</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>3074 out of 140 2195%</td>
</tr>
<tr>
<td>IOB Flip Flops</td>
<td>96</td>
</tr>
<tr>
<td>Number of GCLKs</td>
<td>1 out of 4 25%</td>
</tr>
<tr>
<td>Minimum input arrival time before clock</td>
<td>22.814ns</td>
</tr>
<tr>
<td>Maximum output required time after clock</td>
<td>8.128ns</td>
</tr>
</tbody>
</table>

A. Simulation Results:
Below shows the RTL schematic after the synthesis in the Xilinx 10.1 simulation for the 32*32 bit (1024 bits) matrix coprocessor operations like transpose, addition, subtraction, multiplication etc.
CONCLUSION

32*32 bit matrix is dealing with total 1024 bits at a time for different matrix operations. These operations are very much useful in image processing application when we have to deal with pixel values of the image. It is also useful in satellite applications where satellite has to travel a long distance with different coordinates. The wide use of 32*32 bit matrix makes itself very important in the application of communication like cryptography for authentication process. It can also be used for digital communication where one has to deal with channel coding techniques. Overall performance of this matrix operation like addition, subtraction, multiplication, transpose makes the 32-bit RISC Co-processor which is the part of the Advanced RISC machine.

REFERENCES:
1. Reconfiguring Block RAMs via JTAG http://support.xilinx.com/xlnx/xweb/xil_tx_display.jsp?TechX_ID=krs_blockRAM
2. Xilinx Embedded Development Kit http://www.xilinx.com/ise/embedded_design_product/platform_studio.htm
5. IEEE international conference on Double Precision Hybrid-Mode Floating-Point FPGA CORDIC Co-processor by Jie ZHOU Yong DOU Yuanwu LEI Jinbo XU Yazhuo DONG, Department of Computer Science National University of Defense Technology, Changsha, P.R.China.
6. IEEE international conference on A 21.54 Gbits/s Fully Pipelined AES Processor on FPGA by Alireza Hodjat and Ingrid Verbauwhede, Electrical Engineering Department, University of California, Los Angeles.

