PERFORMANCE ANALYSIS OF AN EFFICIENT ARMV8 PROCESSOR

B. Ravali, N. Mathan and T. Ravi
Department of Electronics and Communication Engineering, Sathyabama University, Chennai, Tamil Nadu, India
E-Mail: ravali2227@gmail.com

ABSTRACT

As everyone is familiar with the processors which places major role in computers, mobiles, tablets, smart phones etc. In past, the processors are of 16-bit, 32-bit. Based on the type of processor used is going to effect the performance of the device. Present scenario represents that more devices prefer 64-bit processors. The type of processors not only affects the performance of devices, it also guides types of software it use. Use of 64-bit OS even supports 32-bit operating system.

Keywords: ARMv8, Instruction set, execution unit, clock signal, performance, fetch, decode, CPU, smart phones.

INTRODUCTION

The processor is also known as CPU (Central processing unit). It's also the brain of the computer. Now-a-days processors became platform to smart phones, tablets.

Among the processors ARM processors plays major role in the performance of the system

Due to ARM processor smart phones, tablets and few laptops are becoming more popular because of their high performance, efficiency and gaming. ARM processors are categories as 34-bit, and 64-bit. The life of ARM started as part of BCC computer, and now ARM designs chip for iPad (apples).

The first ARM was established in Cambridge University in 1978. ARM RISC processor was first developed by Acorn group in 1985 [1]. Microcontroller and Microprocessors are replaced by the latest technology named ARM processor. Basically ARM is categorised as 16 bit/ 32 bit Processors. The heart of the advanced digital products like mobile phones automotive systems digital cameras home networking and wireless technologies is ARM processor.

The main reason for ARM processor is:

- ARM processors are particularly used in portable devices, and they became more popular due to its low power consumption and equitable performance.
- ARM processors are quick and easy to use and even they are more efficient compared to other processors. ARM processors are good enough to produce high performance. The ARM processor consumes less power and it is of low cost.

The features of ARM series are as, the ARMv1 Architecture is of Software interrupts, 26-bit address bus, Data processing is slow, and it support byte, word and multiword load operations [2]. ARMv2 consists of 26-Bit address bus, Automatic instructions for thread synchronization, Co-processor support [4]. ARMv3 is of 32-Bit addressing, it supports Multiple data [6] [7]. ARMversion3 is faster than ARMv1 and ARMv2.ARMv4 feature are 32-bit address space and It also support T variant with 16 bit THUMB instruction set, even it supports M variant with long multiply gives result of 64 bit [8]. ARMv5 has ARM THUMB interworking, it support CCL instructions, it support E variant with Enhanced DSP Instruction set and It also supports S variant with Acceleration of Java byte code execution. ARMv6 supports memory system, It also as the feature of single instruction multiple data.ARMv7 supports 32-bit data and it is also based on applications, here core is of Cortex series.

The latest version of processor is ARMv8 processor which is of 64-bit, till now processor is of 34-bit. This paper is all about the ARMv8 processor. The step towards ARMv8 is due to the performance factor. As the number of calculations per second changes based on their bit size, 64-bit ARM processor can perform more calculations compared to 32-bit ARM processor [9]. By this the performance or the speed of the system increases in 64-bit processors rather than 32-bit processors. The main reason is that a 32-bit processor support the memory size of 3GB to 4GB (RAM), whereas 64-bit processors have the capacity to store is beyond 4GB [10]. Memory is major part for software programs that is mostly used in graphic design, and video editing, these programs have to perform many calculations to extract their images. ARMv8 has the capability to support dual core, quad core, six, and eight cores.

Due to the cheaper cost manufactures are designing 64-bit processors compared to 32-bit. Even number of users preferring 64-bit OS and programs increased [11]. So, 64-bit processors are becoming more common place in home computers.

In past, the instructions, Architecture is of 32-bit. Now ARM announced a new version i.e. ARMv8 with 64-bit. And this architecture is known as 'AArch64'. ARMv8 processor supports both 32-bit and 64-bit applications. Even there is an advantage of using ARMv8processor; it runs the 32-bit programs faster than in ARMv7. Cortex A57 and Cortex A53 core are included in ARMv8 chip. These are the central units of SOC which power the next generation of smart phones and tablets.

In processors register banks are wide important. These registers are used to store the address and the numbers that are used while computing. Register are used even for simple applications so these register will work all the time. The numbers and the address storage increases with greater speed because of 64-bits wide. It doesn't affect the efficiency of the system. ARMv8 processor are also better in parallel processing which make the chip to use multiple cores that reduces the utility of the power.

Application processors with high performance for feature operating system are Cortex series. The processor delivers both 32-bit and 64-bit combined is happening by the new processors which are of Cortex series. Cortex-A72 individually performs this, it can be even processed by both Cortex - A53 and Cortex-A57 [14] [15]. Cortex-A series processors are became responsible for next generation mobiles.

AArch64 belongs to ARMv8-A 64-bit execution state, that uses 64-bit general purpose registers, stack pointer (SP), exception link registers (ELR), and a 64-bit program counter. It provides a single instruction set, A64.

INSTRUCTION SET

AArch64

AArch64 state supports only A64 instruction set. The 32-bit instruction encode is used by A64. The hardware has rejected the original implementation to adapt an existing decode table for sharing the decoder table between 32- and 64-bit instruction sets. This helps in simplifying the decode table by providing clean decode structure with contiguous bit fields for operands and immediate values [12]. Another important advantage is providing JIT compilers with important acceleration techniques which inturn help in high performance of applications (For example, Web browsing). The independent decode also permits some of the more advanced branch prediction techniques.

The higher number of general purpose registers provide improved scheduling options for the increasingly complex algorithms which are common in various software codecs. Therefore, for this purpose, A64 ISA (Instruction set architecture) is introduced with thirty one 64-bit general purpose registers [13]. Although the virtual rename register pooling introduced in the Cortex-A9 provided hardware with an automatic way of unrolling small loops, it is not having higher number of general purpose registers.

Another significant change in the A64 ISA is actually removing the LDM/STM (load/store multiple) instructions. This reduces the cost complexity in the long time implementation of an efficient processor's memory system when compared with the original RISC goals of the ARM ISA.

LITERATURE SUMMARY

In this paper, it is the first DENVER CPU with 64-bit processing. This CPU comes under ARMv8; here the execution unit is of seven superscalar units instead of three superscalar units. This increases the performance of the system [16]

The DENVER CPU can execute seven instructions per cycle and attain clock speed of up to 2.5GHZ.In paper [17] speaks about the Potenza, the first generation ARMv8 processor. Potenza is an integrated design unit. It was designed to be scalable for different server configuration. This processor uses Mesh-on-chip with wide superscalar micro-architecture.

PROPOSED WORK

The increase in performance of the system can be achieved by proper clocking signal. The clock signal of each component is enabled properly to reduce synchronization problems.

The proposed system is used to implement the ARM-V8 pipelined architecture to execute the 10 instruction set in a single clock cycle. Thus in a single clock cycle the three fetch, decode and the execution process of pipelined structure will be done.

In the proposed system instruction set is executed based on the requirement, it may be integer/floating point. Based on the clock signal the input of 64-bit is fetched in parallel processing form. Another clock pulse is to enable decode instruction set, and a clock pulse is used to execute the instruction sets.

There are three internal clock which are used to enable its components (input data, instruction set, ALU). All these clocks are enabled by using the other clock which acts like control clock. When this control clock is on positive level cycle both the fetching and decoding takes place in parallel. When the control clock pulse is negative level cycle then the ALU executes the instruction sets. The moto of this paper is to achieve an execution of 10 instruction set in a single clock cycle which increases the speed of the performance. Finally the ARM-V8 based pipelined architecture with multiple instruction set will be implemented in a single clocking signal.

The 10 instructions are arithmetic, logical operator, instruction decoder, addresses generator and instruction, address, data input, data output register element and increment counter. It also reduced the power consumption.

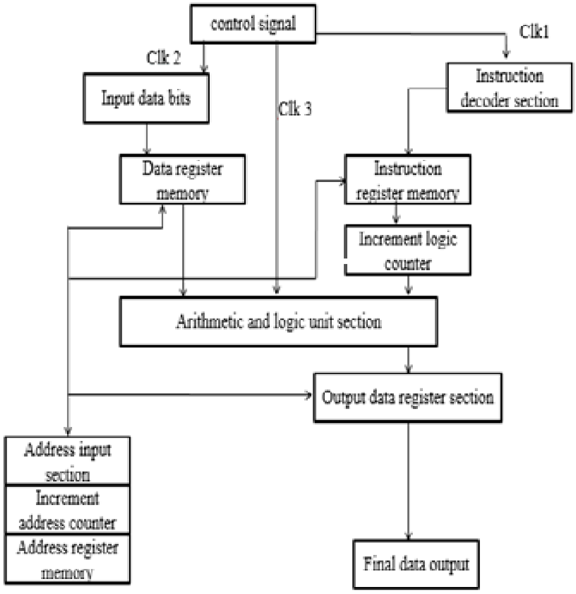


Figure-1. Proposed ARMV8 processor.

RESULTS AND DISCUSSIONS

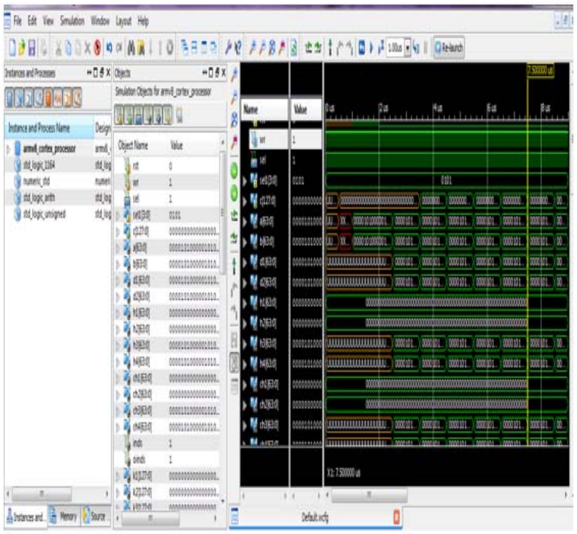


Figure-2. Result of 10 instructions per cycle.

The above Figure-2 represents the output waveform of 10 instructions per cycle.

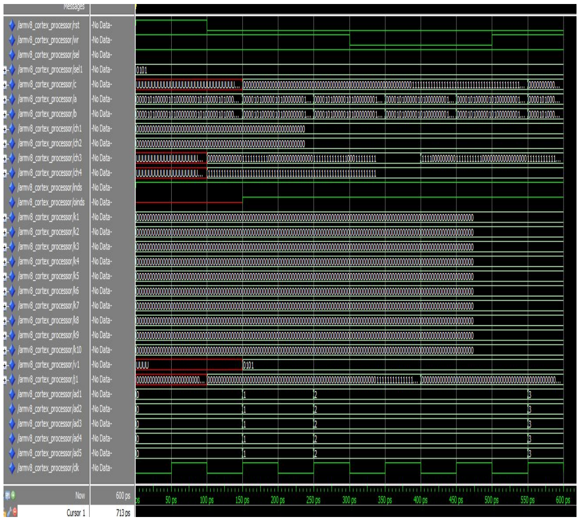
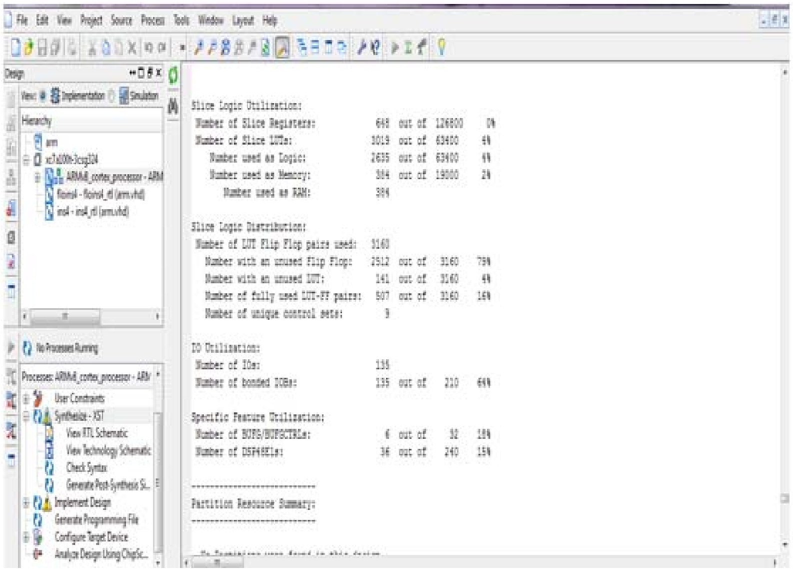


Figure-3. Result of 10 instructions per cycle with clock signal.

The above Figure-3 represents the execution unit of 10 instructions per cycle.

Table-1. Device utilisation and time delay.



File Edit View Project Source Process Tools Window Layout Help
Design
View: Implementation Simulation
Hierarchy
arm
xc7a100t-3csg324
ARMv8_cortex_processor - ARM
flioins4 - flioins4_rtl (arm.vhd)
ins4 - ins4_rtl (arm.vhd)
No Processes Running
Processes: ARMv8_cortex_processor - ARM
User Constraints
Synthesize - XST
View RTL Schematic
View Technology Schematic
Check Syntax
Generate Post-Synthesis Si...
Implement Design
Generate Programming File
Configure Target Device
Analyze Design Using ChipSc...

```
Slice Logic Utilization:
 Number of Slice Registers:            648  out of  126800    0%
 Number of Slice LUTs:                3019  out of   63400    4%
    Number used as Logic:             2635  out of   63400    4%
    Number used as Memory:             384  out of   19000    2%
       Number used as RAM:             384

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:  3160
   Number with an unused Flip Flop:   2512  out of    3160   79%
   Number with an unused LUT:          141  out of    3160    4%
   Number of fully used LUT-FF pairs:  507  out of    3160   16%
   Number of unique control sets:        9

IO Utilization:
 Number of IOs:                        135
 Number of bonded IOBs:                135  out of     210   64%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:               6  out of      32   18%
 Number of DSP48E1s:                    36  out of     240   15%

---------------------------
Partition Resource Summary:
---------------------------

  No Partitions were found in this design.
```

CONCLUSIONS

In this work, the design of efficient architecture based ARM-V8 pipelined architecture with multiple instruction set processor improve the system performance level. Thus the system is to design ARM-V8 pipelined processor architecture and this design is to implement the 10-instruction processing with reduced the register architecture. It consists of arithmetic, logical operator, instruction decoder, addresses generator and instruction, address, data input, data output register element and increment counter. It is designed to analysis the clocking sequence level and to reduce the processing time level.

REFERENCES

[1] www.slideshare.net/ARM_Based_Group/beginning-in-arm-architectures.

[2] everything2.com/title/ARM1.

[3] www.heyrick.co.uk/armwiki/The_ARM_family#The_ARM1_.28ARMv1.29.

[4] web.archive.org/web/20070809230809/http://www.arm.com/pdfs/Apps11vC.html.

[5] www.slideshare.net/yayavaram/unitii-arm-architecture?from_action=save.

[6] homepages.thm.de/~hg10013/Lehre/MMS/WS0304_SS04/Ioannis/PDF/arm.pdf.

[7] www.slideshare.net/ARM_Based_Group/beginning-in-arm-architectures.

[8] http://www.computerhope.com/issues/ch001498.html.

[9] http://arc.applause.com/cards/arm-64-bit-processors-future-mobile/.

[10] http://www.makeuseof.com/tag/64-bit-computing/.

[11] https://people.mozilla.org/~sstangl/arm/AArch64-Reference-Manual.pdf.

[12] http://www.arm.com/files/downloads/ARMv8_white_paper_v5.pdf.

[13] www.arm.com/products/processors/cor tex-a/index.php.

[14] community.arm.com/groups/processors/blog/2015/11/05/introducing-cortex-a35-arms-most-efficient-application-processor.

[15] Boggs, D.; Brown, G.; Tuck N.; venkatraman, K. S. Micro. 2015. Denver: N Vidia first 64-bit ARM processor. IEEE Journals and Magazines. 35(2): 46-55,

[16] Alfred Yeung, Hamid Partovi, Qawi Harvard, Luca Ravezzi, John Ngai, Russ Homer, Matthew Ashcraft and Greg Favor. 2014. A 3GHz 64b ARM v8 Processor in 40nm Bulk CMOS Technology,