Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, No. 1, JAN - JUNE 2023

DESIGN OF ALU MULTIPLEXER IMPLEMENTATION IN SRAM ARCHITECTURE

Dr.R.Yadagiri Rao[1], Dr.D.Lakshmaiah[2], G.Anusha[3], K.Aishwarya[4], k.Sai Kumar[5], M.Hinduja[6], M.Sai sahasri[7]

[1]Professor, Department of CSE, Sri Indu Institute of Engineering & Technology, Hyderabad

[2]Professor, Department of ECE & HOD, Sri Indu Institute of Engineering & Technology, Hyderabad

[3]Assistant Professor, Department of ECE, Sri Indu Institute of Engineering & Technology, Hyderabad

[4,5,6,7] IV[th] Btech Student, Department of ECE, Sri Indu Institute of Engineering & Technology, Hyderabad

ABSTRACT

A processor is a small chip that resides in computers and other electronic devices. Its basic job is to receive input and provide the appropriate output. While this may seem like a simple task, modern processors can handle trillions of calculations per second. ModernCPUs often include multiple processing cores, which work together to process instructions. While these "cores" are contained in one physical unit, they are actually individual processors. This project is going to design & simulate a 8-bit Microprocessor which is near to Reduced Instruction Set Computing (RISC) architecture based processor. The purpose of RISC microprocessor is to execute a minuscule batch of instructions. The designing process is using modules like ALU, Control Unit, Program Counter, MUX, Memory, Register File by using the Verilog Hardware Description Language (HDL).

Microprocessor is a programmable, multipurpose electronic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output. It executes the sequence of instructions one after the other.

1.INTRODUCTION

The microprocessor follows a sequence: Fetch, Decode, and then Execute. Initially, the instructions are stored in the memory in a sequential order. The microprocessor fetches those instructions from the memory, then decodes it and executes those instructions till STOP instruction is reached. Later, it sends the result in binary to the output port. Between these processes, the register stores the temporarily data and ALU performs the computing functions.

Microprocessor is a programmable, multipurpose electronic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output.CISC architecture and Reduced Instruction Set Computing i.e. RISC architecture. The concept of CISC is based on Instruction Set Architecture (ISA) design that redoubles performing further with several instructions utilizing changeable number of operands and an out spread variation of addressing modes in disparate locations in its Instruction Set. Thus causing them to have varying execution time and lengths thereby authoritatively mandating an

intricate Control Unit, which inhabits an immensely existent region on the chip.Compared with their CISC analogue, RISC processors typically support a minuscule set of instructions. A display that just a poses RISC processor with CISC processor, the number of instructions in a RISC Processor is low while the number of general purpose registers, addressing modes fixed instruction length and load-store architecture is more this in turn facilitates the execution of instructions to be carried out in a short time thus achieving higher overall performance.

Currently, the efficiency of the RISC processors is generally accepted to be greater than that of their CISC counterparts. Before their execution the instructions are translated into RISC instructions in even the most popular CISC processors. The attributes mentioned above accentuate the design strength of RISC in the market for embedded systems known as "system-on-a-chip (SoC)". The premier microprocessors exhibiting reduced instruction set are SPARC, ARM, MIPS and IBM's PowerPC. RISC processor typically has load store architecture. This denotes there are two instructions for accessing memory which are a load instruction set to load data from the memory and store instruction set to Write Back (WB) the data into memory.

1.1 Characteristics of RISC

The major characteristics of a RISC processorare as follows:

- It consists of simple instructions.
- It supports various data-type formats.
- It utilizes simple addressing modes and fixed length instructions for pipelining. It supports register to use in any context.
- One cycle execution time.
- "LOAD" and "STORE" instructions areused to access the memory location. It consists of larger number of registers.
- It consists of less number of transistors.

2. LITERATURE SURVEY

A continuous scaling in VLSI technology with each method generation causes Associate in Nursing ecponential increase in power dissipation that imposes a elementary limitation to increasing performance and practically in high performance microprocessors and practically in high perfomance microprocessors escape power has become thus vital that it can be neglected.The appearance of computationally advanced moveable devices and developments in wireless communication aditionally demand a discount of power dissipation while not sacrificing abudant perfomance.

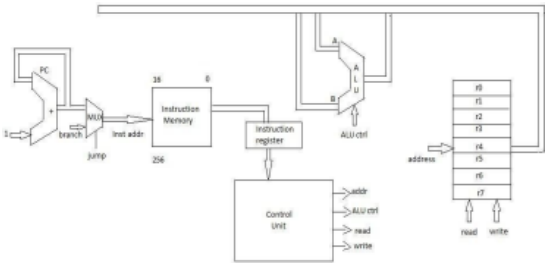
B. G. Kumar et al. [1] proposes that FinFETs are superior to CMOS, in terms of voltage, area and power. The fin that wraps around the channel of a FinFET gives electrical control. The gate in CMOS devices loses channel control, resulting in higher leakage current, parameter fluctuations, worse reliability, yield, and short channel effects. The most often utilised memory cell for preservingdata until the power supply is activated is the Static Random-Access Memory [SRAM] cell. The primary aim is optimizing the power and delay constraints of the SRAM cell. Both FinFETtechnology, as well

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, No. 1, JAN - JUNE 2023

as adiabatic logic has been used to meet the required conditions. Utkarsh Tripathi et al.

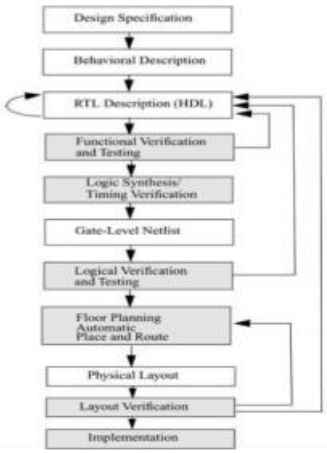
[2] devised an adder circuit including a full adder CMOS circuit and it is developed utilising MOSFETs with a length of 32nm and in FinFETs with 28 transistors in MOSFET and FinFET. Then, a HSPICE software simulation is done and the adder's performance characteristics, such as average power and delay are determined in both FinFET and MOSFET counterparts. Thus, in conclusion, FINFET provided the best results in terms of average power consumption and delay. DGFETs have emerged as a potential way to keep the process going. FinFET has emerged as a leader among DGFETs where scaling up of technology is concerned.

3. EXISTING SYSTEM



Computational RAM is a processor-in-memory architecture that makes highly effective use of internal memory bandwidth by pitch-matching simple processing elements to memory columns. Computational RAM can function either as a conventional memory chip or as a SIMD (single-instruction stream, multiple-data stream) computer. When used as a memory, computational RAM is competitive with conventional DRAM in terms of access time, packaging and cost. Adding logic to memory is not a simple question of bolting together two existing designs. The paper considers how computational RAM integrates processing power with memory by using an architecture that preserves and exploits the features of memory.

VLSI DESIGN FLOW



4. VERILOG HARDWARE DESCRIPTIONLANGUAGE

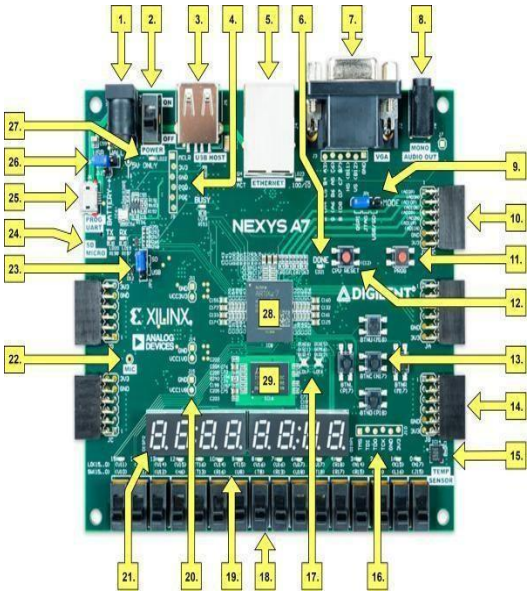
4.1 Hardware Description Language

Hardware description language (HDL) is a specialized computer language used to program electronic and digital logic circuits. The structure, operation and design of the circuits are programmable using HDL. HDL includes a textual description consisting of operators, expressions, statements, inputs and outputs. Instead of generating a computer executable file, the HDL compilers provide a gate map. The gate map obtained is then downloaded to the programming device to check the operations of the desired circuit. The language helps to describe any digital circuit inthe form of structural, behavioural and gate level and it is found to be an excellent programming language for FPGA s and CPLD
s. The three common HDL s are Verilog, VHDL, and System C.

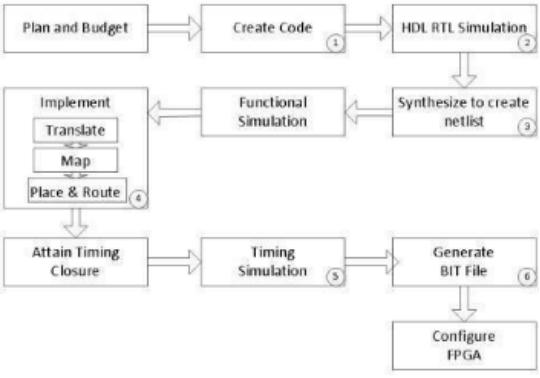
5. VIVADO AND FPGA

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, No. 1, JAN - JUNE 2023

This tutorial guides you through the design flow using Xilinx Vivado software to create a simple digital circuit using Verilog HDL. A typical design flow consists of creating model(s), creating user constraint file(s), creating a Vivado project, importing the created models, assigning created constraint file(s), optionally running behavioral simulation, synthesizing the design, implementing the design, generating the bitstream, and finally verifying the functionality in the hardware by downloading the generated bitstream file. You will go through the typical design flow targeting the Artix-100 based Nexys4 board. The typical



design flow is shown below. The circled number indicates the corresponding stepin this tutorial.



5.1ADVANTAGES OF FPGA

It requires very small time; starting from design process to functional chip.No physical manufacturing steps are involved in it.

The only disadvantage is, it is costlythan other styles.

5.2 NEXYS A7 BOARD

The Nexys A7 board is a complete, ready-to- use digital circuit development platform based on the latest Artix-7™ Field Programmable Gate Array (FPGA) from Xilinx®. With its large, high-capacity FPGA, generous external memories, and collection of USB, Ethernet, and other ports, the Nexys A7 can host designs ranging from introductory combinational circuits to powerful embedded processors. Several built-in peripherals, including an accelerometer, temperature sensor, MEMs digital microphone, a speaker amplifier, and several I/O devices allow the Nexys A7 to be used for a wide range of designs without needing any other components.

NEXYS BOARD

6. RESULT

A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing – hence the term "field-programmable". The FPGA configuration is generally specified using a hardware description language (HDL), similar to that used for an application-specific integrated circuit (ASIC). Circuit diagrams were previously used to specify the configuration, but this is increasingly rare due to the advent of electronic design automation tools.

7.CONCLUSION

In Digital systems for processing the data processors plays a major role.It reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output.In this we implemented a basic RISC processor which can execute all Arithmetic, Logical, Memory Operations and some Branch instructions using Verilog Hardware Description language.

As the machine codes which are taken by the processor for execution are not human friendly to write a program. Along with this processor we also implemented an Assembler which converts assembly level codes to machine codes using Python programming language which makes easier to write programs without any difficulty for this processor.

In order to write Verilog Code with Syntax highlighting we used Visual Studio Code which is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntaxhighlighting and intelligent code completion.For simulating Verilog Code we used Icarus Verilog which is a Verilog simulation and synthesis tool.

8.BIBLIOGRAPHY

[1]Ramesh Gaonkar, Author, “Microprocessor Architecture, Programming and Applications with the 8085 6/e”, Penram International Publishing. (2013).

[2]M. Morris Mano, Author, “Computer System Architecture, 3e”, Pearson Education India. (2007).

[3]Priyavrat Bhardwaj and Siddharth Murugesan, “Design & Simulation of a 32-Bit RISC Based MIPS Processor Using Verilog”, International Journal of Research in Engineering and Technology. vol. 5 (2016).

[4]Ramandeep Kaur and Anuj, “8 Bit RISC Processor Using Verilog HDL”, International. Journal of Engineering Research and Applications. vol. 4, no. 3, (2014), pp. 417- 422.

[5]R Uma "Design and performance analysis of 8 bit RISC processor using Xilinx tools", International Journal of Engineering Research and Applications (2012), pp 053-058.

[6]B. Rajesh Kumar, Ravisaketh and Santha Kumar, "Implementation of A 16-bit RISC Processor for Convolution Application", Research India publications, (2014), pp 441-446.

[7]M. Morris Mano, Author, "Digital Design", 5 edition, Prentice Hall. (2012).

[8]MD.Shabeena Begum and M.Kishore Kumar, "FPGA based implementation of 32 bit risc processor", International Journal of Engineering Research and Applications (IJERA) . vol. 1, pp 1148-1151.

[9]Oscar Camacho Nieto and Jorge A. Huerta Ruelas "Design of a General Purpose 8- bit RISC Processor for Computer Architecture Learning", Research gate publication. vol. 19, no. 2, (2015), pp 371–385.

Oztekin, H., Temurtas, F., & Gulbag, A. "Microprocessor architecture design on reconfigurable hardware as an educational tool.", IEEE Symposium on Computers & Informatics, (2011), pp. 385–389.