

Augmentation of process handling by VLSI and Pipeline to make computer processing more efficient

Prof. Nirmala M. Thorwe¹, Prof. V.V. Giri², Mis. Uma Tarlekar³

¹SGP- Atigre, Kolhapur, India, nilu.chouhan@gmail.com

²SGP- Atigre, Kolhapur, India, virat.giri@gmail.com

ABSTRACT

Enhancement in existing VLSI technology is very challenging because available chip in market is very efficient, portable, powerful, easy to manufacture. Instead of this there is some possibility to enhance VLSI. By using Enhancement Mode Transistors in which power consumption can be reduced, circuitry implementation is also very simple, capacity of chip is also increase, which is cost effective.

Pipeline concept can make system power full. But some disadvantages are also there. Proper study can help to take advantages of technology.

Key words: IC, VLSI, Micro Processors, Pipeline.

I. INTRODUCTION

Very-large-scale integration (VLSI) is the process of creating an integrated circuit (IC) by combining thousands of transistors on a single chip. VLSI began in the 1970s when complex semiconductor and communication technologies were being developed [1]. The microprocessor is a VLSI device.

First, need to know that what is IC, IC is very common in now day. It is stand for Integrated Circuit. IC is used in various places that are PCs, Robots, Electronic devices, Toys , High-tech home appliance, Auto control devices, Bionic body part and so on. Most ICs had a limited set of functions they could perform.

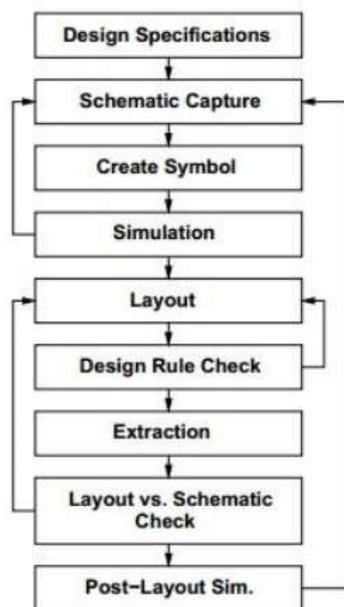


Figure 1: Simplified VLSI Design Flow

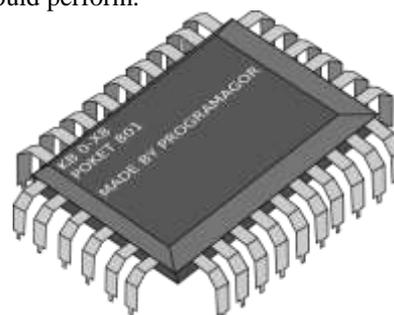


Figure 2: Simple chip daigram

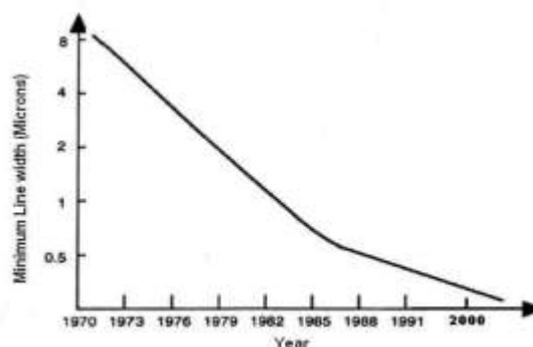


Figure 3: The graph shows a significant decrease in the size of the Chip.

The size of the chip in recent years shows the advancements in the VLSI technology.

An electronic circuit might consist of a CPU, ROM, RAM and other glue logic. VLSI lets IC designers add all of these into one chip.

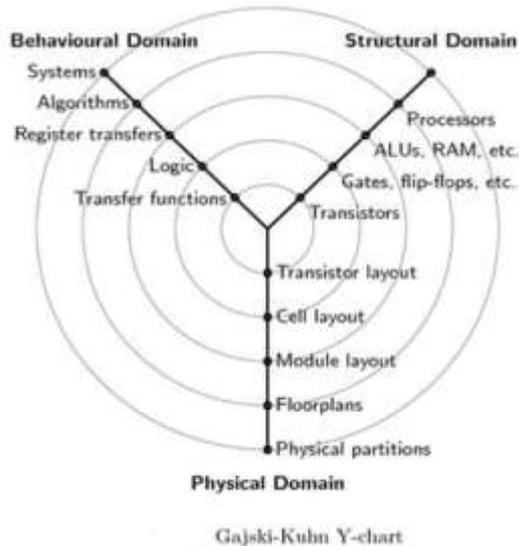


Figure 4: Y chart

III. AVAILABLE VLSI

VLSI stands for Very-large-scale integration in which developer indent to increases chip's functionality. In computer chip every numerical calculation is depicted by SUM /ADD.

3.1 Design Hierarchy-Structural

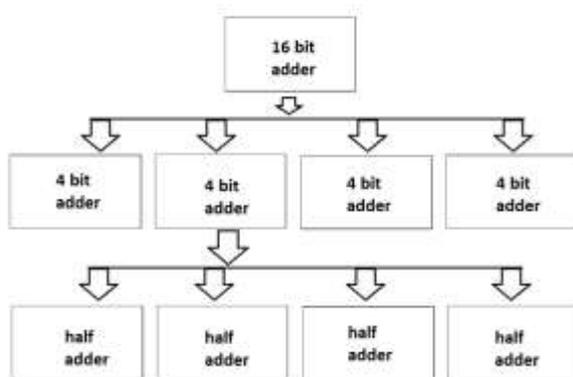


Figure 5: Structural hierarchy of 16 bit adder circuit

Here, the whole chip of 16 bit adder is divided into four modules of 4-bit adders. Further, dividing the 4-bit adder into 1-bit adder or half adder. 1 bit addition is the simplest designing process and it is internal circuit is also easy to fabricate on the

chip. Now, connecting all the last four adders, we can design a 4-bit adder and moving on, we can design a 16 - bit adder.

IV. ENHANCEMENT MODE TRANSISTORS:

1. Enhancement mode transistor action:

To understand the enhancement mechanism, Let us consider the enhancement mode device. In order to establish the channel, a minimum voltage level called threshold voltage (V_t) must be established between gate and source. Where a channel is established but no current flowing between source and drain ($V_{ds} = 0$).

2. CMOS fabrication:

CMOS fabrication is performed based on various methods, including the p-well, the n-well, the twin-tub, and the silicon-on-insulator processes. Among these methods the p-well process is widely used in practice and the n -well process is also popular, and particularly as it is an easy retrofit to existing nMOS lines.

3. Alternative forms of pull –up :

Generally the inverter circuit will have a depletion mode pull-up transistor as its load.

4. The BiCMOS inverter:

A BiCMOS inverter, consists of a PMOS and NMOS transistor (M2 and M1), two NPN bipolar junction transistors,(Q2 and Q1), and two impedances which act as loads(Z2 and Z1).

V. COMPARISON OF BICMOS AND C-MOS TECHNOLOGIES

The BiCMOS gates perform in the same manner as the CMOS inverter in terms of power consumption, because both gates display almost no static power consumption.

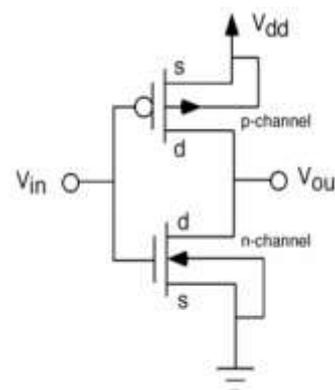


Figure 6: CMOS Inverter Circuit

Table 1: Comparison of BiCMOS and C-MOS technologies

CMOS	BiCMOS
CMOS devices does not have degradation in low supply voltage region.	BiCMOS devices have speed degradation in the low supply voltage region.
Making CMOS consume less power than BiCMOS.	Making BiCMOS consume more power than CMOS.
For very low capacitive loads, the CMOS gate is faster.	Its BiCMOS counterpart due to small values of C_{int} , it is slower.
Driving larger capacitive loads CMOS consuming more power.	Driving larger capacitive loads makes BiCMOS in the advantage of consuming less power.
CMOS is having smaller manufacturing complexity.	BiCMOS is having greater manufacturing complexity.
Driving large capacitive loads CMOS is not effective.	The BiCMOS inverter exhibits a substantial speed advantage over CMOS inverters, especially when driving large capacitive loads.
Implementation of internal gates in CMOS is effective than BiCMOS.	This makes BiCMOS ineffective when it comes to the implementation of internal gates for logic structures such as ALUs.
The construction of CMOS inverter chains are needed to drive large capacitance loads.	Which is not needed in BiCMOS.

VII. INSTRUCTION PIPELINE

In Von Neumann architecture, the process of executing an instruction involves several steps like fetch, decode, execute and store. First, the control unit of a processor fetches the instruction from the cache (or from memory). Then the control unit decodes the instruction to determine the type of operation to be performed.

When the operation requires operands, the control unit also determines the address of each operand and fetches them from cache (or memory). Next, the operation is performed on the operands and, finally, the result is stored in the specified location. An instruction pipeline increases the performance of a processor by overlapping the processing of several different instructions. Often, this is done by dividing the instruction execution process into several stages.

Suppose we consider 3 stages.

1. **Fetch:** Load data from memory.
2. **Decode:** Data translation and interpretation.
3. **Execute:** Processing and termination.

VI. CLASSIFICATION OF LOGIC CIRCUITS

Logical circuits is design by using logical gates (AND, OR, NAND, XOR). Basic classification of circuits shown in figure:

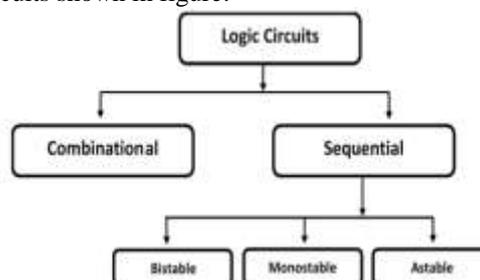


Figure 7: Classification of Logic Circuits

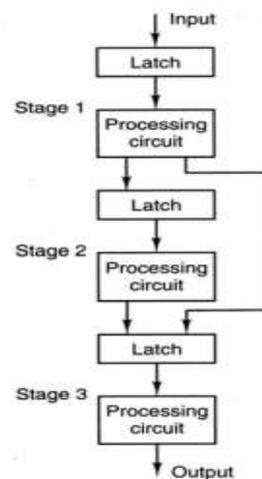


Figure 8: A three-stage dynamic pipeline

7.1 Pipeline Structure

The pipeline design technique decomposes a sequential process into several sub processes, called stages or segments. A stage performs a particular function and produces an intermediate result. It consists of an input latch, also called a register or buffer, followed by a processing circuit. (A processing circuit can be a combinational or sequential circuit.)

The processing circuit of a given stage is connected to the input latch of the next stage (see Figure 12). A clock signal is connected to each input latch. At each clock pulse, every stage transfers its intermediate result to the input latch of the next stage. In this way, the final result is produced after the input data have passed through the entire pipeline, completing one stage per clock pulse.

7.2 Non-pipelining:

Non-pipelining system generally uses simple concept **Fetch** data, **Decode** data and **Execute**.

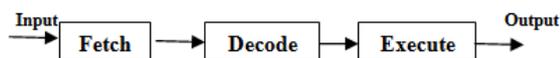


Figure 9: Non-pipeline system.

			Job1 out /			Job2 out /
Execute			J ₁			J ₂

In non-pipeline system, one phase is busy to perform job_i then remaining all phases/stages n_j-1 should be idle. In other words efficiency and performance of system is very low.[3]

7.3 Pipelining:

If pipeline concept is used then performance of the system will increase. When system processing is divided in to n phases/Stage then system can respond maximum n jobs at the same time.

			Job1 /	Job2 /	Job3 /	Job4 /	Job5
Execute			J ₁	J ₂	J ₃	J ₄	J ₅
Decode		J ₁	J ₂	J ₃	J ₄	J ₅	
Fetch	J ₁	J ₂	J ₃	J ₄	J ₅		
Time	1	2	3	4	5	6	

Figure 11: Pipeline system.

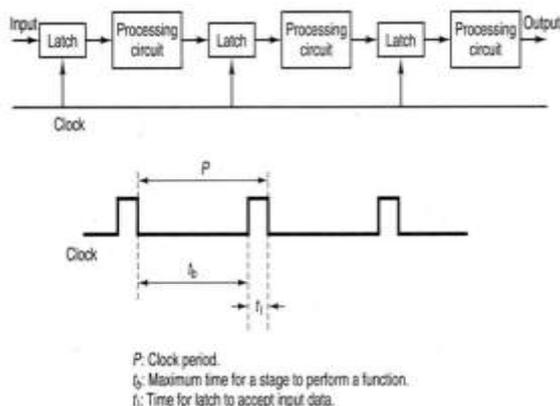


Figure 12: Basic structure of a pipeline

The period of the clock pulse should be large enough to provide sufficient time for a signal to traverse through the slowest stage, which is called the bottleneck (i.e., the stage needing the longest amount of time to complete).

In addition, there should be enough time for a latch to store its input signals. If the clock's period, P, is expressed as

$$P = t_b + t_l$$

Then t_b should be greater than the maximum delay of the bottleneck stage, and t_l should be sufficient for storing data into a latch.

Decode		J ₁		J ₂		
Fetch	J ₁			J ₂		
Time	1	2	3	4	5	6

Figure 10: Non-pipeline system chart

As shown in Figure 13, an instruction pipeline often consists of five stages, as follows:

- 1. Instruction fetch (IF):** Retrieval of instructions from cache (or main memory).
- 2. Instruction decoding (ID):** Identification of the operation to be performed.
- 3. Operand fetch (OF):** Decoding and retrieval of any required operands.
- 4. Execution (EX):** Performing the operation on the operands.
- 5. Write-back (WB):** Updating the destination operands

On the other hand, in a non-pipelined processor, the above sequential process requires a completion time of

$$T_{seq} = n * \sum_{i=1}^m \tau_i$$

Where, τ_i is the delay of each stage[1]. For the ideal case when all stages have equal delay $\tau_i = \tau$ for $i = 1$ to m , T_{seq} can be rewritten as

$$T_{seq} = n * m * \tau.$$

If we ignore the small storing time t_l that is required for latch storage (i.e., $t_l = 0$), then

$$T_{seq} = n * m * P.$$

Now, speedup (S) may be represented as:

$$S = T_{seq} / T_{pipe} = n * m / (m + n - 1).$$

In figure 13 instruction executions divided in to 5 stages but in figure 8 shows 3 stages. Difference between both systems is performance (Throughput). If stages increase then the ability of parallelism is also rise.

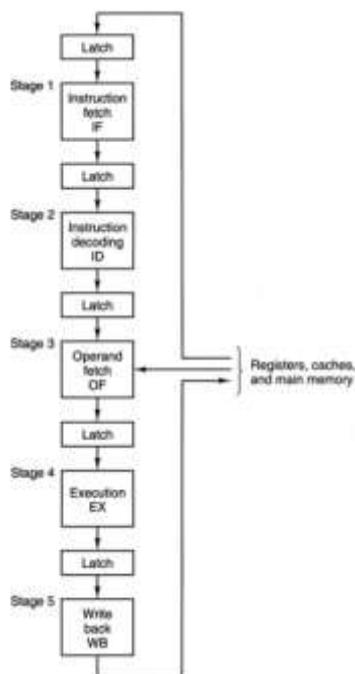


Figure 13: Stages of an instruction pipeline.

Summary:

Study of both concepts (*see*, in figure 10, and figure 11) shows that in 6 Time-cycle non-pipeline system can execute only 2 jobs/instructions which is consider as poor throughput of processor. But in pipeline system 5 jobs are responding and 4 jobs executed/completed in same span of time that is 6 time-cycle. In this case result shows better performance with pipeline. For VLSI, Chip size small then it is easy to fit and use. Main advantage is communication, it takes less time so functioning of device is also fast. But designing is complex and time consuming.

VIII. CONCLUSION

In VLSI with very minor changes we can get good results in term of input, output, power, speed and so on. CMOS and BiCMOS is good example of IC. And we can take advantage of their features. Pipeline increases performance when pipeline applicable on it highest performance then result is outstanding but Value of *n* (number of stages) more than the limit then system suddenly decreases performance, because when distribution take more time than execution of instruction. It should be effective and efficient.

REFERENCES

- [1]. **IBM RESEARCH | RESEARCH AREAS | VLSI DESIGN** THU, 11 MAY 2017 08:45:00 GMT IBM RESEARCH VLSI DESIGN PUBLICATIONS MULTI-GHZ ... ISSCC DIGEST OF TECHNICAL PAPERS, FEBRUARY, 1999, PP. 88-89.
- [2]. **VLSI DESIGN - BOOKS DELIVERY** SAT, 20 MAY 2017 01:47:00 GMT VLSI DESIGN BY V.SGAD FROM TECHNICAL PUBLICATIONS, EC6601, 9789333206822
- [3]. **FUNDAMENTALS OF CMOS VLSI: A COMPREHENSIVE APPROACH** BY V.S ... TUE, 23 NOV 2010 23:56:00 GMT ABEBOOKS: FUNDAMENTALS OF CMOS VLSI: A COMPREHENSIVE APPROACH: ... PUBLISHED BY TECHNICAL PUBLICATIONS 0. ISBN 10: 9350993805 / ISBN 13: 9789350993804.
- [4]. **IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI ...** MON, 26 JUN 2017 22:17:00 GMT
- [5]. **VLSI TECHNICAL PUBLICATIONS BY VS BAGAD PDF DOWNLOAD** SAT, 17 JUN 2017 02:18:00 GMT VLSI TECHNICAL PUBLICATIONS BY VS BAGAD PDF DOWNLOAD REPLY. JASEELA123 SUPER MODERATOR. POSTS: 9,951 THREADS: 61 JOINED: OCT 2014 #2. 20-08-2016, 09:33 AM .