Parallel Processing Techniques: History and usage of MIPS approach for implementation of Fast CPU

Drashti Joshi [1], Pooja Thakar [2]

[1] M.E. Student, Communication System Engineering, SAL Institute of Technology and Engineering Research, Gujarat, India.
[2]Assistant Professor, Communication System Engineering, SAL Institute of Technology and Engineering Research, Gujarat, India.

ABSTRACT

Parallel Processing is a term used to denote a class of technique that are used to provide simultaneous data processing task for the purpose of increasing computational speed of a computer system. Instead of processing each instruction sequentially as in conventional computer, a parallel processing system is able to perform concurrent data processing to achieve faster execution time. The Purpose of parallel processing techniques such as Pipeline processing, Vector processing and Array processing is to speed up the computer processing capability and increase its throughput, that is, the amount of processing that can be accomplished during a given interval of time. The amount of hardware increases with parallel processing, and with it, the cost of the system increases, However, technological developments have been reduced hardware cost to the point where parallel processing techniques are economically feasible. Advantage of parallel processing will be implemented using MIPS architecture for efficient implementation and of course its cost.

Keyword: - *Parallel processing, Pipelining, MIPS, and Interlock.*

1. INTRODUCTION

Basic computer architectures have been designed using two philosophies : 1) CISC (Complex Instruction set computer) 2) RISC (Reduced Instruction set computer) CISC support variety of addressing modes ,variable number of operands in various location in instruction set, wide variety of instructions with varying lengths and execution time and thus demanding complex control unit, which occupies large area on chip. Intel x86, Motorola 68000, System-360 IBM processors uses CISC architectures. On contrast, RISC architecture support simplified addressing modes, small number of fixed length and faster instructions that can execute quickly, more number of general purpose registers and LOAD/STORE architecture and occupies less silicon consumption and provide net gain in performance and simple design compared to CISC. The number of semiconductor companies implementing RISC processors on “system on chip” such as Atmel AVR, Micro Blaze, ARM, SPARC, MIPS, IBM PowerPC [3][6][4].

Parallel processing can be viewed from various level of complexity. At the lowest level, we distinguish between parallel and serial operations by the type of registers used. Shift registers operates in serial fashion one bit at a time, while register with parallel load operate with all the bits of the words simultaneously [7]. The MIPS Microprocessor without interlocked pipeline stages) is a RISC architecture introduced for implementation of faster computer that takes the advantage of parallel processing technique, such a technique (pipelining) is established by dividing the data to be processed among different processing units. MIPS uses the pipelined configuration in which over lapping of instruction is possible during execution to achieve higher processing speed [8].

MIPS Architectural design remains the same for all MIPS processors while the implementation may differ like single cycle, multi cycle and pipelined implementation. The Stanford MIPS architecture made the pipeline structure purposely visible to the compiler and allowed multiple operations per instruction. Simple schemes for scheduling the pipeline in the compiler were described by Sites (1979), by Hennessy and Gross (1983) and by Gibbons and Muchnik (1969) [1].

2. PARALLEL PROCESSING TECHNIQUES:

Parallel processing computers are required to meet the demands of large scale computations in many scientific, engineering, military, medical, artificial intelligence and basic research areas. Followings are some representative applications of parallel processing computers: Numerical whether forecasting, computational aerodynamics, Finite element analysis, Remote sensing applications and Weapon research and defence [8].

1) Pipeline processing
2) Vector Processing
3) Array processors

2.1 Pipeline processing:

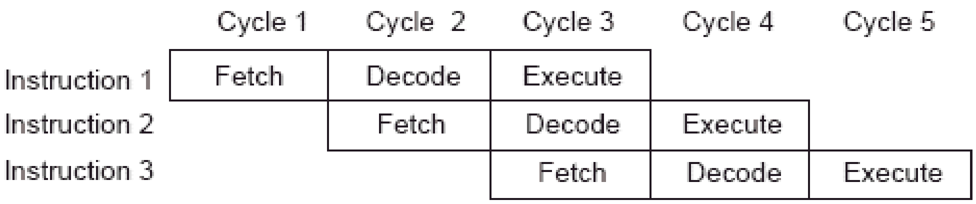
Pipelining is the technique of decomposing a sequential process in to sub operation with each sub process being executed in a special dedicated segment that operates concurrently with all other units [8]. Pipelining can be viewed as a combination of a processing units through which binary information flows. Total task is partitioned and each unit performs partial processing step. The results obtained are circulated to the next processing step to complete the execution. Finally the result is obtained after the data have passed through all segments. Major advantage of the pipeline is that several computations can be in progress in distinct segments at the same time.

The first general-purpose pipelined machine is considered to be Stretch, the IBM 7030 (1959). Stretch followed the IBM 704 and had a goal of being 100 times faster than the 704. The goal was a stretch from the state of the art that time – hence the nickname. The plan was to obtain a factor of 1.6 from overlapping fetch, code and execute, using a four-stage pipeline [1]. In Pipelining each segment consist of input register followed by the combinational circuit, Register hold the data and combinational circuit performs sub operation in a particular unit. Output of the previous stage is forward to the input register of the next segment.

1) **Three stage Pipeline:**

A three-stage pipeline is used, so instructions are executed in three stages:

- Fetch
- Decode
- Execute



	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
Instruction 1	Fetch	Decode	Execute		
Instruction 2		Fetch	Decode	Execute	
Instruction 3			Fetch	Decode	Execute

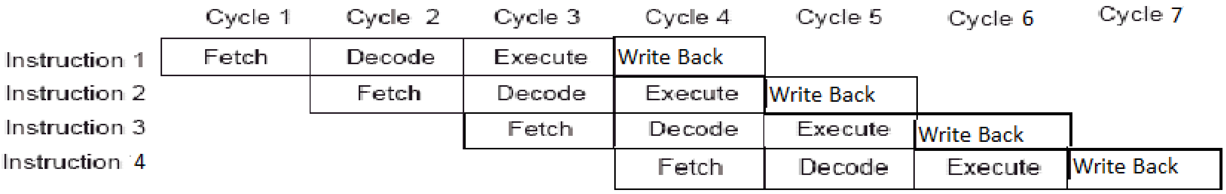
Fig.1 Three stage pipeline [13]

Fetch: The instruction is fetched from memory „

Decode: The instruction is decoded and the data path control signals prepared for the next cycle „

Execute: The operands are read from the register bank, shifted, and combined in the ALU and the result written back.

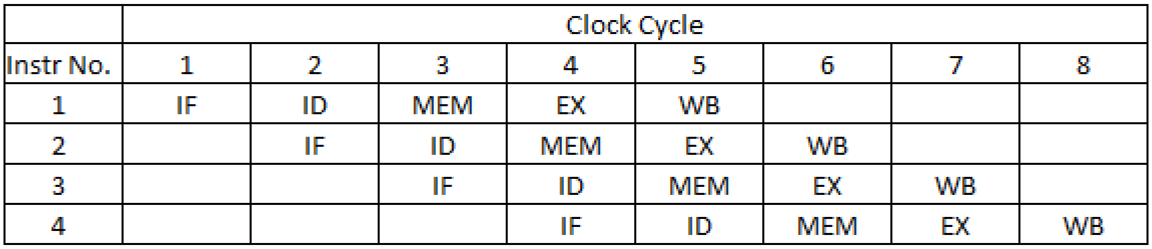
2) **Four Stage Pipeline:**



	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7
Instruction 1	Fetch	Decode	Execute	Write Back			
Instruction 2		Fetch	Decode	Execute	Write Back		
Instruction 3			Fetch	Decode	Execute	Write Back	
Instruction 4				Fetch	Decode	Execute	Write Back

Fig.2 Four Stage Pipeline [13]

3) Five Stage Pipeline:



	Clock Cycle							
Instr No.	1	2	3	4	5	6	7	8
1	IF	ID	MEM	EX	WB			
2		IF	ID	MEM	EX	WB		
3			IF	ID	MEM	EX	WB	
4				IF	ID	MEM	EX	WB

Fig.3 Five stage Pipeline [14]

Instruction Fetch:

The Program Counter or PC ,is a register that holds the address of the current instruction. It feeds into the PC predictor, which then sends the Program Counter (PC) to the Instruction Cache to read the current instruction. At the same time PC predictor predicts the address of the next instruction by incrementing the PC by 4(all the instructions were 4 byte long) [15].

Instruction Decode:

Once fetched from the instruction cache, the instruction bits werw shifted down the pipeline, so that simple combinational logic in each pipeline stage could produce the control signals for the data path directly from the instruction bits. As a result, very little decoding is done in the stage traditionally called the decoding stage [15].

Execute:

The Execute stage is where the actual computation occurs. Typically this stage consists of an Arithmetic and Logic Unit, and also a bit shifter. It may also include a multiple cycle multiplier and divider. The Arithmetic and Logic Unit is responsible for performing Boolean operations (and, or, not, nand, nor, xor, xnor) and also for performing integer addition and subtraction. Besides the result, the ALU typically provides status bits such as whether or not the result was 0, or if an overflow occurred [15].

Memory Excess:

If data memory needs to be accessed, it is done so in this stage. During this stage, single cycle latency instructions simply have their results forwarded to the next stage. This forwarding ensures that both single and two cycle instructions always write their results in the same stage of the pipeline, so that just one write port to the register file can be used, and it is always available [15].

Write back:

During this stage, both single cycle and two cycle instructions write their results into the register file [15].

Advantages and Disadvantages of Pipelining:

The main advantage of pipelining is that it keeps all portions of the processor occupied and increases the amount of useful work the processor can do in a given time. Reducing the processor's cycle time usually increases the throughput of instructions. By making each dependent step simpler, pipelining can enable complex operations more economically than adding complex circuitry, such as for numerical calculations [1].

There are two disadvantages of pipeline architecture. The first is complexity. The second is the inability to continuously run the pipeline at full speed, i.e. the pipeline stalls. There are many reasons as to why pipeline cannot run at full speed. There are phenomena called pipeline hazards, which disrupt the smooth execution of the pipeline. The resulting delays in the pipeline flow are called bubbles. These issues can and are successfully dealt with. But detecting and avoiding the hazards leads to a considerable increase in hardware complexity. The control paths controlling the gating between stages can obtain more circuit levels than the data paths being controlled [1].

2.2 Vector Processing:

Many scientific Problems require arithmetic operation on large array of numbers. These numbers are formulated as vector and matrices of floating point numbers. A vector is an ordered set of one dimensional array of data items. A vector V of length n is represented as a row vector by V=(V1,V2,V3….Vn). It may be represented as a column vector if the data items are listed in column.

A Conventional Sequential computer is capable of processing operands one at a time. Consequently operations on vectors must be broken down into single computation with subscripted variables. The element V_i of vector v is written as V(I) and I refers to the memory address or register where the data is stored [8].

2.3 Array Processors:

An array processor is a processor that performs computations on large array of data. The term is used to refer to two different types of processors. An *attached array processor* as shown in Fig.4 is an auxiliary processor attached to general purpose computer. It is intended to improve the performance of the host computer in specific numerical computation task. An SIMD array processor shown in Fig.5that has a single instruction multiple data organization. It manipulates vector instructions by means of multiple functional unit responding to a common instruction [8].

Attached Array Processor:

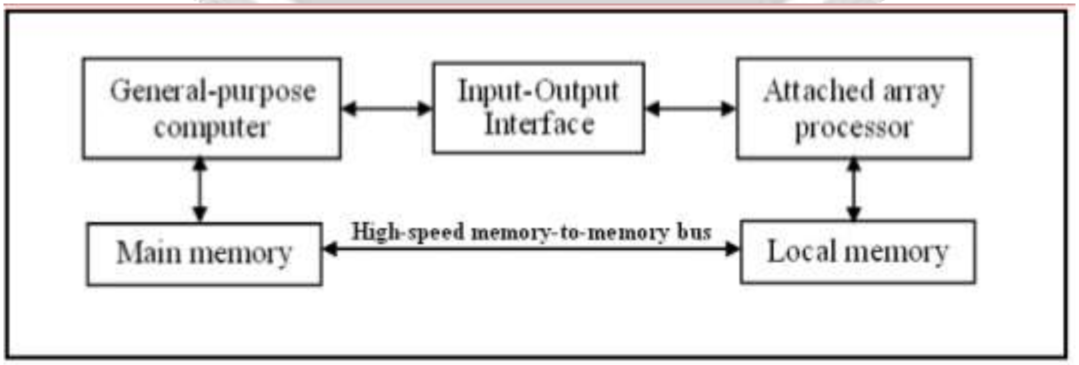


Fig.4 Attached Array Processor [10]

SIMD Array Processor:

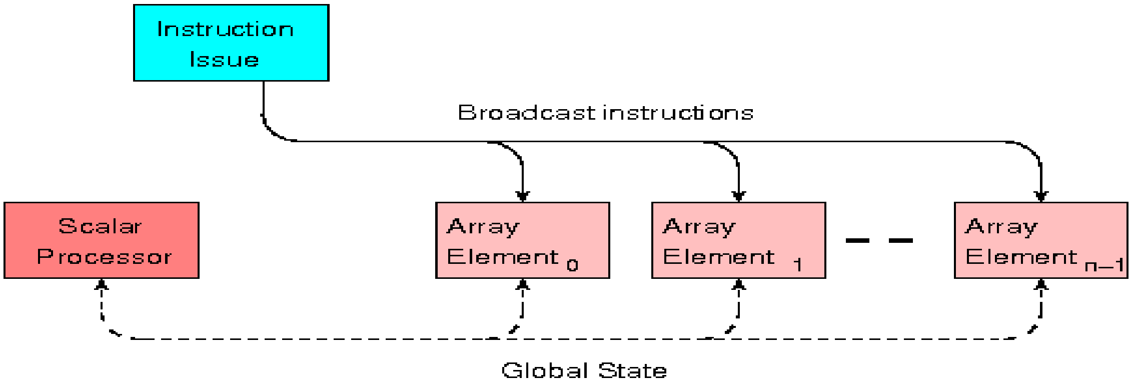


Fig.5 Classical SMID array Architecture [9]

3. MIPS

MIPS (originally an acronym for **Microprocessor without Interlocked Pipeline Stages**) is a reduced instruction set computer (RISC) instruction set architecture (ISA) developed by MIPS Technologies (formerly MIPS Computer Systems, Inc.). The early MIPS architectures were 32-bit, with 64-bit versions added later. Multiple revisions of the MIPS instruction set exist, including MIPS I, MIPS II, MIPS III, MIPS IV, MIPS V, MIPS32, and MIPS64. The current revisions are MIPS32 (for 32-bit implementations) and MIPS64 (for 64-bit implementations) [11].

3.1 Comparison of different MIPS architectures:

NO.	MIPS name	Version	Year	Developed by	Sub Versions
1	MIPS 1	R2000	1986	MIPS computer system.	**R 2000**: Execute all non floating point instructions, with simple and short pipeline, control External code and data caches with direct indexing. **R2010**: include Floating Point registers, data path, longer simple pipeline. **R2020** : Write in to main memory take tens of cycles To fully complete but R2020 chips queued and completed up to 4 pending writes to main memory in one clock cycle.
2	MIPS1	R3000	1988	MIPS computer system.	**R 3000**: operated at clock speed 20, 25, 33.33 MHz, Includes 4 co processors control processor, External R 3010 Numeric Processor, External Processor 1, External processor 2. **R 3010**: operated at clock speed 20, 25, 40 MHz, On chip Cache controller. **PR 3400**: operated at clock speed 20, 25, 40 MHz, CPU can access both data cache and instruction cache in same CPU cycle.

3	MIPS2	R6000	1989	MIPS computer system.	**R 6010**: Floating Point Unit. **R6020**: System Bus controller using ECL technologies and fabricated by Bipolar integrated technologies.
4	MIPS3	R4000	1991	MIPS computer system.	**R4000 SC**: Supports secondary cache but multiprocessor capability is not available. **R4000 MC**: Secondary Cache with multiprocessor capabilities.
5	MIPS4	R8000	1994	MIPS computer system.	**R8000**: Super scalar, capable of issuing up to four instructions per cycle, it has a five stage pipeline. **R8010**: Includes Floating Point unit , Two tag RAM, and streaming cache.
6	MIPS5	-	1997	MIPS computer system.	Improve the performance of 3D graphics applications MIPS V added a new data type, the pair-single (PS), which consisted of two single-precision (32-bit) floating-point numbers stored in the existing 64-bit floating-point registers.
7	MIPS 32/64	Release 1	1999	MIPS computer system.	The first release of MIPS32, based on the MIPS II instruction set, added conditional moves, prefetch instructions, and other features from the R4000 and R5000 families of 64-bit processors. The MUL and MADD multiply-add instructions.
8	MIPS 32/64	Release 2 /3/5	-	MIPS computer system.	Include Repeated Features of previous Release with minor changes
9	MIPS 32/64	Release 4	SKIPPED	MIPS computer system.	skipped due to the number 4 being perceived as unlucky in the Asia Pacific Rim market
10	MIPS 32/64	Release 6	2014	MIPS computer system.	Unconditional branches (BC) & branch-and-link (BALC) with a 26-bit offset. Full set of branch-and-link which compare a register against zero (e.g. BGTZALC). Index jump instructions with no delay slot designed to support large absolute addresses. Instructions to load 16-bit immediates at bit position 16, 32 or 48, allowing to easily generate large constants. PC-relative load instructions, as well as address generation with large (PC-relative) offsets. Multiply & divide instructions redefined so that they use a single register for their result).

Table 1. Comparison of different MIPS architectures [12]

3.2 Interlocking:

An interlock is a feature that makes the state of two mechanisms or functions mutually dependent. Prevent undesired states in a FSM. The hardware to detect a data hazard and stall the pipeline until the hazard is cleared is **called a pipeline interlock.**

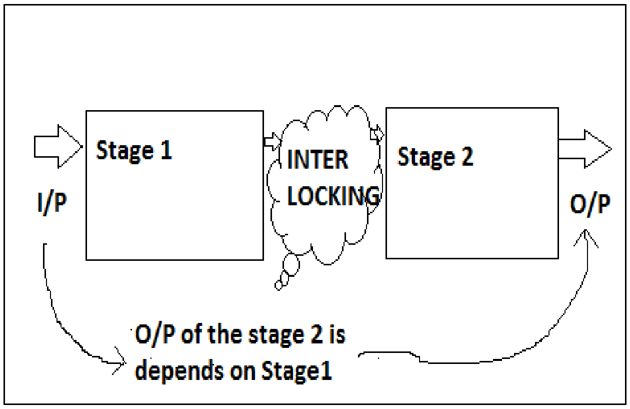


Fig.6 Interlocking

Major barrier to pipelining was that instructions, like division, take longer to complete. The CPU therefore has to wait before passing the next instruction into the pipeline. One solution to this is to use a series of interlocks that allows stages to indicate that they are busy, pausing the other stages. Interlocks are the major performance barrier since they had to communicate to all other modules in the CPU. It takes more time and appeared to be limit the clock speed. Thereby removing any need for interlocking and permitting single cycle throughput.

4. CONCLUSIONS

In this review various basic computer architecture and parallel processing techniques has been discussed for implementing fast CPU. Among all of them Pipelined processing approach is used for MIPS Implementation. Pipelined MIPS uses different pipelined styles like three stage, four stage and five stage structure for high performance implementation.

5. REFERENCES

[1]. “Iro Pantazi – Mytarelli “The history and use of pipelining computer architecture: MIPS pipelining implementation. New York Institute of Technology Old Westbury, New York, 1568.
[2]. “ Hadeel Sh. Mahmood ,Safaa S. Omran “ Pipelined MIPS Processor with Cache Controller using VHDL Implementation for Educational Purposes. The First International Conference of Electrical, Communication, Computer, Power and Control Engineering ICECCPCE'13/December17-18, 2013 .
[3] “Mr. S. P. Ritpurkar, Prof. M. N. Thakare, Prof. G. D. Korde” Synthesis and Simulation of a 32Bit MIPS RISC Processor using VHDL, IEEE International Conference on Advances in Engineering & Technology Research (ICAETR - 2014).
[4] ”Mohit N. Topiwala, N. Saraswathi ” Implementation of a 32-bit MIPS Based RISC Processor using Cadence, 2014 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT).
[5] “Nikunj Hinsu , Digvijaysinh Suryavanshi” A Prototype Design For Microprocessor Based on Verilog HDL, International conference for convergence of technology -2014.
[6] “Mr. S. P. Ritpurkar, Prof. M. N. Thakare, Prof. G. D. Korde” Design and Simulation of 32-Bit RISC ArchitectureBased on MIPS using VHDL, 2015 International Conference on Advanced Computing and Communication Systems (ICACCS -2015)
[7] Computer Architecture: A Quantitative Approach Fourth Edition, by John L Hennessy, David A Patterson, Morgan kaughmann publishers.
[8] Computer System Architecture by Morris Mano, Pearson publication.
[9] http://homepages.inf.ed.ac.uk/cgi/rni/comp-arch.pl?ArrayP/intro.html,ArrayP/intro-f.html,ArrayP/menu.html

[10]https://www.google.co.in/search?q=attached+array+processors&safe=active&source=lnms&tbm=isch&sa=X&ved=0ahUKEwj48YuZi9XSAhULrI8KHTZkCQ0Q_AUIBigB&biw=1366&bih=589&dpr=1#imgrc=WjqyK0IHkgU5EM:
[11] https://en.wikipedia.org/wiki/MIPS_instruction_set
[12] https://en.wikipedia.org/wiki/MIPS_instruction_set#MIPS32.2FMIPS64_Release_2
[13]https://www.google.co.in/search?q=three+stage+pipelining&safe=active&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiijev409fSAhVDMo8KHbeqBwsQ_AUIBigB&biw=1366&bih=589#imgrc=25azP2esoDmuPM:
[14]https://www.google.co.in/search?q=three+stage+pipelining&safe=active&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiijev409fSAhVDMo8KHbeqBwsQ_AUIBigB&biw=1366&bih=589#safe=active&tbm=isch&q=five+stage+pipelining&*&imgrc=7dt-_WQyjc172M:
[15] https://en.wikipedia.org/wiki/Classic_RISC_pipeline