MIT World Peace University

Digital Electronics and Computer Architecture
Second Year B. Tech, Semester 1

Processor Structure, function and register organization And 1 Numerical.

Practical Report

Prepared By

Krishnaraj Thadesar, PA20
Atharva Yadav, PA24
Anuj Choudhary, PA26
Harshal Thoke, PA77
Yukta Hande, PA99
Shreny Jain, PA66

November 26, 2022

Contents

1 Problem Statement

Illustrate the Processor Structure, function and register organization using, the Mind Mapping concept.

2 Numerical

Consider a computer system with byte addressable primary memory of size 2^32 bytes. It has direct mapped caches of size $32KB(1KB = 2^10bytes)$, and each cache block is 64 bytes. What is the size of the tag field bit?

3 Theory

3.1 Processor Requirements

1. Requirements placed on the processor:
2. Fetch instruction: reads an instruction from memory;
3. Interpret instruction: determines what action to perform;
4. Fetch data: if necessary read data from memory or an I/O module.
5. Process data: If necessary perform arithmetic / logical operation on data.
6. Write data: If necessary write data to memory or an I/O module.

To do these things the processor needs to:

1. • Store some data temporarily • Remember the location of the next instruction;

While an instruction is being executed: In other words, the processor needs a small internal memory, called **Registers.**

3.2 Major Parts of a CPU

1. Arithmetic and Logic Unit (ALU): Performs computation or processing of data
2. Control Unit: Moves data and instructions in and out of the processor. Also controls the operation of the ALU;
3. Registers: Used as internal memory;
4. System Bus: Acting as a pathway between processor, memory and I/O module(s);

3.3 Register Organization

1. Registers in the processor perform two roles:
2. User-visible registers:
3. Used as internal memory by the assembly language programmer;
4. Control registers: Used to control the operation of the processor;
5. Status Register: Used to check the status of the processor or ALU;

3.4 User Visible Register

1. May be referenced by the programmer, categorized into:
2. General purpose
3. Data
4. Address
5. Condition codes

3.5 General Purpose Registers

1. Can be assigned to a variety of functions by the programmer:
2. Memory reference and backup;
3. Register reference and backup;
4. Data reference and backup;
5. These are the ones you use in the laboratory

3.6 Data Registers

1. May be used only to hold data and cannot hold addresses:
2. Must be able to hold values of most data types;
3. Some machines allow two contiguous registers to be used:
4. For holding double-length values

3.7 Flags or Condition Code Bits

1. Condition code bits are collected into one or more control register
2. Interruption results in all user-visible registers being saved;
3. These are then restored on return;
4. Allows each subroutine to use the user-visible registers independently;

3.8 Examples of Control Registers

1. Program counter (PC): Contains instruction address to be fetched;
2. Instruction Register (IR): Contains last instruction fetched;
3. Memory address register (MAR): Contains memory location address;
4. Memory buffer register (MBR): Contains:
5. Word of data to be written to memory;
6. Word of data read from memory.

3.9 PSW Register

1. Many processors include a program status word (PSW) register:
2. Contains condition codes plus other status information
3. Common fields or flags include the following:
4. Sign: Sign bit of the result of the last arithmetic operation;
5. Zero: when the result is 0;
6. Carry: Set if an operation resulted in a carry/borrow bit;
7. Equal: Set if a logical compare result is equality.
8. Overflow: Used to indicate arithmetic overflow.
9. Interrupt Enable/Disable: Used to enable or disable interrupts.

4 Numerical

Concept
Memory address = 2^{32} bytes
Data Cache = 32 KB = 2^{15} B
Block Size = 64 bytes = 2^6 B
Concept:
In a direct Mapped, main memory can be represented as

Tag	Lines	Block offset

32 = tag + lines + Block offset (In bits)
Calculation:
Assume byte addressable:
number of lines in cache memory

$$= \frac{2^{15}}{2^6} = 2^9$$

$$32 = \text{tag} + 9 + 6$$

$\therefore$ tag = 17 bits

5 Prodecure

1. Draw Flowcharts, Mind maps and Diagrams on Paper
2. Try to draw them on Online collaborative websites like draw.io and excalidraw.com
3. Solve the Numerical
4. Combine pdfs into 1.

6 Conclusion

Thus learnt more about processor structure, architecture, and Register Types. Also learnt to solve numericals based on Computer architecture.