Memory Segmentation

Angel Deborah S

SSN College of Engineering

angeldeborahs@ssn.edu.in

December 15, 2017

Overview

1. Introduction

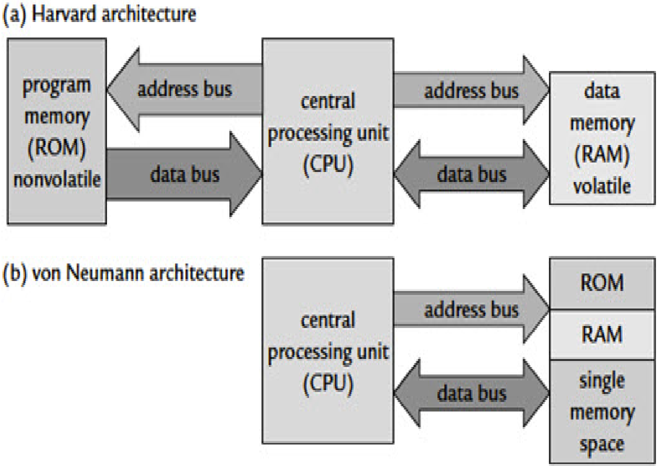
2. Von Newman architecture and Harvard architecture

3. Segmented Memory

Introduction

- Von Newman architecture and Harvard architecture
- Program Memory and Data Memory
- Need for Segmentation
  - To implement Harvard architecture
  - Easy to debug
  - Same Interfacing ICs can be used
  - To avoid overlap of stack with normal memory

Von Newman architecture and Harvard architecture


(a) Harvard architecture
program memory (ROM) nonvolatile
address bus
data bus
central processing unit (CPU)
address bus
data bus
data memory (RAM) volatile
(b) von Neumann architecture
central processing unit (CPU)
address bus
data bus
ROM
RAM
single memory space

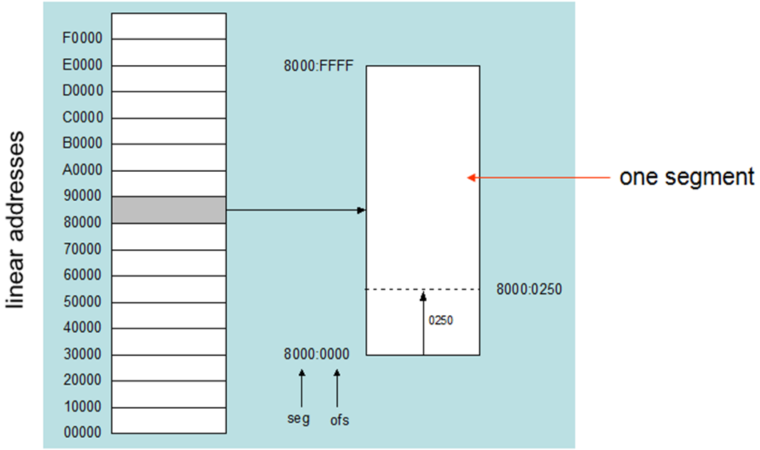
Von Newman architecture

- It is named after the mathematician and early computer scientist John Von Neumann.
- The computer has single storage system(memory) for storing data as well as program to be executed.
- Processor needs two clock cycles to complete an instruction.Pipelining the instructions is not possible with this architecture.
- In the first clock cycle the processor gets the instruction from memory and decodes it. In the next clock cycle the required data is taken from memory. For each instruction this cycle repeats and hence needs two cycles to complete an instruction.
- This is a relatively older architecture and was replaced by Harvard architecture.

Harvard architecture

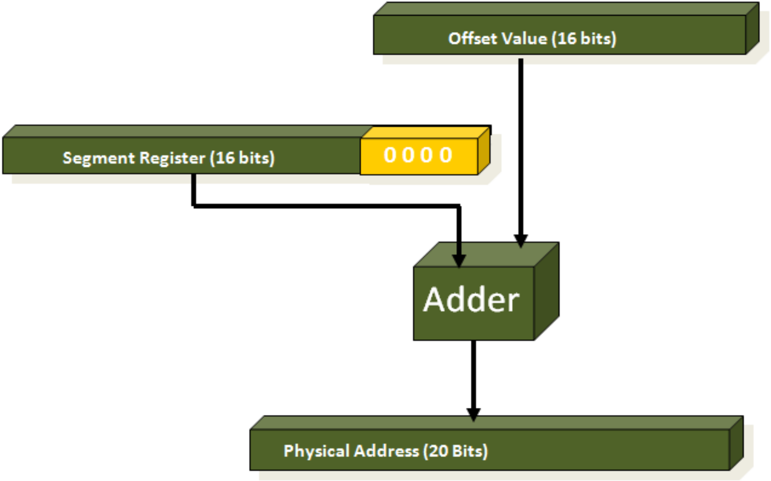
- The name is originated from "Harvard Mark I" a relay based old computer.
- The computer has two separate memories for storing data and program.
- Processor can complete an instruction in one cycle if appropriate pipelining strategies are implemented.
- In the first stage of pipeline the instruction to be executed can be taken from program memory.In the second stage of pipeline data is taken from the data memory using the decoded instruction or address.
- Most of the modern computing architectures are based on Harvard architecture.But the number of stages in the pipeline varies from system to system.

Segmented Memory



Memory Address Generation

- The BIU has a dedicated adder for determining physical memory addresses.



Segment:Offset Address

- Logical Address is specified as segment:offset
- Physical address is obtained by shifting the segment address 4 bits to the left and adding the offset address.
- Thus the physical address of the logical address A4FB:4872 is:

$$\begin{array}{r} \text{A4FB0} \\ +\ 4872 \\ \hline \text{A9822} \end{array}$$

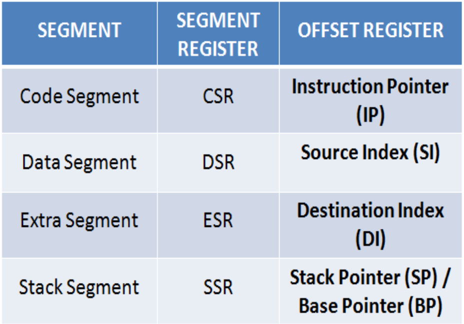
Segments, Segment Registers and Offset Registers

- Segment Size $= 64$KB
- Maximum number of segments possible $= 14$
- Logical Address 16 bits
- Physical Address 20 bits
- 2 Logical Addresses for each Segments.
    - Base Address (16 bits)
    - Offset Address (16 bits)
- Segment registers are used to store the Base address of the segment.

Segments, Segment Registers and Offset Registers

4 Segments in 8086

- Code Segment (CS)
- Data Segment (DS)
- Stack Segment (SS)
- Extra Segment (ES)



SEGMENT	SEGMENT REGISTER	OFFSET REGISTER
Code Segment	CSR	Instruction Pointer (IP)
Data Segment	DSR	Source Index (SI)
Extra Segment	ESR	Destination Index (DI)
Stack Segment	SSR	Stack Pointer (SP) / Base Pointer (BP)