Embedded System Design and Programming

17BM663

Module -3
Processor and Memory Organization

Syllabus

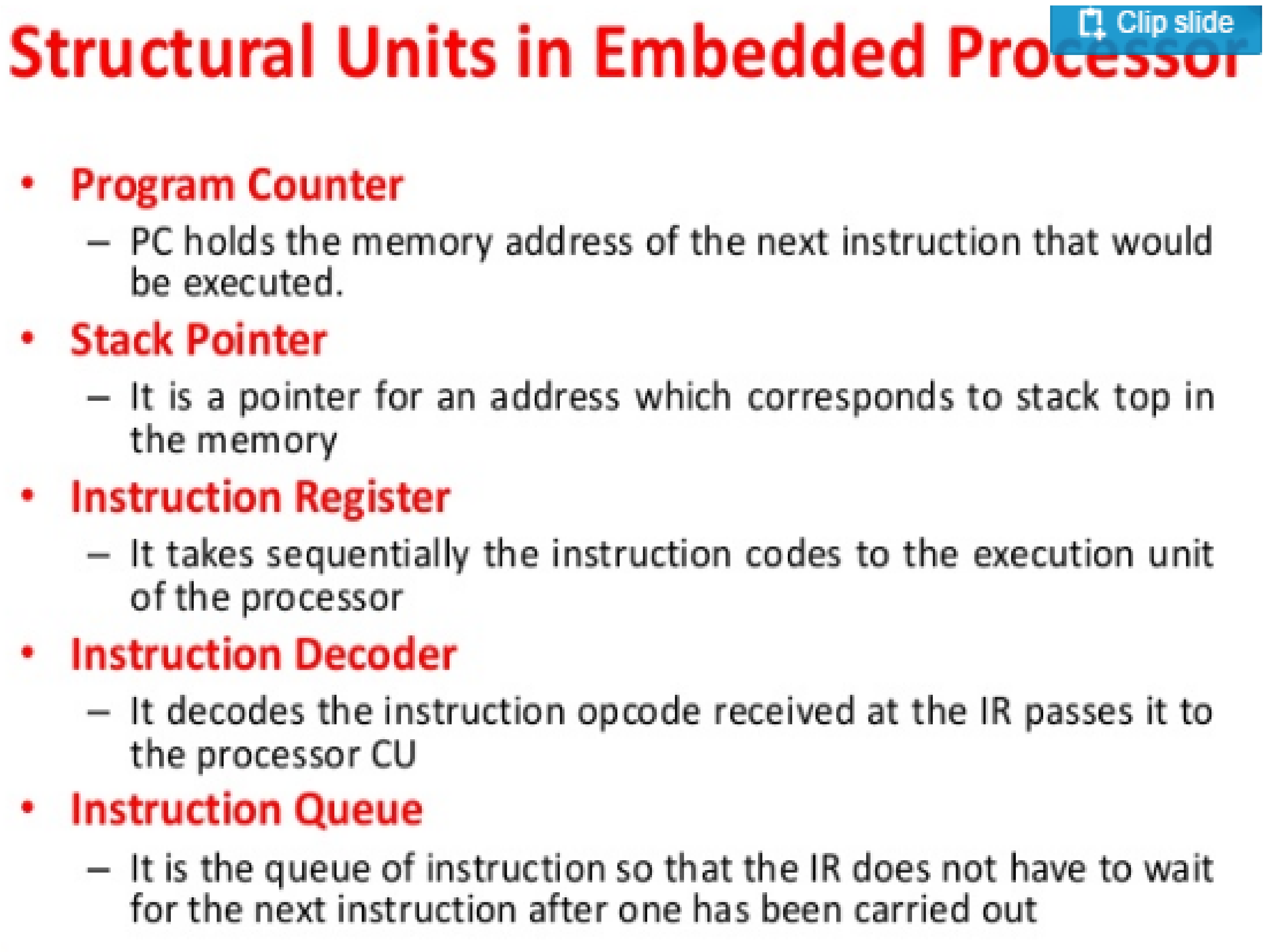
- Structural Units in a Processor
- Memory Devices
- Memory selection for an embedded system
- Processor selection
- Direct Memory Access
- DMA controllers.

Structural Units in a Processor

- General structural units in a processor architecture and their functions:

Structural Units in Embedded Processo

- **Internal Buses:**
  - It internally connects all the structural units inside the processor.
  - Its width can be 8,16,32 or 64 bits
- **Address Bus**
  - It is the external bus that carries the address from the MAR to the memory as well as the IO devices and the other units of the system
- **Data Bus**
  - It is an external bus that carries the data from or to the address
- **Control Bus**
  - It is an external bus to carry control signal between the processor and memory devices
- **Bus Interface Unit**
  - It is the interface unit between the processor's internal units with the external buses
- **MAR:**
  - Memory Address Register
  - It holds the address of the byte or word to be fetched from external memories
- **MDR:**
  - It holds the byte or word fetched from/to external memory or I/O address



Structural Units in Embedded Processor

- **Program Counter**
  - PC holds the memory address of the next instruction that would be executed.
- **Stack Pointer**
  - It is a pointer for an address which corresponds to stack top in the memory
- **Instruction Register**
  - It takes sequentially the instruction codes to the execution unit of the processor
- **Instruction Decoder**
  - It decodes the instruction opcode received at the IR passes it to the processor CU
- **Instruction Queue**
  - It is the queue of instruction so that the IR does not have to wait for the next instruction after one has been carried out

Structural Units in Embedded Processor

- **Control Unit**
  - It controls all the bus activities and unit functions needed for processing
- **Memory management Unit (MMU)**
  - It manages the memory such that instruction and data are readily available for processing
- **Application Register Set**
  - It is set of on-chip registers used during processing the instruction of the **application program of the user**

Structural units in an advanced processor architecture

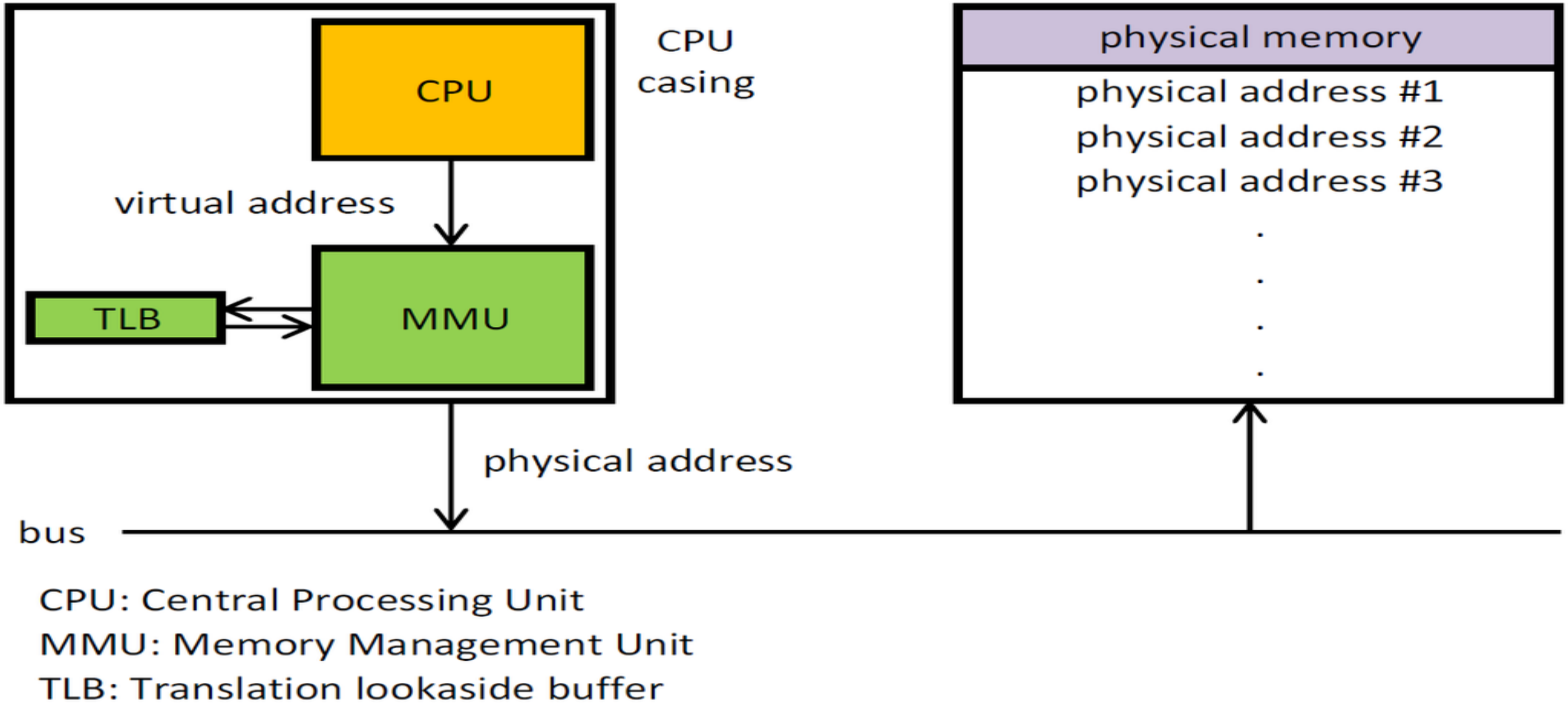
- **Instruction-level parallelism** (**ILP**) is a measure of how many of the instructions in a computer program can be executed simultaneously.
- An **instruction queue** is a structure into which the processor fetches instructions. A separate scheduler usually identifies and dispatches from this **queue instructions** that are ready to be executed. i.e instruction register (IR) does not wait for the next

- Instruction cache (I-Cache) : it sequentially stores the instructions in FIFO mode. It lets the processor executes instructions at greater speed.
- Data cache (D-cache): It stores the perfected data from external memory. A data cache generally holds the key(addresses) and value(word) together at a location.
- Branch target cache (BT cache): It keeps the next instructions ready for execution when jump, loop or call is encountered.

- Prefetch control units (PFCH) : This unit controls the fetching of data into I-cache and D-cache in advance from the memory units. The instructions and data are delivered when needed by the processors execution units. Prefetching unit improves performance by fetching instructions and data in advance for processing .

- Memory management unit (MMU):

It is responsible for managing of memories such that the instructions and data are readily available for processing.



- System register set (SRS): set of registers used for preprocessing the instructions.
- Floating point processing unit (FLPU) : unit which is required for mathematical functions.
- Floating point register set : register sets used to store floating point numbers.
- Multiply and accumulate unit (MAC): unit responsible for multiplying series of coefficients and accumulating them.
- Atomic operation unit (AOU): it lets a user instructions when broken into processor instructions called atomic operations , finish before an interrupt of processor occurs.

Processor selection

- A designer must take into account following processor specific features:

❑ Processing speed : capable of executing more instructions per second

❑High computing performance is seen when there is: pipeline and superscalar architecture, prefetch cache unit etc

❑Processor having auto shutdown feature

- Processor having burst mode accesses to external memory fast , reads fast and writes fast
- Presence of Atomic operation unit

1. *Case 1:* Systems in which processor instruction cycle time ~ 1 μs and on-chip devices and memory can suffice. Examples are automatic chocolate vending machine, 56 kbps modem, robots, data acquisition systems like an ECG recorder or weather recorder or multipoint temperature and pressure recorder and real-time robotic controller.
2. *Case 2:* Systems in which processor instruction cycle time ~ 10 to 40 ns required, on-chip devices and memory do not suffice and medium processor performance is required. Examples are 2 Mbps router, image processing, voicedata acquisition, voice compression, video decompression, adaptive cruise control system with string stability and network gateway.
3. *Case 3:* Systems in which instruction cycle times of 5 to 10 ns required and high MIPS or MFLOPS performance is needed. Examples are multiport 100 Mbps network transceiver, fast 100 Mbps switches, routers, multichannel fast encryptions and decryptions systems.
4. *Case 4:* Systems in which instruction cycle time of even 1-ns does not suffice and multi-processor system is required along with use of the floating point and MAC units. Examples are voice processing, video processing, realtime audio or video processing and mobile phone systems.

Memory in an system

A simple credit–debit transaction card may require just 2 kB of memory. On the other hand, a smart card for secure transactions when embedding a Java program for cryptographic functions may require 32 kB (typical value) memory. A complex embedded system may need huge memory.

ROM :its uses forms and variants

- Rom non volatility is most important asset and is extremely useful to embed codes and data in the system.
- ROM might be masked ROM, PROM, OTP-ROM, EPROM, and EEPROM.
- masked ROM, PROM or EPROM embeds the software or application logic circuit.
- If ROM has to be programmed during runtime EEPROM or flash memory is used.

- **Mask ROM** (**MROM**) is a type of read-only memory (ROM) whose contents are programmed by the integrated circuit manufacturer (rather than by the user). The terminology *mask* comes from integrated circuit fabrication, where regions of the chip are masked off during the process of photolithography

- **EPROM (erasable programmable read-only memory)** is programmable read-only memory (programmable ROM) that can be erased and re-used. Erasure is caused by shining an intense ultraviolet light through a window that is designed into the memory chip.

- **EEPROM** (also **E^{2}PROM**) stands for **electrically erasable programmable read-only memory** and is a type of non-volatile memory used in computers, integrated in microcontrollers for smart cards and remote keyless systems, and other electronic devices to store relatively small amounts of data but allowing individual bytes to be erased and reprogrammed.

- **Flash memory** is an electronic (solid-state) non-volatile computer memory storage medium that can be electrically erased and reprogrammed.
- **PROM** : It is one type of ROM (read-only memory). The data in them are permanent and cannot be changed. PROMs are used in digital electronic devices to store permanent data, usually low level programs such as firmware or microcode. The key difference from a standard ROM is that the data is written into a ROM during manufacture, while with a PROM the data is programmed into them after manufacture.

Uses of ROM

- EEPROM is usable by erasing over one million times, and can be erased during run time itself.
  - Storing current date and time
  - Storing port status
  - Storing driving, malfunctions and failure history in automobile.
- Flash memory is usable about 10,000 times
  - Storing pictures in digital camera
  - Storing compressed voice for voice recorders
  - Storing messages and contacts in phone

- The PROM is written only once by a device programmer or the first time run

OTP ROM examples:

- Smart cards
- Storing boot programs
- ATM, DEBIT, or identity card

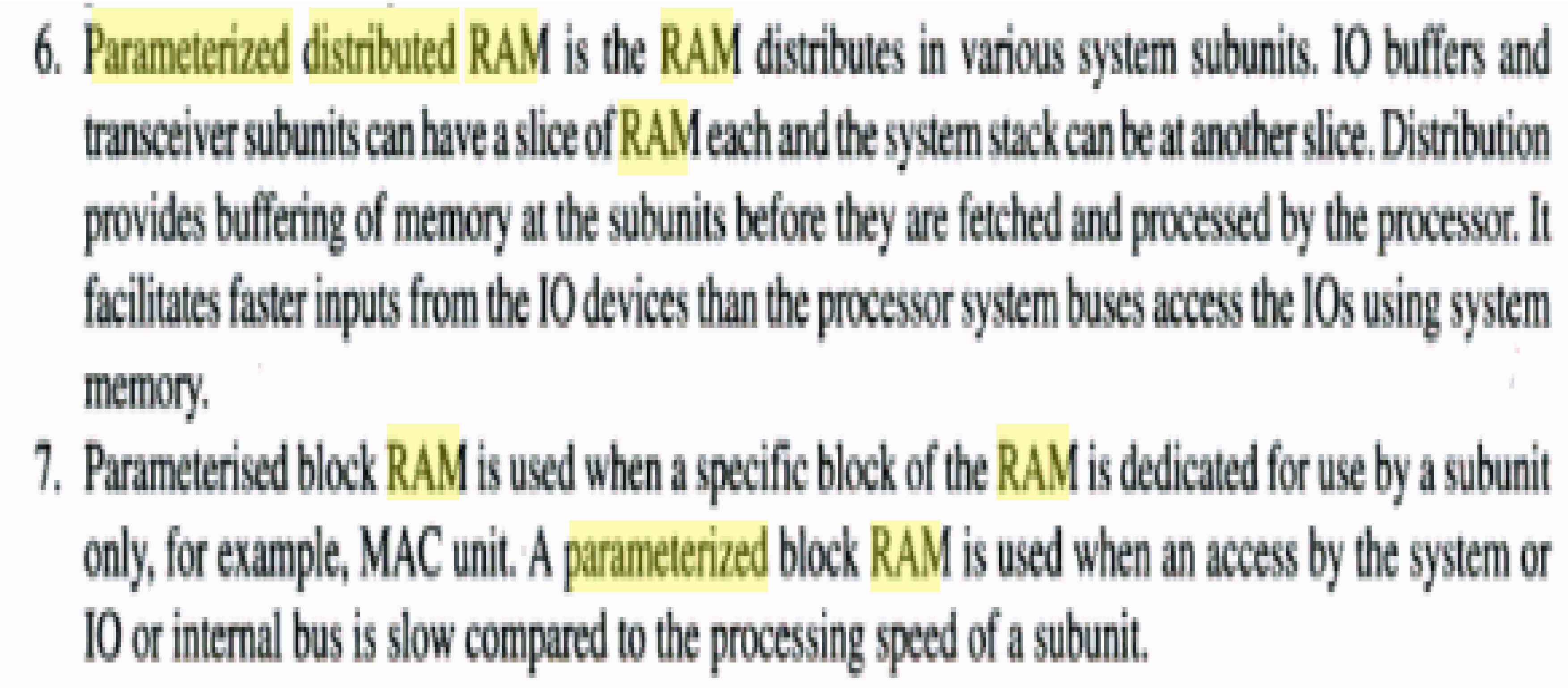
RAM

- Random Access Memory: cane be written and read . Ram is generally volatile, meaning that it does not retain the data stored in it when the systems power is turned off.
- RAM stores the variable during program run and stores the stack
- SRAM :**Static random-access memory** (**static RAM** or **SRAM**) is a type of semiconductor random-access memory (RAM) that uses bistable latching circuitry (flip-flop) to store each bit SRAM is faster and more expensive than DRAM; it is typically used for CPU cache

- **Dynamic random-access memory** (**DRAM**) is a type of random access semiconductor memory that stores each bit of data in a memory cell consisting of a tiny capacitor and a transistor, both typically based on metal-oxide-semiconductor (MOS) technology and is mostly used in high performance computers or high memory density system.

- **EDO (extended data output) RAM** is a type of random access memory (RAM) chip that improves the time to read from memory on faster microprocessors such as the Intel Pentium
- It is used in a systems with bases to the devices when operating with clock rates upto 100MHZ
- **Synchronous dynamic random-access memory** (**SDRAM**) is any dynamic random-access memory (DRAM) where the operation of its external pin interface is coordinated by an externally supplied clock signal.
- Synchronizes the read operations and keeps the next word ready while the previous one is being fetched. This is used when processor speed of 1GHZ is required.

- RDRAM: Rambus DRAM operates on basis of fetching four words in a single fetch. Performance of the system is above 1GHZ.



6. Parameterized distributed RAM is the RAM distributes in various system subunits. IO buffers and transceiver subunits can have a slice of RAM each and the system stack can be at another slice. Distribution provides buffering of memory at the subunits before they are fetched and processed by the processor. It facilitates faster inputs from the IO devices than the processor system buses access the IOs using system memory.
7. Parameterised block RAM is used when a specific block of the RAM is dedicated for use by a subunit only, for example, MAC unit. A parameterized block RAM is used when an access by the system or IO or internal bus is slow compared to the processing speed of a subunit.

Memory organization

- Most of the systems have two types of memory Random Access memory and Read only Memory
- Read Only Memory: ROM is an acronym for Read-Only Memory. It refers to computer memory chips containing permanent or semi-permanent data. Unlike RAM, ROM is non-volatile; even after you turn off your computer, the contents of ROM will remain unchanged. ROM is used to hold “bootstrap” i.e program which instructs to load its operating system every time system is turned on or reset.

- Random Access Memory: cane be written and read . Ram is generally volatile, meaning that it does not retain the data stored in it when the systems power is turned off.
- Addresses : Memory (both RAM and ROM) is divided into storage locations. Storage locations are numbered and assigned number is called as addresses.
- Random accesses model of memory: all operations take same amount of time independent of the address of the location.

- Little Endian and Big Endian in a Memory Organization
- Processor Memory Organization: Princeton Architecture
- Processor Memory Organization: Harvard Architecture

Memory selection

- When an ROM image is ready, hardware designer faces problem to choose which type of memory to be used and it should be of what size.
- The design table is built. The memory having required features and address is chosen.
- The actual requirement is known only after the coding as per the design functions and specifications .
- Memories are of 1kb, 4kb, 16kb, 32kb, 64kb, 128kb, 256kb, 512kb, 1MB. When 92kB of memory is required then device of 128kB is chosen

Read out:

I. Case study on Automatic wending machine

II. Case study of Robotic

III. Case study of DAC

IV. Case study on mobile phone system

Pg.no 118 and 119 from text

Direct memory access and DMA controller

- REFERENCE: https://www.youtube.com/watch?v=v58UFPKa8zs
- Direct memory access (DMA) is a method that allows an input/output (I/O) device to send or receive data directly to or from the main memory, bypassing the CPU to speed up memory operations. The process is managed by a chip known as a DMA controller (DMAC).

- Data transfer occurs efficiently between the I/O devices and system main memory with the least intervention of processor using DMAC
- The system address and data buses become unavailable to processor and available to the IO device that interconnects using DMAC during the data transfer.
- DMA controller requests the CPU to relinquish the control of the buses.
- CPU acknowledges external DMA controller that the buses are in high impedance state and the requesting DMA can take control of the buses. Once the DMA has taken the control of the buses it transfers the data. This transfer can take place in many ways.

**refer figure 4.13 pg no 218 from text for DMA

Three modes of data transfer:

- 1. single transfer at time (bytes)
- 2. burst at a time

(**Burst** mode is a temporary high-speed data transmission mode used to facilitate sequential data **transfer** at maximum throughput. **Burst** mode data **transfer** rate (DTR) speeds can be approximately two to five times faster than normal transmission protocols)

- 3.Bulk transfer

Uses of DMAC

Whenever DMA, request is made to DMAC , DMAC is first initialized. It is programmed for

I. Read or write
II. Data transfer
III. Total no of bytes to be transferred
IV. Starting memory address

- A multi channel DMAC provides DMA action for system memories and two (or more) devices.
- DMAC facilitates fast direct byte transfer between memory and I/O devices compared with interrupt driven data transfer
- Designers can use DMAC in sophisticated systems so that the system performance improves by separate processing of bulk or burst data transfer from and to the peropherals.

- Good feature of DMA based data transfer service has very small latency period when compared to when compared with data transfer using multiple Io interrupt source.
- DMA channels provide efficient method when the devices has to transfer large amount of data by I/O.

Question bank and assignment for the module

- Explain about the structural units in processor.
- Explain the memory selection for an Embedded system
- Explain about the processor selection for Embedded system.
- With a neat diagram, explain Direct memory Access
- Explain types of ROM and its uses for an embedded system
- Explain types of RAM and its uses for an embedded system
- Briefly explain two types of memory architecture.