**3-DIRECT MEMORY ACCESS (DMA):**

**The DMA controller** is a piece of hardware that controls one or more peripheral devices. It allows devices to transfer data to or from the system’s memory without the help of the processor.

Having peripheral devices access memory directly would allow the CPU to do other work, which would lead to improved performance, especially in the cases of large transfers.



A DMA controller has an **address register**, a **Data count register**, and a **control logic**:

•**The address register**: contains an address that specifies the memory location of the data to be transferred. The DMA controller automatically increment the address register after each word transfer, so that the next transfer will be from the next memory location.

•**The Data count register**: Holds the number of words to be transferred. The word count is decremented by one after each word transfer.

•**The control logic**: Specifies the transfer mode (number of DMA channels they support).

The following steps summarize the **DMA operations**:

**1.** DMA request.

**2.** DMA accept.

**3.** Release buses.

**4.** DMA controller initiates data transfer.

**5.** Data is moved (increasing the address in memory, and reducing the count of words to be moved).

**6.** When word count reaches zero, the DMA informs the CPU of the termination by means of an interrupt.

**7.** The CPU regains access to the memory bus.

**DMA transfer types**

**1-** **Burst mode**: The DMA controller keeps control of the bus until all the data has been transferred to( from) memory from( to) the peripheral device.

This mode of transfer is needed for fast devices where data transfer cannot be stopped until the entire transfer is done.

**2-** **Single-cycle mode** (**cycle stealing or flyby**): the DMA controller releasing the bus after each transfer of one data word. This minimizes the amount of time that the CPU is not controlling the bus. The single-cycle mode is preferred if IO devices can store very large amounts of data.

**DMA configurations:**

The DMA mechanism can be configured in a variety of ways. Some possibilities are shown in Figure below:

**DMA configuration(1)**



**(a) Single Bus, Detached DMA controller**

• Each transfer uses bus twice( IO to DMA then DMA to memory)

• CPU is suspended twice .

**DMA configuration (2)**



• Single Bus, **Integrated** DMA controller( support one IO device or more)

• Each transfer uses bus once( DMA to memory) so CPU may be suspended only once.

**DMA configuration(3)**



•Separate IO Bus

•IO Bus connect between DMA and all enabled devices

 •Each transfer uses bus once (DMA to memory) so CPU suspended once.