SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME :19IT301 COMPUTER ORGANIZATION AND ARCHITECTURE

II YEAR /III SEMESTER

Unit 5: I/O ORGANIZATION AND PARALLELISM

Topic 3: Direct Memory Access

Topic 4: Buses

Direct Memory Access(DMA)

Polling or interrupt driven I/O
considerable overhead
- Multiple program instructions
- Saving program state
- Incrementing memory addresses
- Keeping track of word count

DMA
- Transfer large blocks of data directly between an external device and memory
- Without continuous intervention by the processor

Direct Memory Access(DMA)

DMA Transfer:

Processor sends

- ✓Starting address
- ✓Number of words in block
- ✓Direction of transfer

DMA controller

- ✓Performs requested operation
- ✓Interrupts processor when done

Direct Memory Access(DMA)

A DMA controller usually has 5 registers

1. block length register
2. source address register
3. destination address register.
4. byte counter
5. a temporary data register

A user can initialize the first three registers start the DMA process

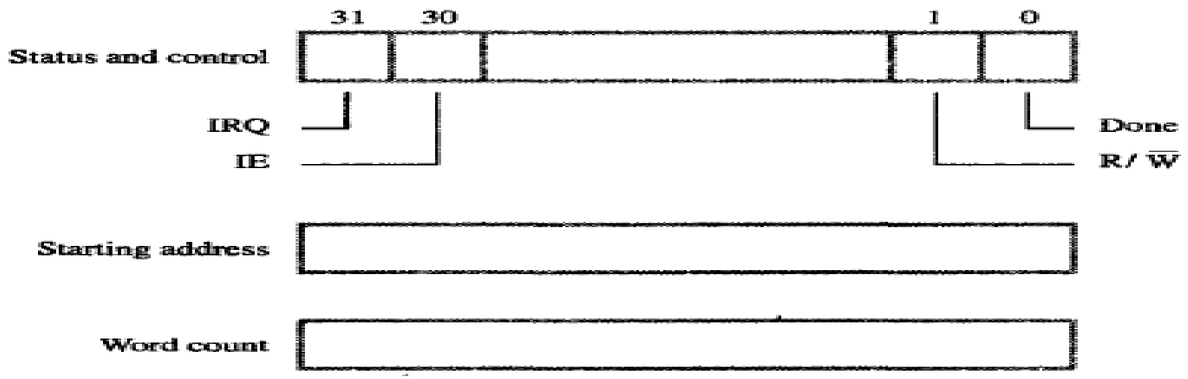
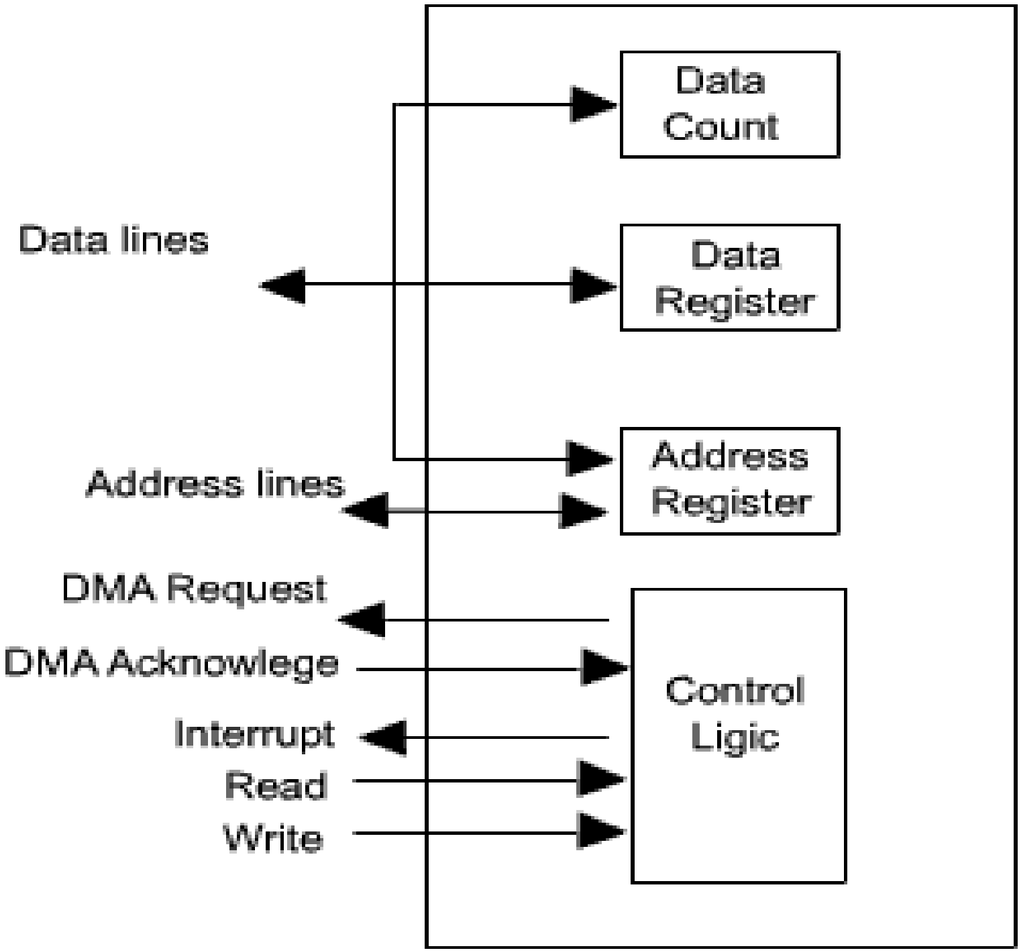


Figure 4.18 Registers in a DMA interface.

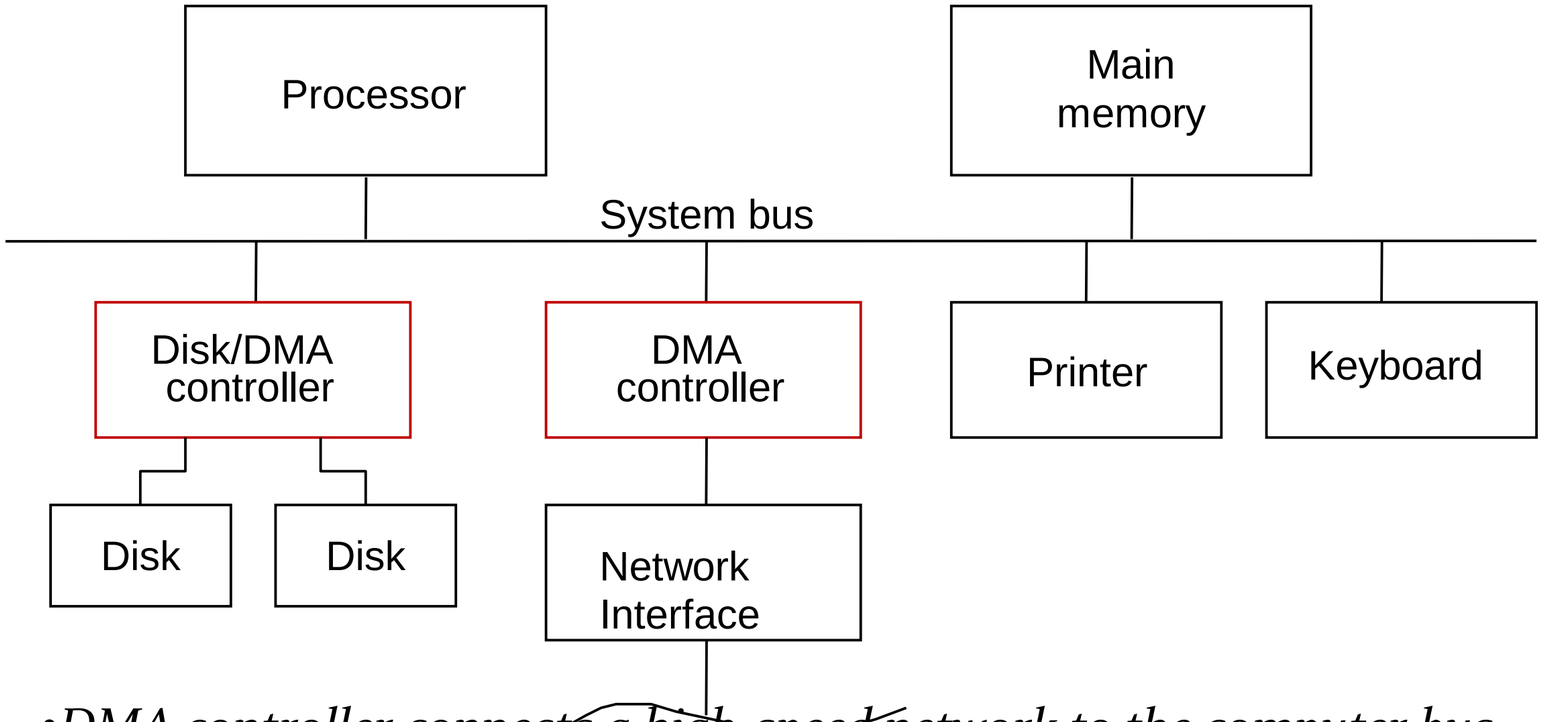


Direct Memory Access(DMA)

Steps in Memory to Memory DMA Operation

1. Microprocessor writes to DMA controller to request a memory to memory DMA operation.
2. DMA starts and requests CPU for buses (Hold Request)
3. CPU gives the buses to DMA (Hold Ack) and disconnects itself from the buses.
4. DMA puts source address and a read signal on address and control buses
5. DMA gets the data from data bus
6. DMA puts destination address, data and a write signal on the buses
7. DMA increments source and destination address registers and byte counter accordingly
8. If the byte counter is not equal to block size, go to step 4; else, DMA gives
. the buses back to CPU (withdraws Hold Request)

Direct Memory Access



- *DMA controller connects a high-speed network to the computer bus.*
- *Disk controller, which controls two disks also has DMA capability. It provides two DMA channels.*
- *It can perform two independent DMA operations, as if each disk has its own DMA controller. The registers to store the memory address, word count and status and control information are duplicated.*

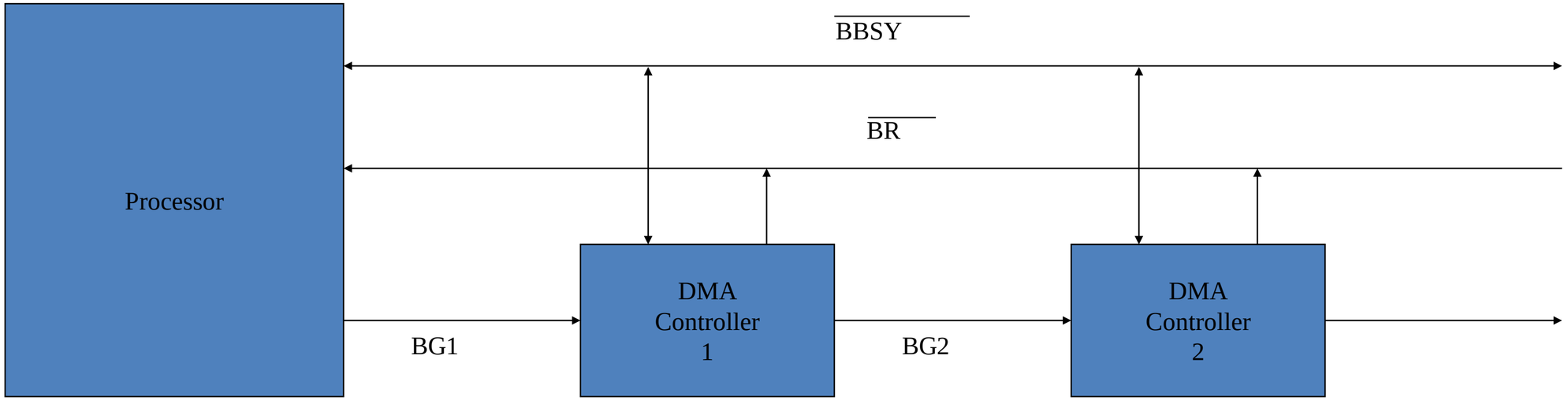
Direct Memory Access(DMA)- Bus Arbitration

- Processor and DMA controllers both need to initiate data transfers on the bus and access main memory.
- The device that is allowed to initiate transfers on the bus at any given time is called the bus master.
- When the current bus master relinquishes its status as the bus master, another device can acquire this status.
  - The process by which the next device to become the bus master is selected and bus mastership is transferred to it is called bus arbitration.
- Centralized arbitration:
  - A single bus arbiter performs the arbitration.
- Distributed arbitration:
  - All devices participate in the selection of the next bus master.

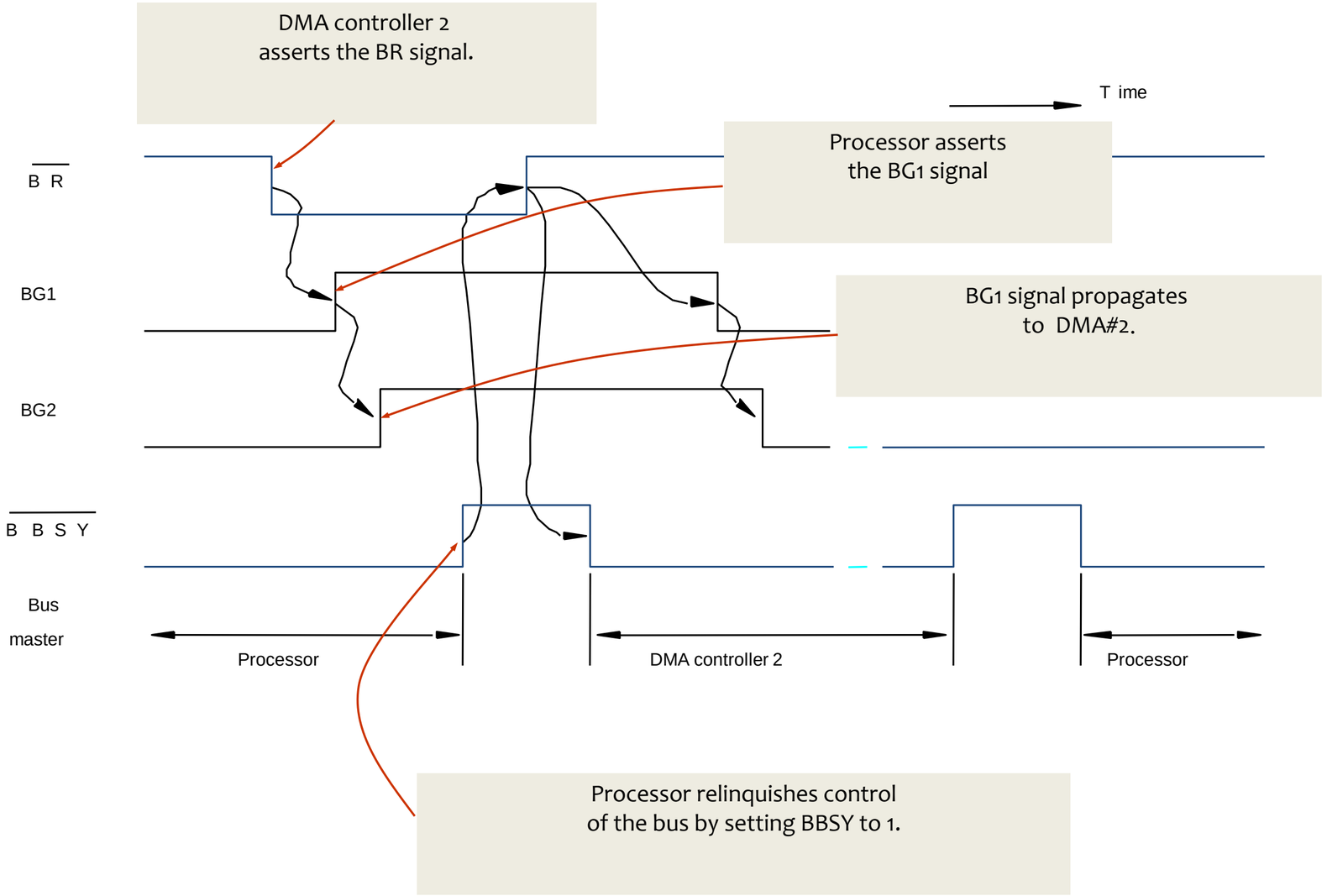
Centralized Arbitration

Daisy Chain

Bus-Request (BR)- DMA to become bus master
Bus-Grant (BG*i*)—Bus control to DMA
Bus-Busy (BBSY)— DMA is using the bus-information to other devices. *BBSY signal is 0, it indicates that the bus is busy. When BBSY becomes 1, the DMA controller which asserted BR can acquire control of the bus.*



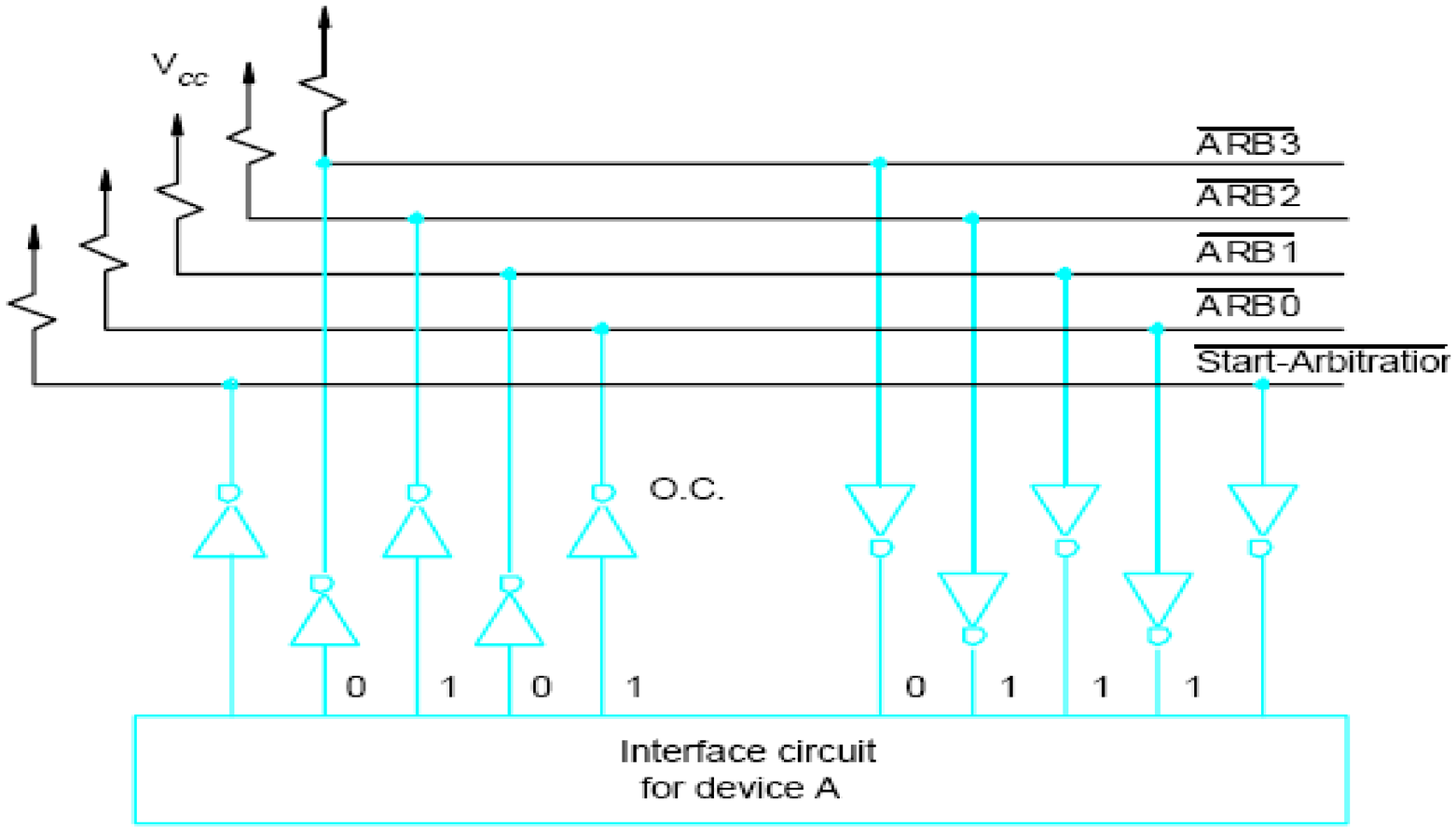
Direct Memory Access(DMA)-Centralized Arbitration



Direct Memory Access(DMA)- Distributed Arbitration

- All devices waiting to use the bus share the responsibility of carrying out the arbitration process.
  - Arbitration process does not depend on a central arbiter and hence distributed arbitration has higher reliability.
- Each device is assigned a 4-bit ID number.
- All the devices are connected using 5 lines, 4 arbitration lines to transmit the ID, and one line for the Start-Arbitration signal.
- To request the bus a device:
  - Asserts the Start-Arbitration signal.
  - Places its 4-bit ID number on the arbitration lines.
- The pattern that appears on the arbitration lines is the logical-OR of all the 4-bit device IDs placed on the arbitration lines.

Direct Memory Access(DMA)- Centralized Arbitration



Direct Memory Access(DMA)-Centralized Arbitration

Example : Assume A (ID=5) and B (ID=6) are requesting the bus

- ✓ Device A transmits the pattern 0101
- ✓Device B transmits the pattern 0110
- ✓The arbitration lines are active when low , the code seen by the two devices is 0111
- ✓ Each device compares this pattern with its own ID starting with the most significant bit if it detects a difference at any bit position it disables its drivers at that bit position and all lower-order bits

- ✓Device A detects a difference on line ARB1
- ✓Thus it disables its drivers on lines ARB1 and ARB0
- ✓The pattern on the arbitration lines changes to 0110
- ✓ B wins the contention

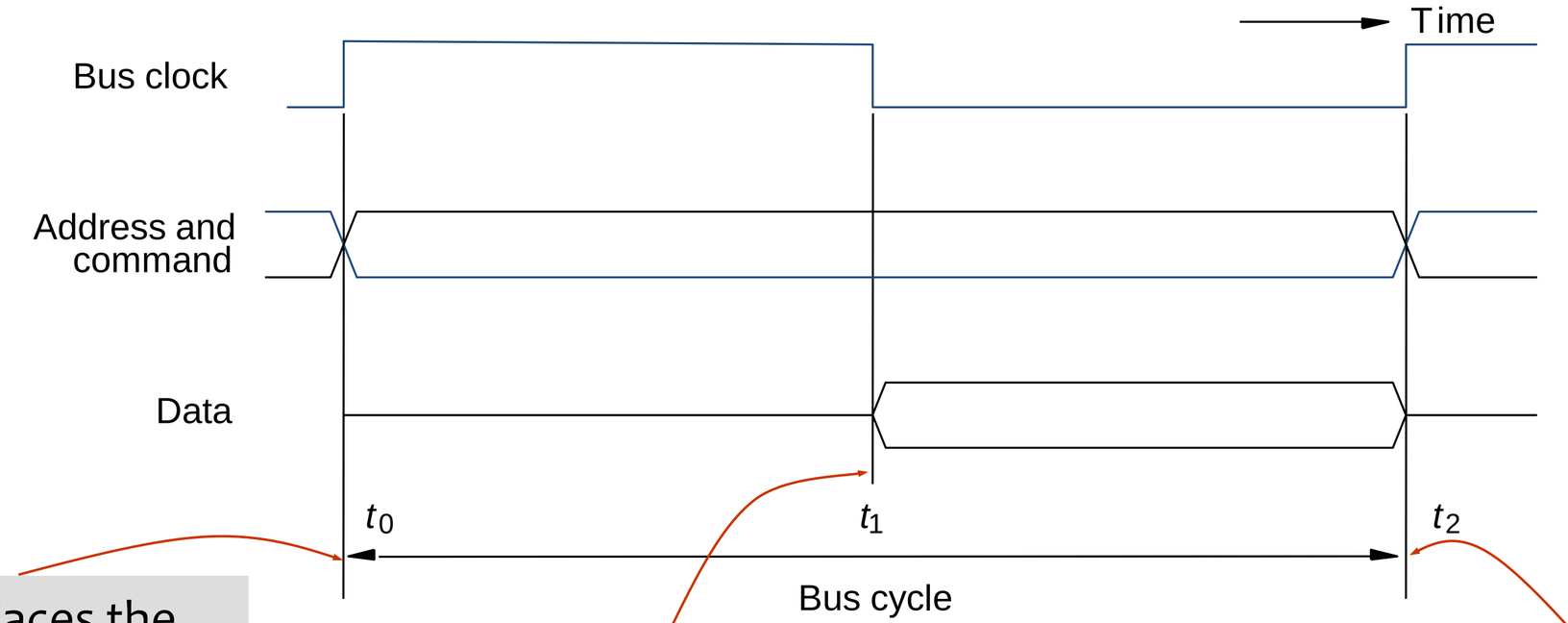
Buses

Processor, main memory, and I/O devices are interconnected by means of a bus.

Bus provides a communication path for the transfer of data.

- Bus lines may be grouped into three types:
  - Data
  - Address
  - Control
- Control signals specify:
  - Whether it is a read or a write operation.
  - Required size of the data, when several operand sizes (byte, word, long word) are possible.
  - Timing information to indicate when the processor and I/O devices may place data or receive data from the bus.
- Schemes for timing of data transfers over a bus can be classified into:
  - Synchronous,
  - Asynchronous

Synchronous bus



Master places the device address and command on the bus, and indicates that it is a Read operation.

Addressed slave places data on the data lines

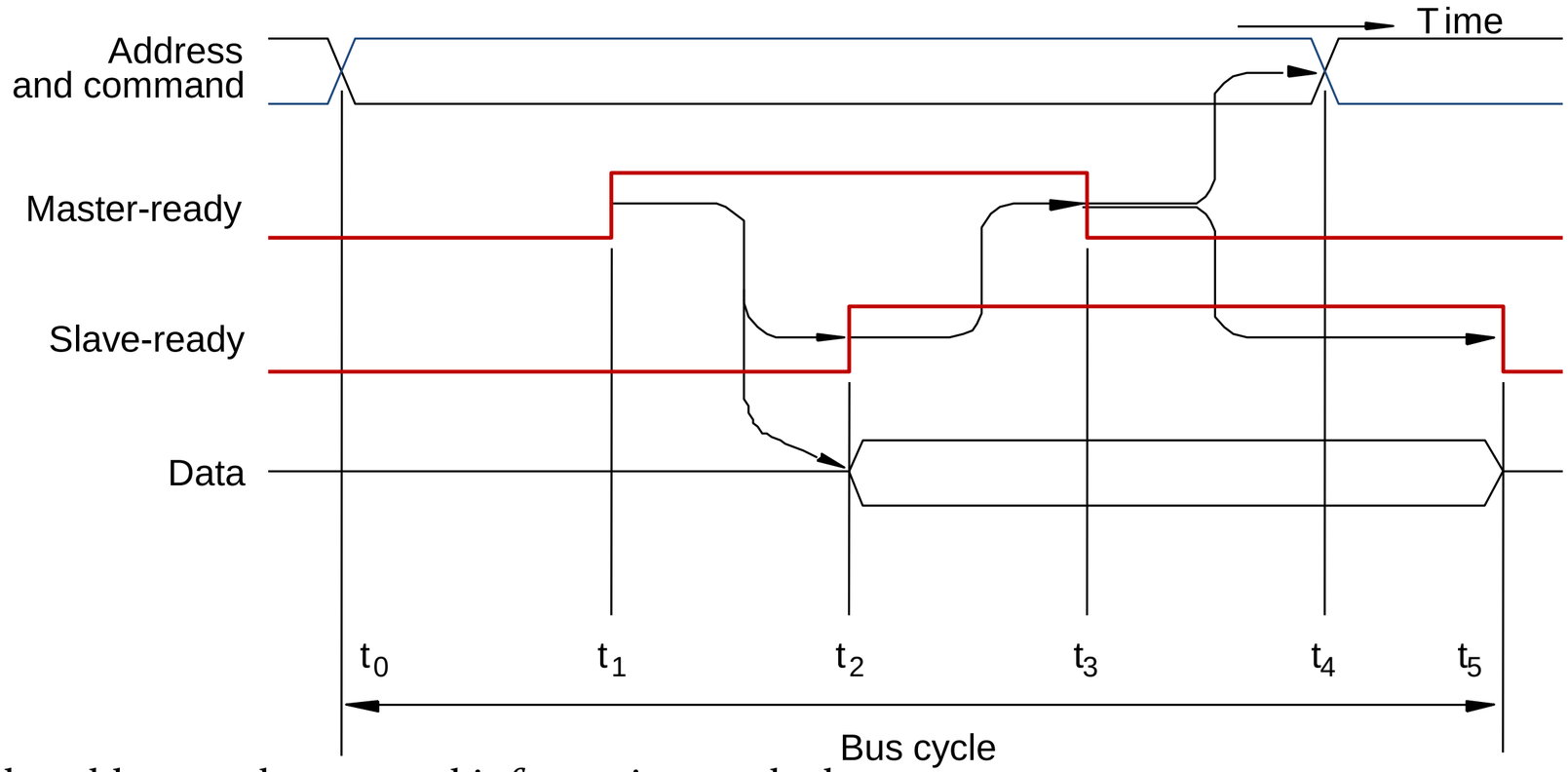
Master "strobes" the data on the data lines into its input buffer, for a Read operation.

- *In case of a Write operation, the master places the data on the bus along with the address and commands at time t_0.*
- *The slave strobes the data into its input buffer at time t_2.*

Asynchronous bus

- Data transfers on the bus is controlled by a handshake between the master and the slave.
- Common clock in the synchronous bus case is replaced by two timing control lines:
  - Master-ready,
  - Slave-ready.
- Master-ready signal is asserted by the master to indicate to the slave that it is ready to participate in a data transfer.
- Slave-ready signal is asserted by the slave in response to the master-ready from the master, and it indicates to the master that the slave is ready to participate in a data transfer.

Asynchronous bus (contd..)



t_o - Master places the address and command information on the bus.
t_1 - Master asserts the Master-ready signal. Master-ready signal is asserted at t_1 instead of t_o
t_2 - Addressed slave places the data on the bus and asserts the Slave-ready signal.
t_3 - Slave-ready signal arrives at the master.
t_4 - Master removes the address and command information.
t_5 - Slave receives the transition of the Master-ready signal from 1 to 0. It removes the data and the Slave-ready signal from the bus.

Assessment

1. What is DMA?
2. Compare Interrupt IO and DMA
3. What is synchronous bus?
4. What is asynchronous bus?

Reference

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, “Computer Organization”, McGraw-Hill, 6th Edition 2012.