Module 5- Buses

- Synchronous bus
- Asynchronous bus

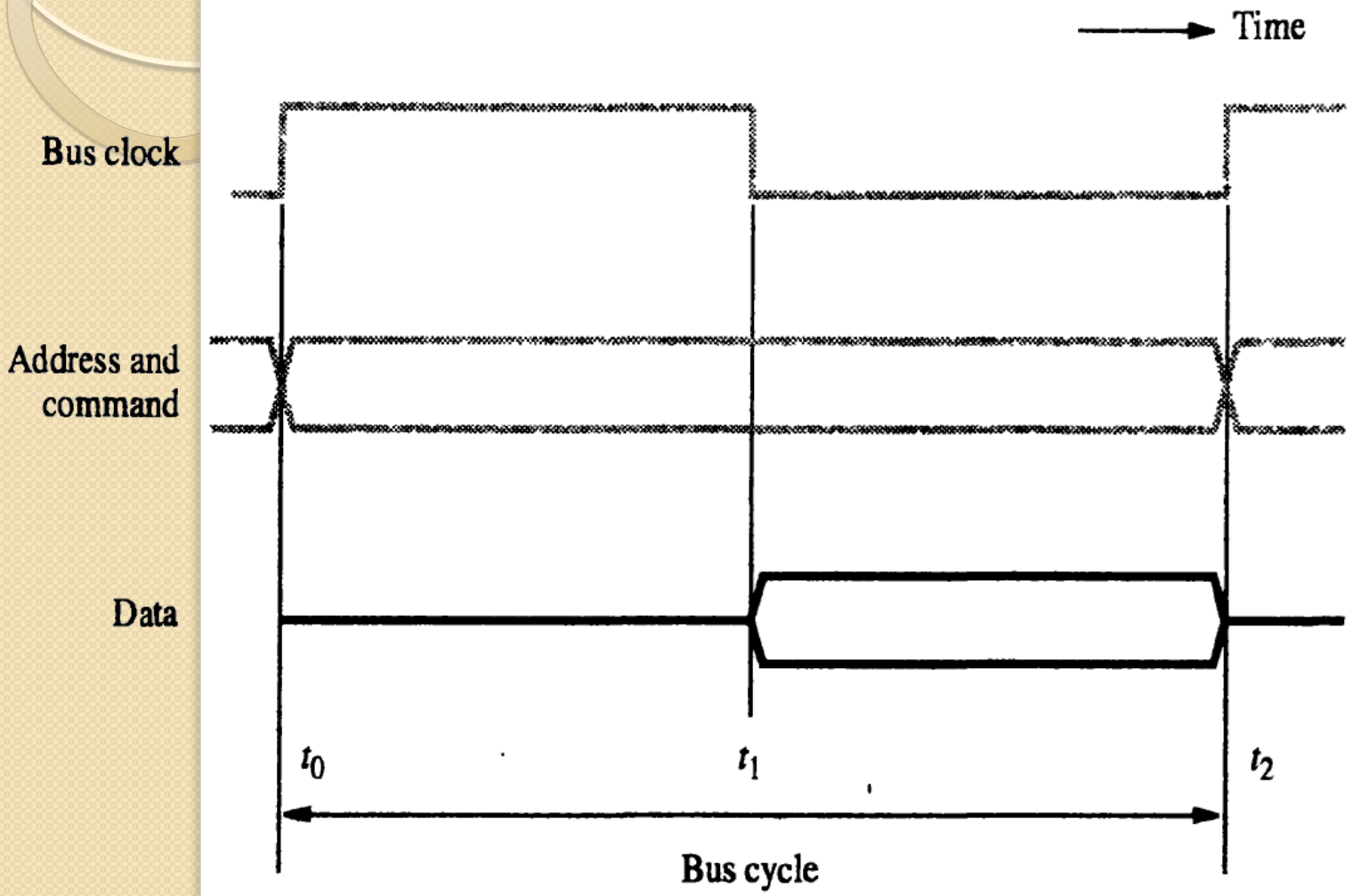
Introduction

- Processor, main memory, and I/O devices connected by a common bus to provide a communication path for the transfer of data
- Bus protocol – set of rules that govern the behaviour of devices connected to the bus, like when to place information, assert control signals, etc.
- Three types of bus lines – address, data, control
  - Control signals – Read/Write, size of data, timing information
- Bus master – initiates data transfers (also called initiator) – processor or DMAC
- Slave or target – device addressed by the bus master
- Schemes for timing of data transfers over the bus
  - Synchronous
  - Asynchronous

Synchronous Bus - Introduction

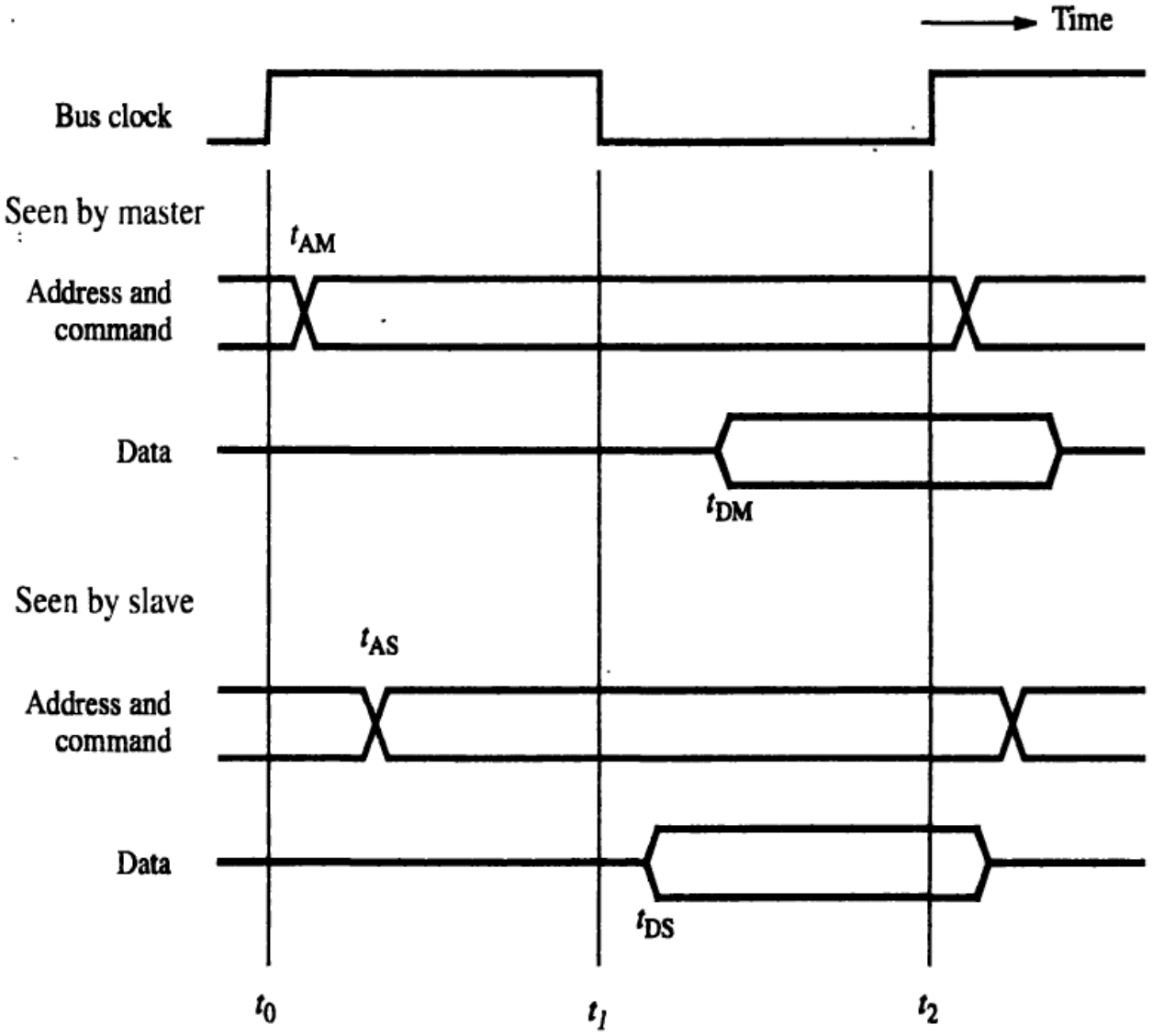
- All devices derive timing information from a common clock line.
  - This clock is not necessarily the processor clock.
  - Equally spaced pulses on this line define equal time intervals.
  - Each interval constitutes a bus cycle during which one data transfer may take place.
- Note:
  - In the timing diagram, address and data lines are shown as high and low at the same time since some lines may be high and some may be low. Crossing points indicate when the values change.
  - Indeterminate or high-impedance state – intermediate level between high and low signal levels

Single Cycle Transfer - Read



- t_0: Master places address and commands
- t_1: Slave transfers data
- $t_1 - t_0 >$ the maximum propagation delay between 2 devices
- Data transfer should not happen between t_0 and t_1 since information on bus is unreliable
- At the end of t_2, the master strobes the data into its input buffer.
- $t_2 - t_1 >$ max propagation time + set up time of input buffer

Read Operation - Detailed

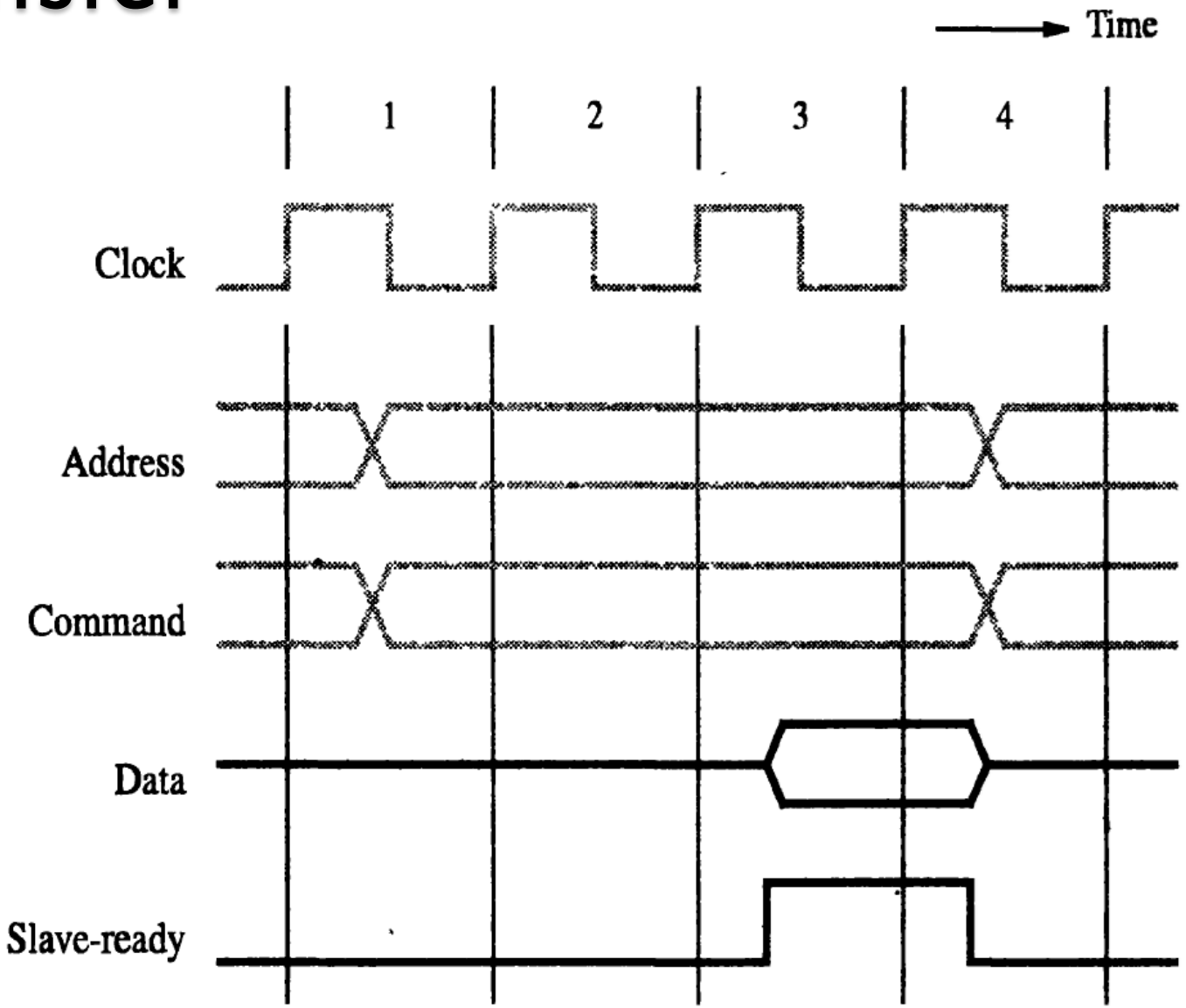


Multiple-Cycle Transfer

- In the previous case, data has to be transferred within one clock cycle.
- Limitations:
  - The clock cycle period must be chosen to accommodate the longest delays on the bus and the slowest device interface.
  - This forces all devices to operate at the speed of the slowest device.
  - The processor has no way to determine if the addressed device has responded. If the device does not respond, malfunction will not be detected.
- To overcome these limitations, incorporate control signals to represent response from the device and use a high frequency clock.
  - Duration of data transfer can be adjusted to suit the device.
  - No. of clock cycles involved will vary from device to device.

Multiple-Cycle Transfer

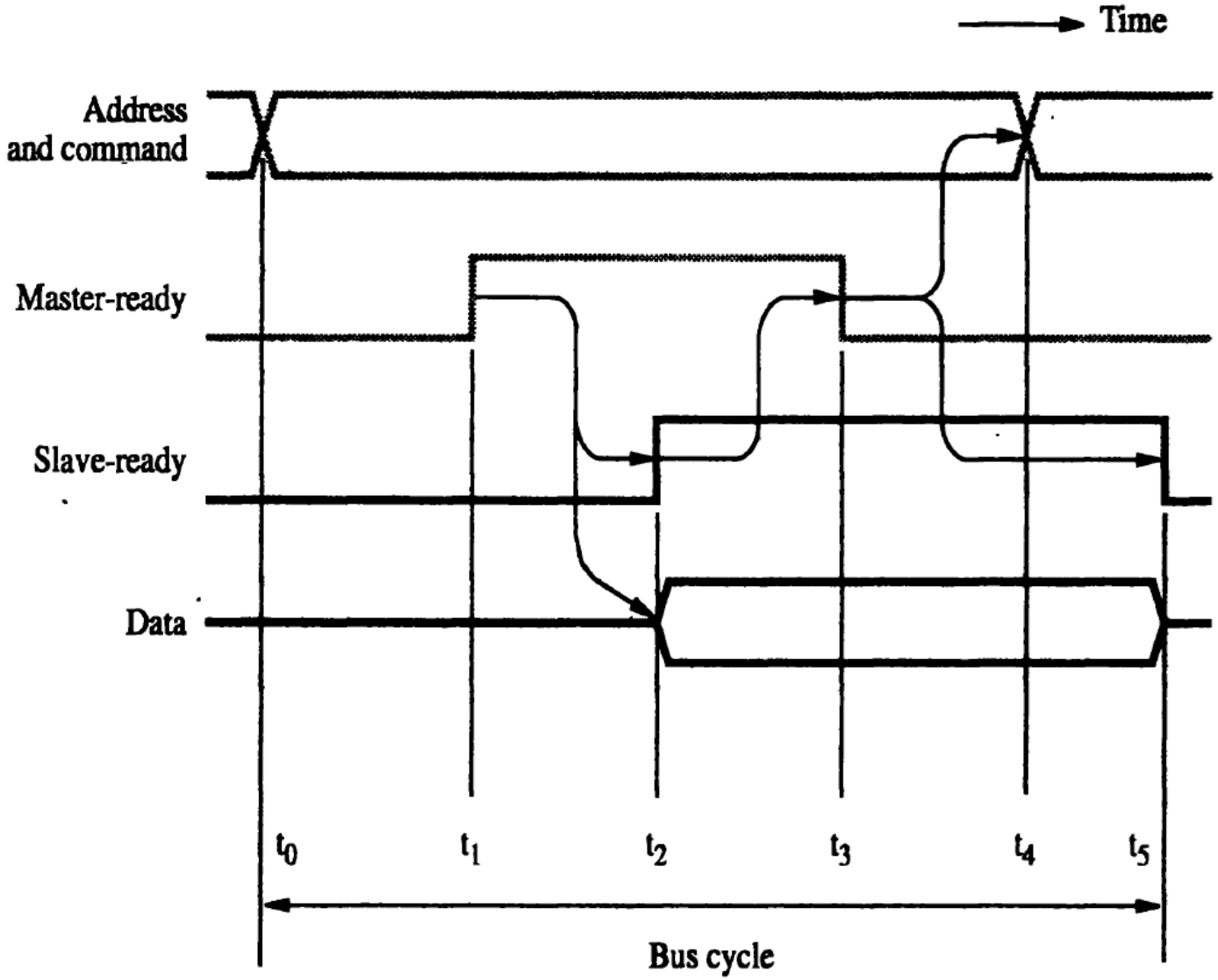
- Clock cycle 1: Master sends address and command information.
- Slave receives the info and decodes.
- Clock cycle 2: Slave decides to respond and starts to access the requested data.
- Clock cycle 3: Data is ready and slave places it on the bus and asserts Slave-ready signal.
- Master strobes the data into its buffer at the end of clock cycle 3.
- Master can start another transfer in cycle 4.
- If addressed device does not respond, master aborts the operation after some cycles.



Asynchronous Bus - Introduction

- Data transfers controlled by handshake between master and slave
- Common clock is replaced by two timing control lines
  - Master-Ready
  - Slave-Ready
- Data transfer by handshake protocol
  - Master places address and command info on the bus and asserts the Master-
  - Ready signal after a delay to allow for any skew that may occur.
  - All devices on the bus decode the address.
  - Selected slave performs the required operation and asserts the Slave-Ready

Read / Input Operation



Write / Output Operation