DAY 13 – SESSION 27 & 28

Tujuan: Sistem bus lengkap, serta pengaruh beban pada bus

Pokok bahasan:

- Sistem bus lengkap
- Konfigurasi sistem kabel pada diagram ruang dan waktu

Reference: Buku 1 (Chapter 7.2)

INTERCONNECTION NETWORKS

Interconnection networks in multiprocessor system provides resource sharing mechanism for the whole system. There are 2 interconnection networks in most multiprocessor system:

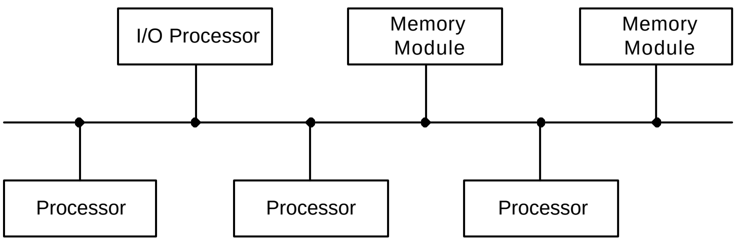
- One Interconnection Network between the processors and memory modules
- One Interconnection Network between the processors and the I/O subsystem

In implementation of an interconnection network (IN), it can have one of three different physical forms:

1. TIME SHARED / COMMON BUSES

SINGLE - BUS MULTIPROCESSOR ORGANIZATION

This is the simplest IN for multiple processors. It basically is a common communication path connecting all of the functional units. The common path is called time shared or commom bus.



Figrure. Single – bus multiprocessor organization

This configuration of interconnection network constructred using totally passive unit with no active components such as switches. Thus transfer operation are controlled completely by and are fully the responsible of the bus interfaces which are located on each of the sending and receiving units. Any connected devices (I/O, Processors, or Memory) can use the common bus for sending or receiving data. Imagine a public road used by various vehicles to move around between buildings. Thus control or syncronization mechanism is required to resonve contention.

Each unit (processor or I/O) that wishes to initiate a transfer (use the common bus) must first determine the availability status of the bus, then address the destination unit to determine its availability and capabilities to receive the transfer. The sending unit will also send a command to specify what operation the receiving unit must do to the data transferred. After the bus is ready and both units agree on the transfer mechanism, the sending unit will initialize transfer. The receiving unit will then see and recognizes the its address placed on the bus, responds to the control signal from the sender accordingly.

The advantages of Time Shared / Commoon Bus Interconnection Network are:

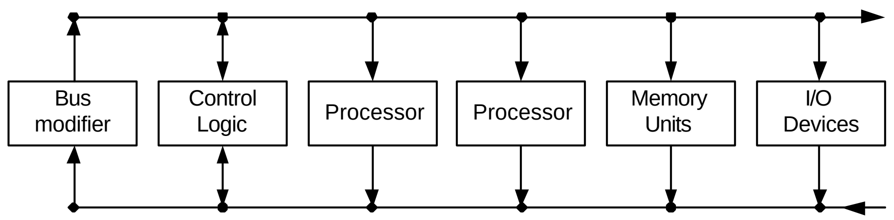
- Leleast complex configuration of Interconnection Network (IN)
- Easiest to reconfigure

The disadvantage of Time Shared / Common Bus Interconnection Network are:

- Additional processor or memory will increase the possibility for bus contention and thus degrade the overall system performance and increase arbitration logic to resolve conflict.
- The total transfer rate within the system is limited by the bandwidth and speed of this single path (the common bus), thus devices capable of higher transfer rate than the bus (e.g. private memories and private I/O) will experience significant disadvantage since they have to adjust to slower speed than what they can achieve independetly.

MULTIPROCESSOR WITH UNIDIRECTIONAL BUSES

Extension of the single path organization is the two unidirectional paths. This configuration is used to overcome some problems found in the previous configuration without significantly increase the complexity of the design.

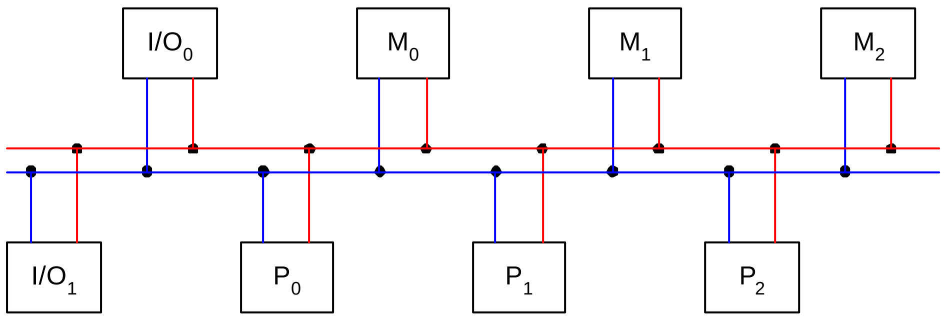


Figrure. Multiprocessor with multi-unidirectional buses

The problem with this configuration is that in every single transfer operation usually requires the use of both buses, thus not much improvement can be gained from the modification.

MULTIPLE BI-DIRECTINAL BUS WITH MULTIPLE PROCESSOR ORGANIZATION

All the above mentioned vconfiguration is usually suitable for small systems, but larger system requires different level of Interconnection Network. The next level of complexity is to permit multiple simultaneous bus transfers. This configuration will increase the system complexity significantly and requires the interconnection subsystem to become an active device.



Figrure. Multiprocessor with multi-unidirectional buses

There are several conflict-resolution methods used in this configuration to arrange which device can send or receive data (use the bus) at each time.

STATIC PRIORITY ALGORITHM.

Each devices requesting for access to the bus will be assigned unique static priorities. When multiple devices concurrently request the use of the bus, devices with higer priority will be granted access rights first. This algorithm will give high priority treatment to devices that are in close proximity to the central bus controller. The algorithm is usually implemented using the "Daisy Chainning" scheme.

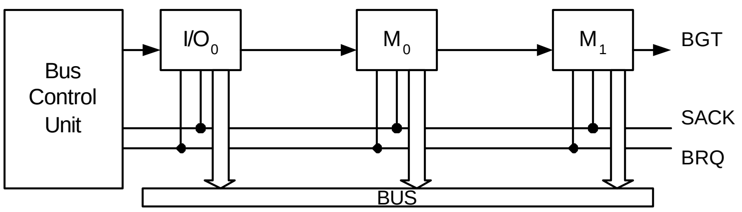


Figure. Static Priority Algorithm

All request made by the devices are sent to the Central Bus Control through a commnon request line, BRQ (Bus Request). The Central Bus Control will check if the bus is idle and receive acknowledge signal (SACK) when it does. The Central Bus Control then send Bus Grant signal (BGT) to the requesting device, give bus control to the device and set the bus-busy flag in the controller. During the use, all BRQ will be turned down until the current device release control of the bus and bus-busy flag is reset.

THE FIXED TIME SLICE ALGORITHM.

The algorithm devide the bus bandwidth into several fixed-length time slices that are sequentially offered to each device in a round-robin fashion. If the offered device elect not to use its time share, the time slice will remain unused and cannot be used by any other devices. This is known as *fixed time slicing* (FTS) or *time division multiplexing* (TDM). If the devices choose to use the bus, they can take turns to access the bus for a fixed priod of time (time slice) before have to hand over the bus to the next devices in the queue and wait for its next turn.

The algorithm usually implement using synchronous buses which all devices will synchronized to a common clock. The algorithm is also called symetric since it assigned the time slice with looking at the type or location of each of the devices.

The advantage of this approach is that:

- It load-balance all bus request since no preference is given to any device. Thus all devices will be treated equally and have equal chance and time to use the bus.
- It also have fixed/bounded maximum wait time to devices for transfer time calculation.

The disadvantage of the approach is that:

- With the increase in the amount of connected devices, it will suffer from long wait time and thus lower overall performance.
- Lower bus utilization, since not all device will require to use the bus all the time. Much of the time slice provided by the bus will be empty and unused (wasted time slice). Lowering the overall bus utilization.

The configuration is more suitable for relatively constant heavy load bus traffic by all devices. Lower load will cause larger standard deviation between FTS and the wait times. But both algorithm (Static Priority and Fixed Time Slicing) has proof to be quite reliable and simple implementation, thus gainning widespread popularity.

DYNAMIC PRIORITY ALGORITHM.

This approach try to dynamically adjust the load-balancing mechanism to avoid high wait time penalty. Each devices are assigned unique priorities such as those in the Static Priority Algorithm and have to compete for access to the bus, but these priorities can dynamically changed to give greater change for requesting devices to access the bus.

The algorithm uses Least Recently Used (LRU) and Rotating Daisy Chain (RDC) algorithm to determin whose turn is it to use the common bus.

The **LRU algorithm** give the highest priority to the requesting device sthat has not used the bus for the logest interval. Thus it requires to reassign the priorities after each bus cycle.

The **RDC algorithm** have no central controller, but bus-grant line is connected from the last device back to the first device in a closed loop. The device currently in control of the bus will act as the bus conotroller for the next bus request and each devices priorities in the system is determined by the distance from each devices to the current device controlling the bus. Thus the priorities change dynamically and bus access is will be given to the requesting devices closest to the one currently controlling the bus.

[Figure 7.19, Page 486 of Book 1]

THE FIRST COME FIRST SERVED (FCFS) ALGORITHM.

The algorithm uses 'stack-like' approach (FIFO) to queue the request for bus use. Any device requesting the bus will place their request in a queue and wait for the bus to serve them accordingly.

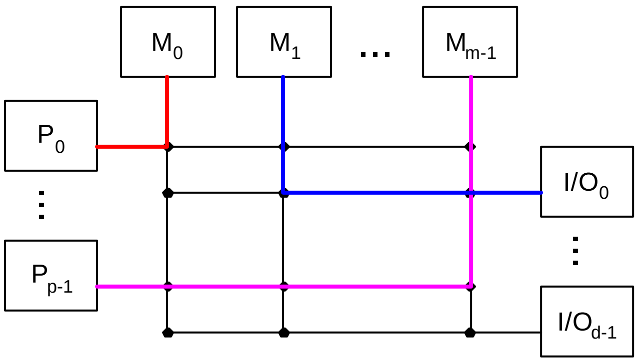
FCFS is considered symetric since it honored the the order of request received and provide better load-balancing approach since it minimized wait time and remove bus inefficiency due to un-used time slice in the previous approach. FCFS is hard to implement since the bus controller has to 'remember' the order of incoming BRQ.

Other technique in bus-control algorithm is *pooling* and *independent requesting*.

2. CROSSBAR SWITCH AND MULTIPORT MEMORIES

When the number of buses in the time-shared bus system is increased, at some point there will be requirement to have separate path for each memory unit in the system. Thus the IN will be called non-blocking crossbar.

Each crossbar switch has the complete connectivity to the memory modules because thereis separate bus associated with each memory module in the system. Thus the maximum concurent transfer (transfer that can take place at the same time) is limited by the number of memory modules and the bandwidth-speed of the bus, not the path.



Figrure. Crossbar switch system organization for multiprocessor

The complexity of this configuration lies on the interconnection matrix switches, each switch must have the capability to switch parallel transmission and also to resolve conflicting request to the same memory module during single memory cycle.

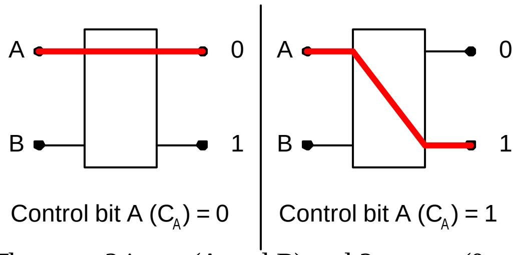
Such conflicting request usually handle following the pre-determined priorities. Implementation of the switch usually requires VLSI (Very Large Scale Integration) circuit to reduce the size of the switch.

Each requesting device (e.g. a processor) generates Request Signal (REQ) to the arbitration unit, which will then select the highest priority devices to grant access to by returning ACK acknowledge signal to the requesting devices. After receiving the ACK, the device will then initiate memory opeartion.

Several variation of the configuration are available to provide a more flexible access and overcome problem encounter when faulty unit exists in the system.

3. MULTISTAGE NETWORKS FOR MULTIPROCESSORS

To understand the multistage networks, first we have to pay close attention to the simple 2x2 crossbar switch below.



There are 2 input (A and B) and 2 output (0 and 1), depending on the Control bit of A, the Input A will be connected to the appropriate output port. The same goes with Input B.

To construct a 1 by 8 demultiplexer using the above switch is not a matter of difficulties.

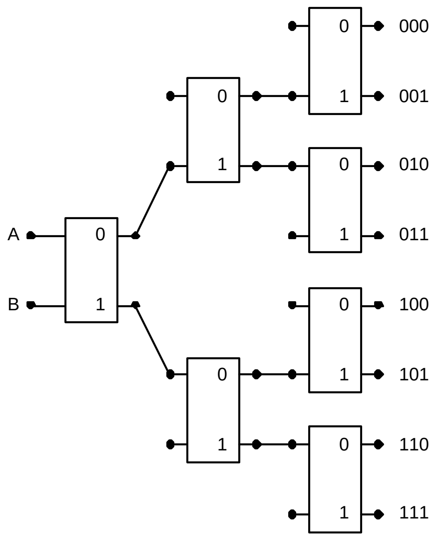


Figure. 1 by 8 demultiplexer

The configuration can be found by constructing 1 x 2^n binary (demultiplexer) tree, where each switch in the stages will determine the path of which the input A (for example) has to follow to connect to one of the8 available output terminal (e.g. output 010). The 1 x 2^n demultiplexer can then further extended to build a 2^n x 2^n multistage network.

BANYAN NETWORK

Banyan network is a partially ordered graph divided into distinct levels. Nodes with no fan-out arc is called base nodes and nodes with no fan-in arc is the apex nodes. Fanout **f** is the number of arcs fanning out from a node, while the spread **s** is the number of arcs fanning into a node.

Banyan network ***(f,s,l)*** is partially ordered graph with ***l*** level of which there is exactly one path from every base to every apex node with ***f*** fanout of each nonbase node and ***s*** spread of nonapex node. Thus each node is a ***s x f*** crossbar switch.

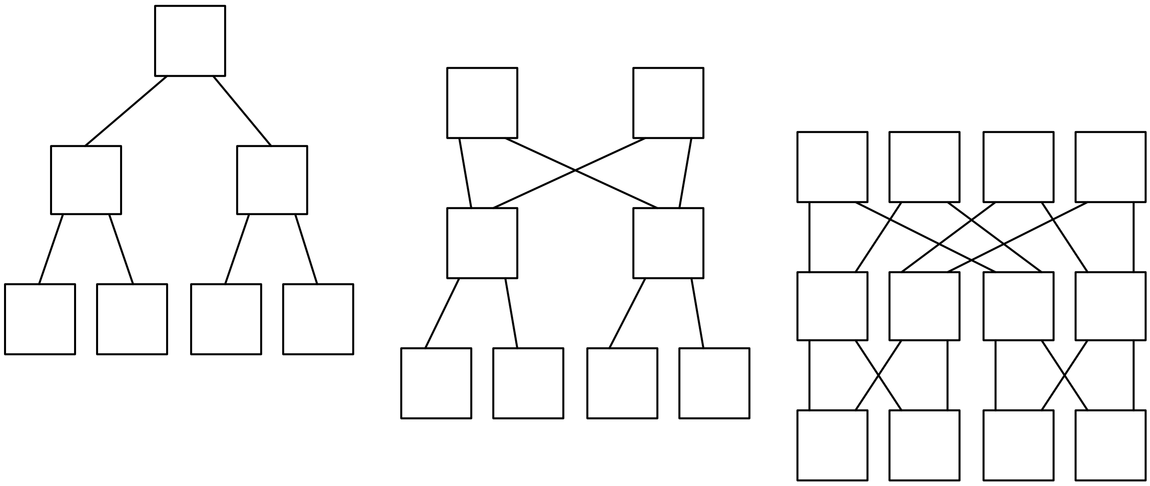


Figure. Foramtion of (2,2,2) Banyan Network.

Advantage of the Banyan Network is it provide complete interconnection of one st of n devices to another set of n devices at cost in swtiching circuitry that grows as n log n. if it is to be implemented using a switch is would cost n^2.

DELTA NETWORK

Delta Network is a a^n x b^n switching network with n stages consisting of a x b crossbar modules. The path configuration is arrange such that there exists a unique path of constant length from any source to any destination. And the switching is digit controlled single base-b digit taken from one of the b destination available. No terminal is left unconnected.

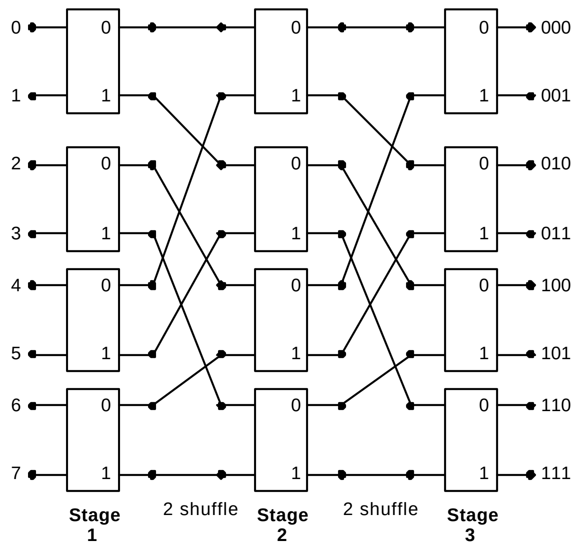


Figure. 2^3 x 2^3 Delta Network

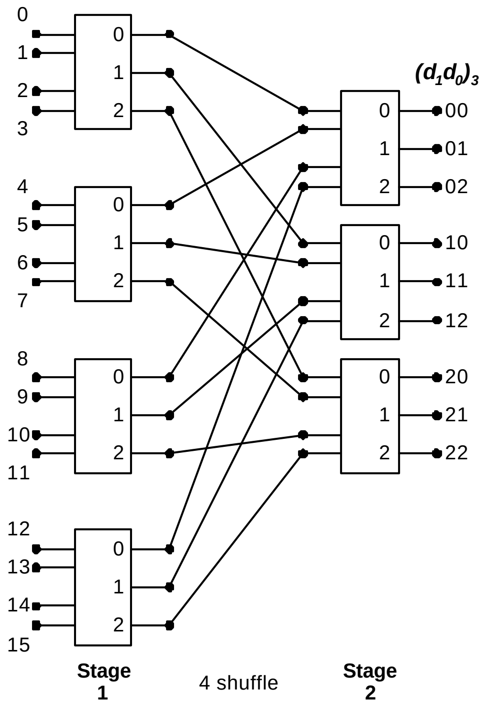


Figure. 4^2 x 3^2 Delta Network