TSN: Lecture 1
Switch and Router Architectures

Topics Covered

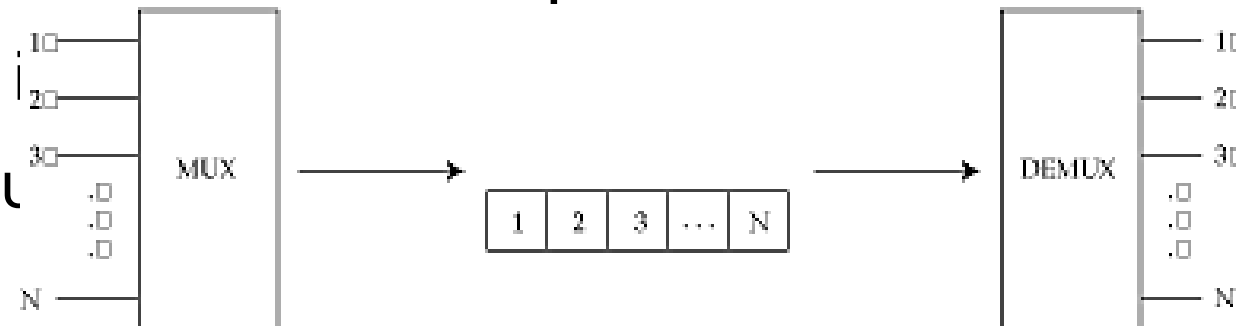
- Circuit switching
- Packet switching
  - Switch generations
  - Switch fabrics
  - Buffer placement
  - Multicast switches

Circuit switching

- Moving 8-bit samples from an input port to an output port
- Recall that samples have no headers
- Destination of sample depends on *time* at which it arrives at the switch
  - actually, relative order within a *frame*
  - *once connection is setup, a time slot assigned, the sample always arrives in that slot*
  - *No other header are necessary*
- We'll first study something simpler than a switch: a multiplexor

Multiplexors and demultiplexors

- Most trunks time division multiplex voice samples
- At a central office, trunk is demultiplexed and distributed to active circuits
- Synchronous multiplexor
  - N i
  - Ou



More on multiplexing

- Demultiplexor
  - one input line and N outputs that run N times slower
  - samples are placed in output buffer in round robin order
- Neither multiplexor nor demultiplexor needs addressing information (why?)
- Can cascade multiplexors
  - need a standard
  - example: DS hierarchy in the US and Japan
    - DS0 = 64Kbps single voice circuit
    - T1/DS1 = 24 DS0 = 1.544Mbps
    - T3/DS3=28 T1 = 672 DS0

Inverse multiplexing

- Takes a high bit-rate stream and scatters it across multiple trunks
- At the other end, combines multiple streams
  - re-sequencing to accommodate variation in delays
- Allows high-speed virtual links using existing technology

A circuit switch

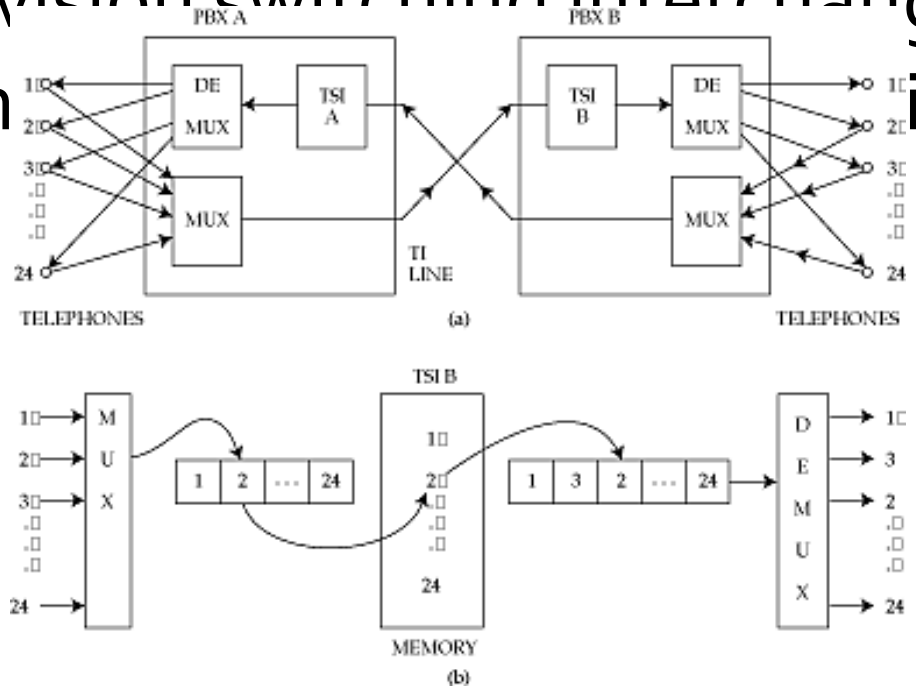
- A switch that can handle N calls has N logical inputs and N logical outputs
  - N up to 200,000
- In practice, input trunks are multiplexed
  - example: DS3 trunk carries 672 simultaneous calls
- Multiplexed trunks carry *frames* = set of samples
- Goal: extract samples from frame, and depending on position in frame, switch to output
  - each incoming sample has to get to the right output line and the right slot in the output frame
  - demultiplex, switch, multiplex

Call blocking

- Can't find a path from input to output
- Internal blocking
  - slot in output frame exists, but no path
  - Switches are classified as blocking or non-blocking
    - depends upon the architecture
    - a characteristic of the switch architecture
- Output blocking
  - no slot in output frame is available (no resources)
  - independent of switch internal blocking
    - occurs for either blocking or non-blocking switches
  - causes Head of Line (HOL) blocking

Time division switching

- Key idea: when demultiplexing, position in frame determines output trunk
- Time division switching interchanges sample position (TSI)

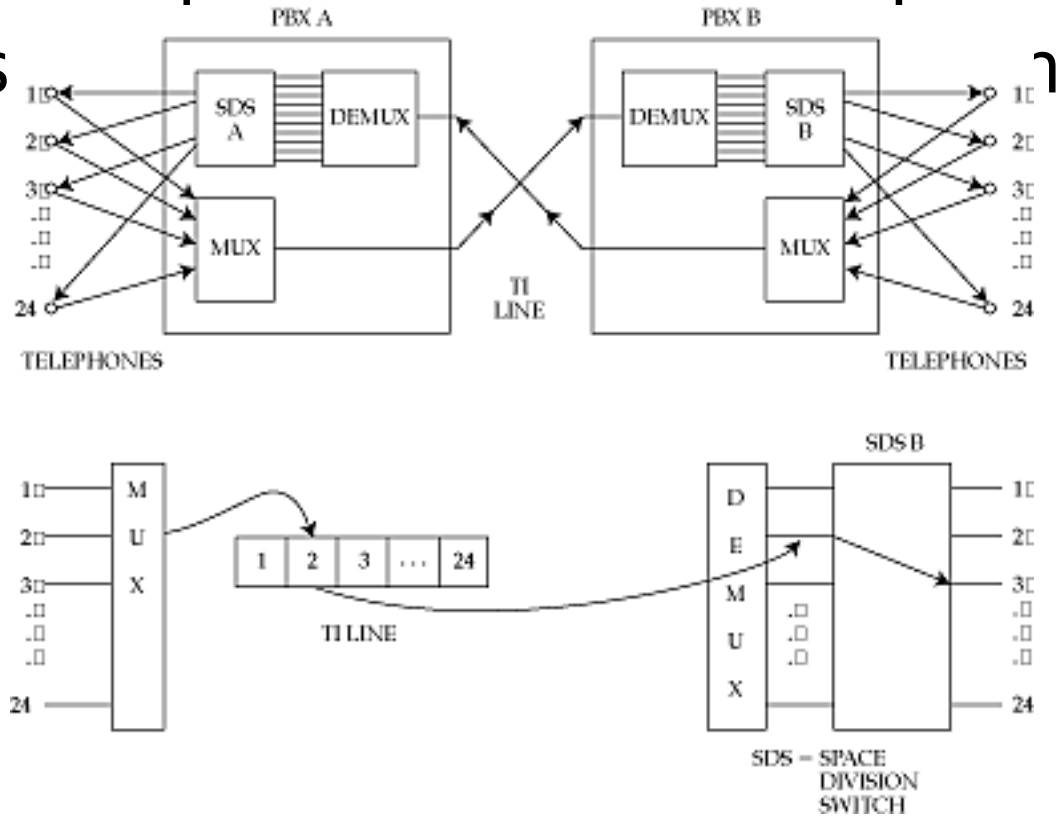


How large a TSI can we build?

- Limit is time taken to read and write to memory
- For 120,000 circuits
  - need to read and write memory (2 operations) once every 125 microseconds
    - Voice = 64Kbps
    - Sample = 8 bytes
    - Rate = 8000 samples / second
    - Time = 1/8000 = 125 microseconds per sample
  - each operation (read or write) takes around 0.5 ns for 120000 circuit TSI
    - impossible with current technology

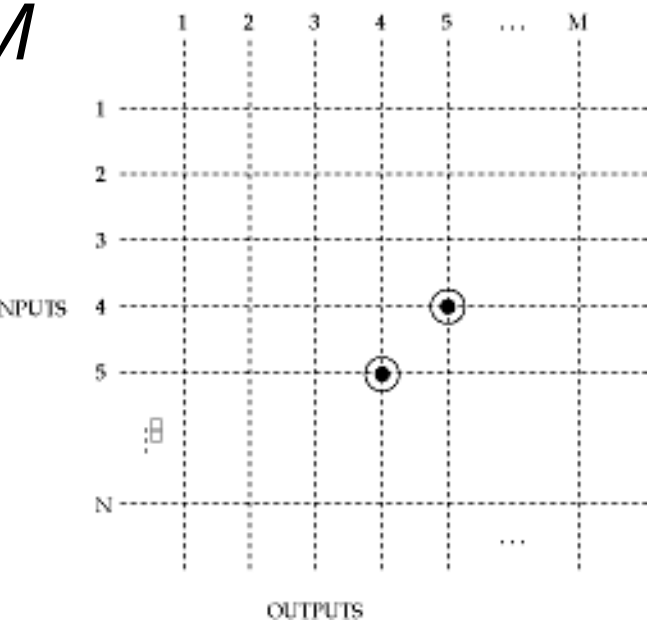
Space division switching

- Each sample takes a different path through the s nation



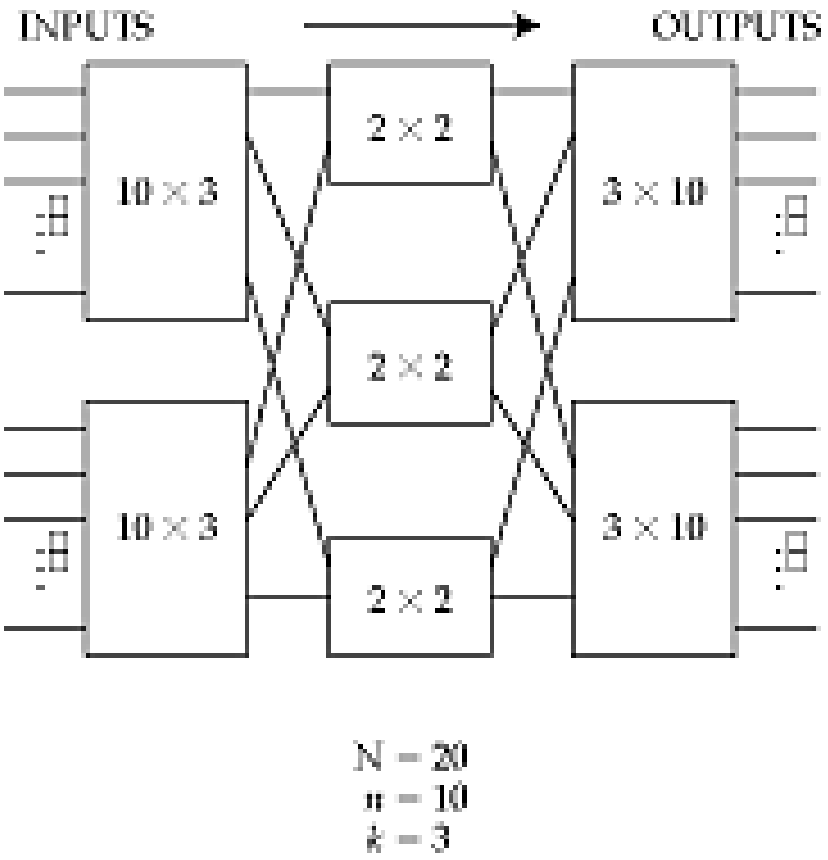
Crossbar

- Simplest possible space division switch. *NxM* crossbar has *N* inputs and *M* outputs
- *Crosspoints* can be turned on or off (think of a design)
- Need a switching *schedule* (why and what frequency??)
  - Multiplex and non-multiplex signals
- Internally non-blocking (why?)

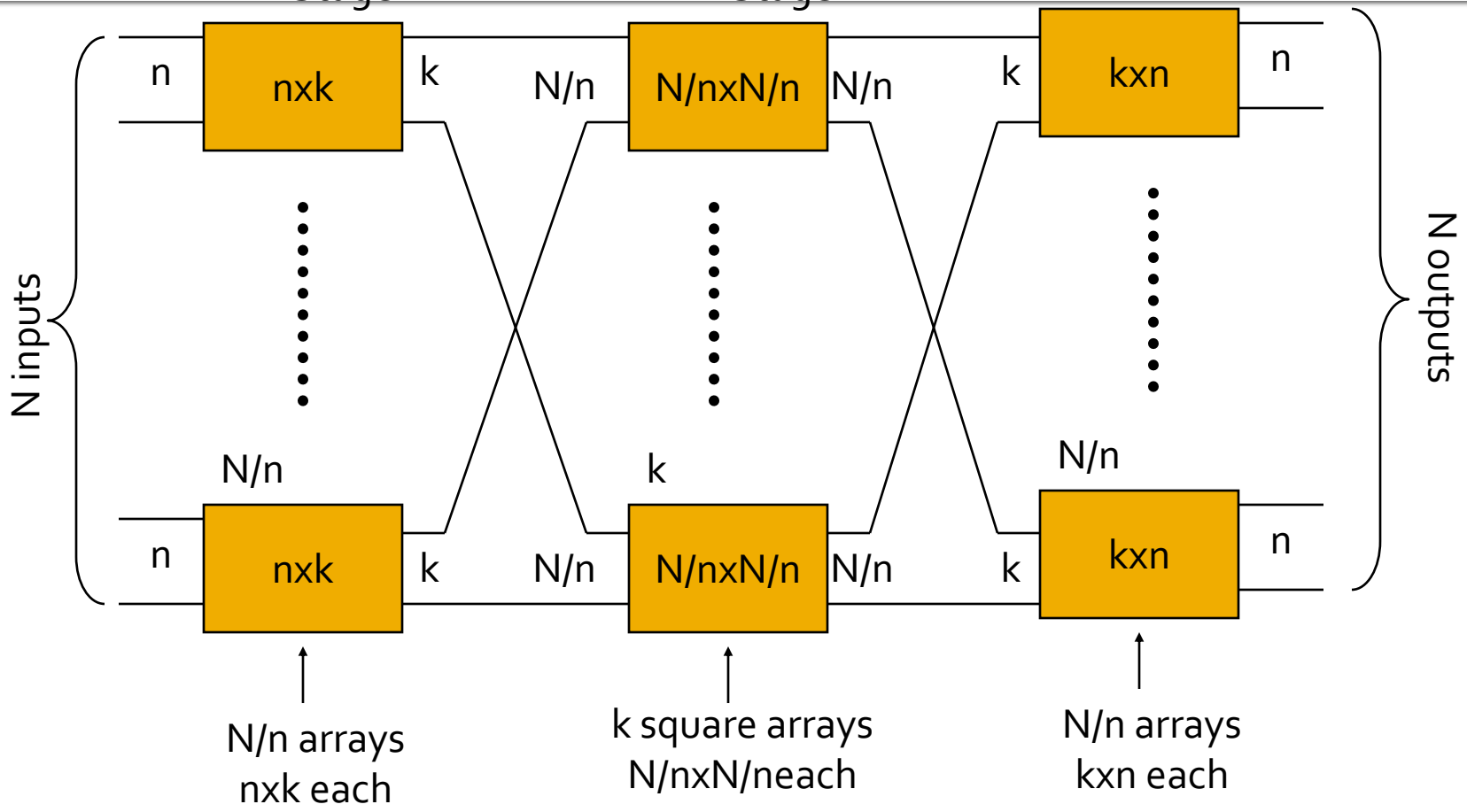


Multistage crossbar

- In a crossbar during each switching time only one crosspoint per row or column is active
- Can save crosspoints if a crosspoint switch can attach to more than one input line (why?)
- This is done in a multistage crossbar
- Inputs are broken into groups (e.g, 20 lines, 2 groups of 10 lines each)
- Multiple paths between inputs and output group share a centre stage switch
- Need to rearrange connections every switching time (switching schedule)



Multistage Switching



Total Number of Cross Points = $2Nk+k(N/k)^2$

Multistage crossbar

- First stage consists of *N/n* arrays of size *nxk each*
- Second stage consists of *k* arrays of size *N/n x N/n each*
- Third stage consists of *N/n* arrays of size *kxn each*
- *Can suffer internal blocking*
  - *unless sufficient number of second-level stages*
- Number of crosspoints < N^2
- Finding a path from input to output
  - switch controller needs to find a path at the time of call setup
  - uses path search algorithms, such as depth-first-search
  - the path is then stored in the switch schedule
- Scales better than crossbar, but still not too well
  - 120,000 call switch needs ~250 million crosspoints