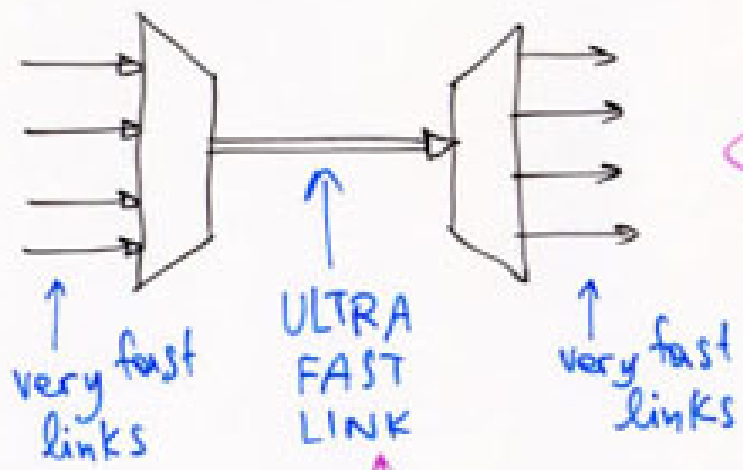


## 4.1 Space Switching, The Crossbar, Time-Space-Time (TST) Switching, and the (Static) Crossbar Scheduling Problem

- Space Switching
  - there is no single “point” where all traffic passes through
- The Crossbar
  - a space switch where every output independently selects any input
- Time-Space-Time (TST) Switching
  - a crossbar, time-multiplexed among several sets of connections
- The Crossbar Scheduling Problem
  - Statically – circuit switching (here): schedule a TST switch
  - Dynamically – packet switching (next para.): input queued switches

Can all switching be performed using Time Division Switching?

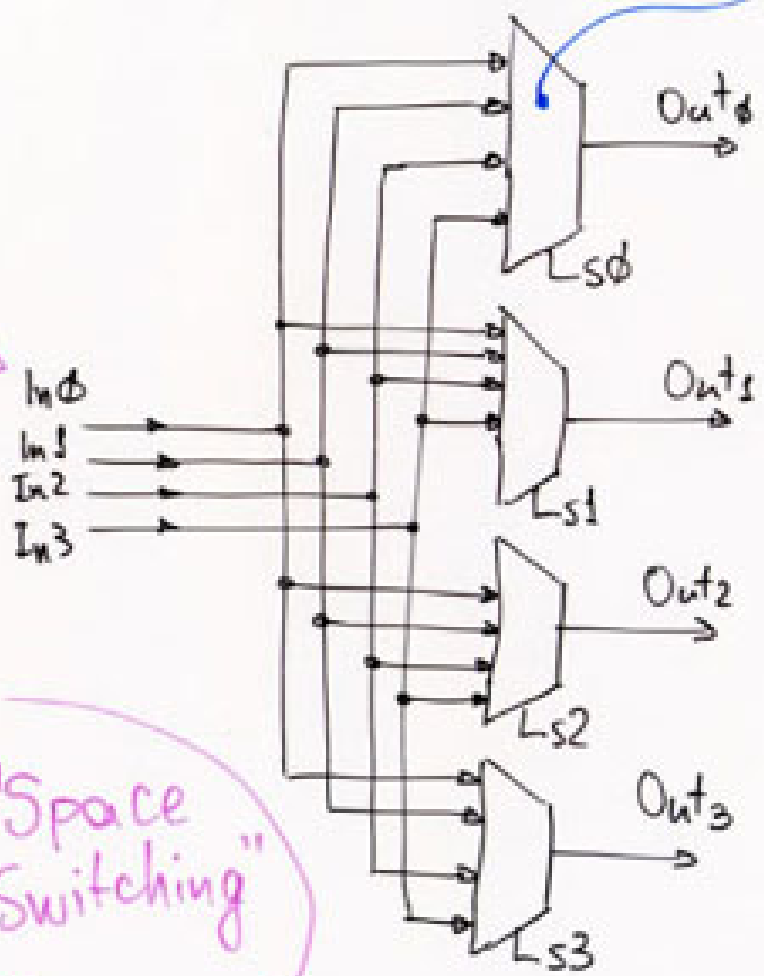
selection (traditional mux)  
 - not parallel-to-serial  
 - not TDM



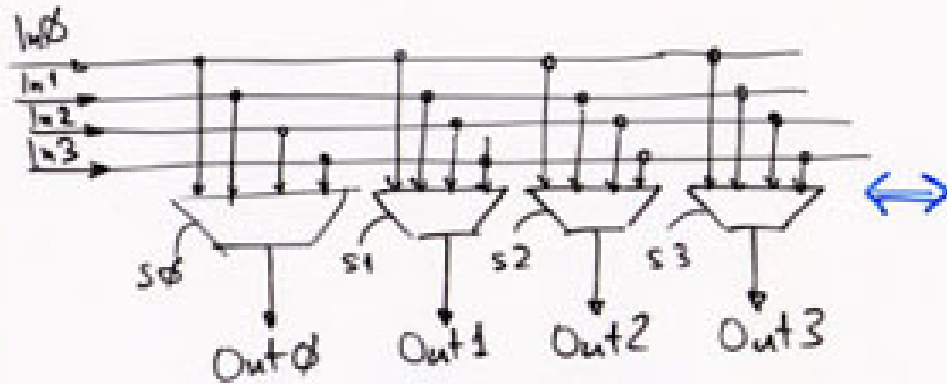
what if this is not implementable?.....



"Space Switching"



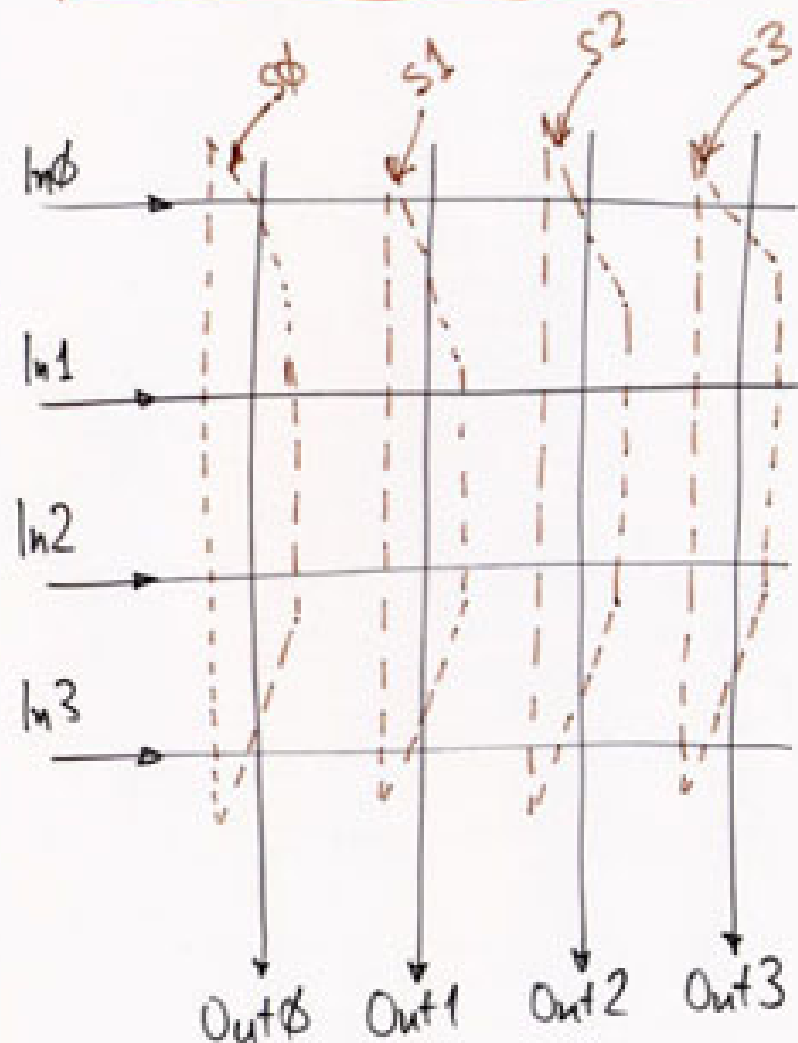
Turn the picture  
90° clockwise...



All Switching/Networking  
is a combination of:

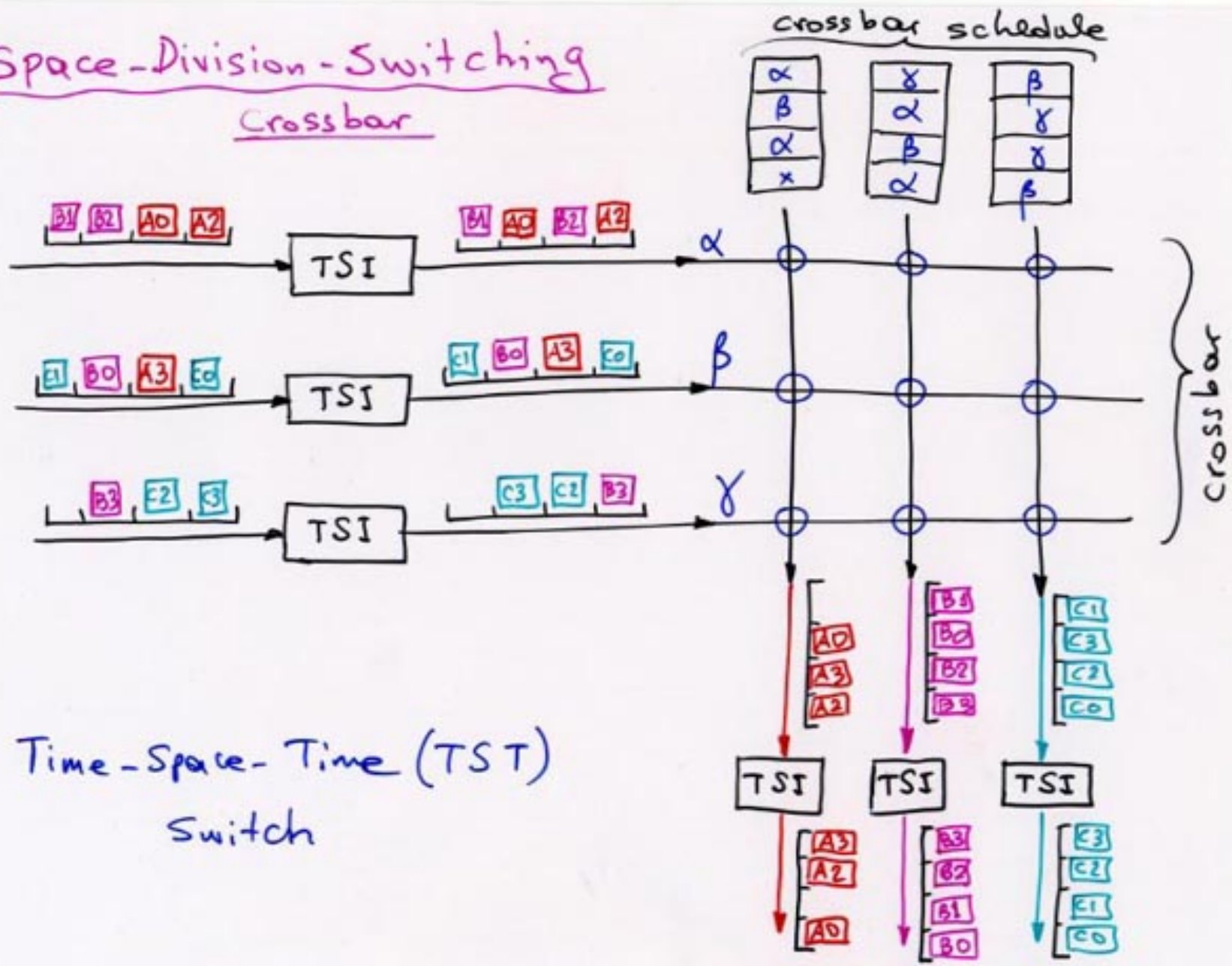
- multiplexing (time div. or statistical)
- cross bars
- buffering
- hierarchical subnetwork-network  
Composition

## Crossbar Switch:



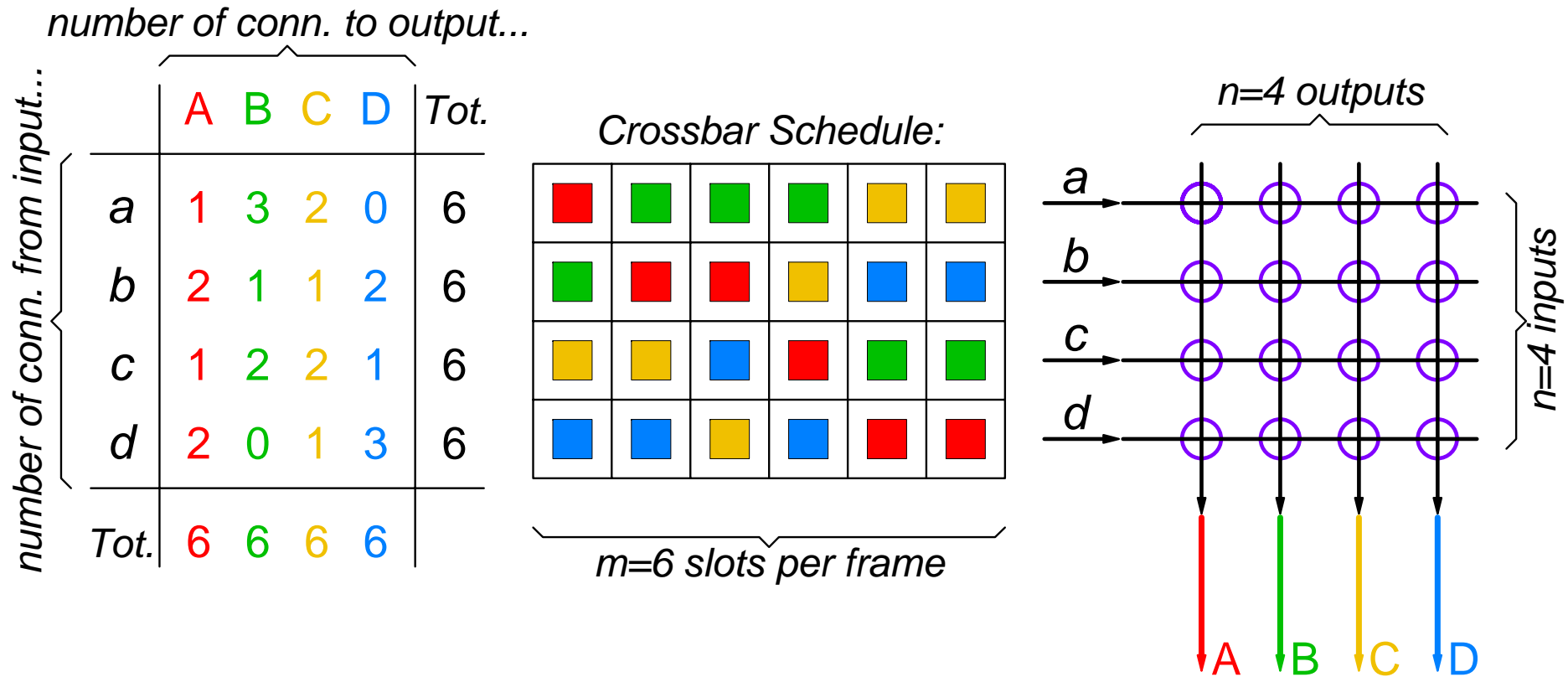
# Space-Division-Switching

## Crossbar



Time-Space-Time (TST)  
Switch

# The (Static) Crossbar Scheduling Problem







- Given the number of connections requested from each input to each output (left matrix), find a schedule (middle)

# Building the Schedule one Step at a Time

number of conn. to output...

		A B C D				Tot.
number of conn. from input...	a	1	3	2	0	6
	b	2	1	1	2	6
	c	1	2	2	1	6
	d	2	0	1	3	6
	Tot.	6	6	6	6	

=

		A	B	C	D	Tot.
a	1	0	0	0	1	
b	0	0	1	0	1	
c	0	1	0	0	1	
d	0	0	0	1	1	
		1	1	1	1	

+

		A	B	C	D	Tot.
a	0	3	2	0	5	
b	2	1	0	2	5	
c	1	1	2	1	5	
d	2	0	1	2	5	
		5	5	5	5	

- Decompose the (full) request matrix into the sum of a permutation matrix plus a “smaller” (full) request matrix
- What about requests arriving one-at-a-time? (exer. 7)