Speedup S

$$S = \frac{\tau_{seq}}{\tau_{pipe}}$$

- In a well-designed pipe the statges are balanced so $\tau_i = \tau_j$ for all i,j
- The latch time is small compared to τ_j
  - $t_e << \tau_i$ for all i

Speedup S

$\tau_i = \tau_j$ and $t_e << \tau_i$

$$\tau = P_{min} = \max(\tau_i) + t_e \approx \tau_i$$

$$\tau_{seq} = n \times \sum_{i=1}^{m} \tau_i = n \times m \times \tau$$

$$S = \frac{\tau_{seq}}{\tau_{pipe}} = \frac{n \times m \times \tau}{(m + n - 1) \times \tau}$$

$$S = \frac{n \times m}{m + n - 1}$$

Speedup S

$S = (n \times m)/(m + n - 1)$

$$S = \frac{n \times m}{m + n - 1}$$

Thus:

$N = 1 \Rightarrow S \approx 1$ but $S < 1$

$N = m \Rightarrow S \approx \frac{m}{2}$ but $S < \frac{m}{2}$

$N >> m \Rightarrow S \rightarrow m$

Pipe Control and Clocking

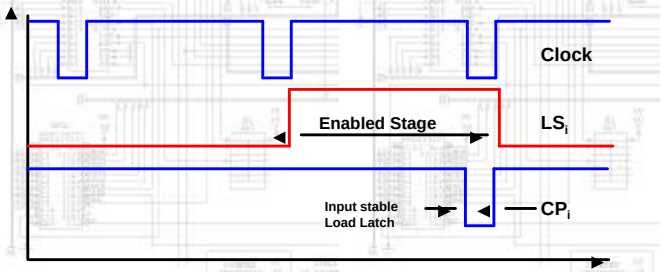
- To operate a pipe we must connect
  - Its input to a source memory M_{so}
  - Its output to a sink memory M_{si}
- The input to the pipeline is controlled by the ENABLE signal and the loading of intermediate results into registers is controlled by signals LS_i derived from ENABLE

Pipe Latch and Clock Design

- Pipe latches must be simple in order to reduce propergation delay
- Simple D latches controlled by a clock pulse CP_i just long enough to laod data.
- I f our latches laod when the CP_i is low and when the enable LS_i is high

Schematically

Two stages example



CP_i Truth Table

CLOCK	LS_i	CP_i
0	0	1
0	1	0
1	0	1
1	1	1

$CP_i = CLOCK + \overline{LS_i}$

Pipe Control

For a pipe with two stage S_1 and S_2 we have