

Latches and flip-flops

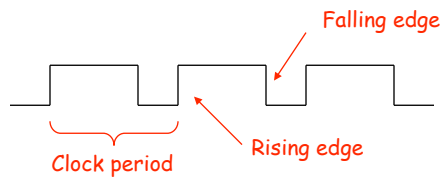
Fundamental elements of memory



CS240 Computer Organization
Department of Computer Science
Wellesley College

Clocks

- o A **clock** is a free-running signal with a fixed cycle time.
- o **Clock frequency** is the inverse of its cycle time.

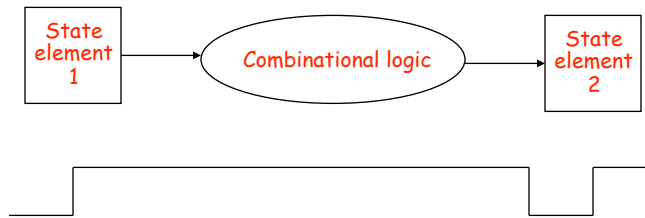


- o Clocks are needed in sequential logic to decide when to update an element's state.



Synchronous systems

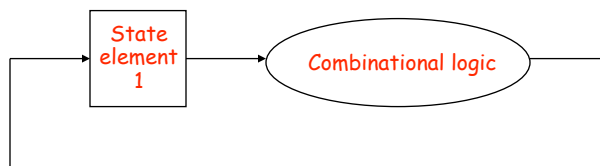
- The signals that are written into state elements must be **valid** when the active clock edge occurs.



Flip-flops 13-3

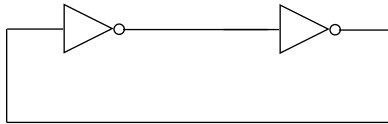
Reading and writing in the same cycle

- An edge-triggered methodology allows a state element to be read and written in the same clock cycle.



Flip-flops 13-4

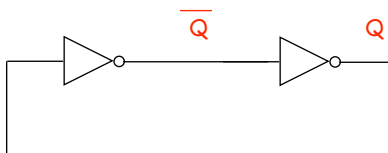
What an odd little circuit



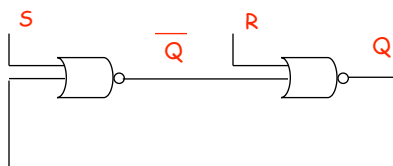
Flip-flops 13-5

Bistable latches

- o Memories



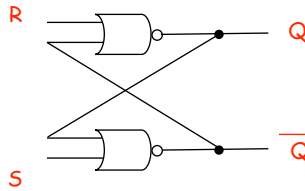
- o New memories



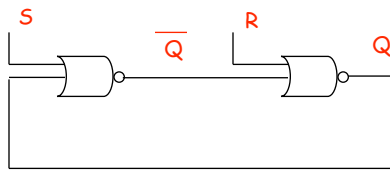
Flip-flops 13-6

Redrawing the SR latch

- o New



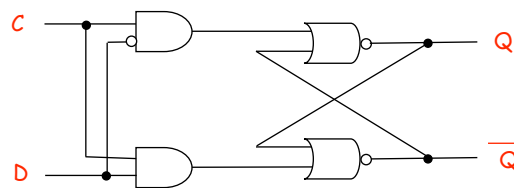
- o Old



Flip-flops 13-7

A D latch

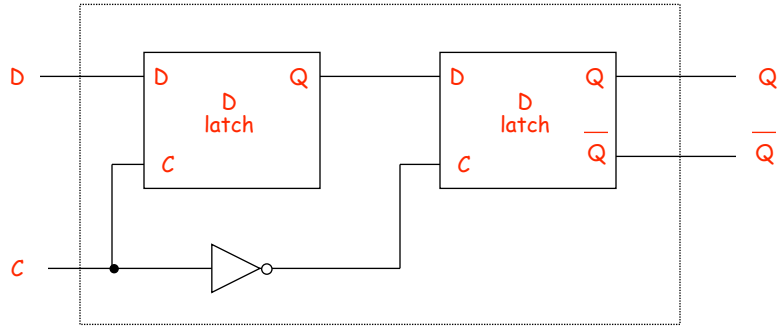
- o When the clock is unasserted, the cross-coupled pair of NOR gates acts to store the state value.



- o When the clock goes high, the Q value tracks the value of D.

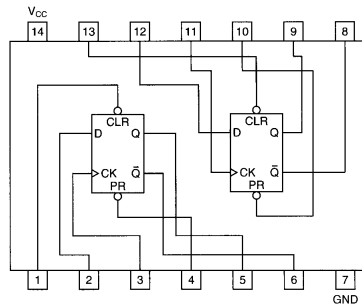
Flip-flops 13-8

A D flip-flop with a falling-edge trigger



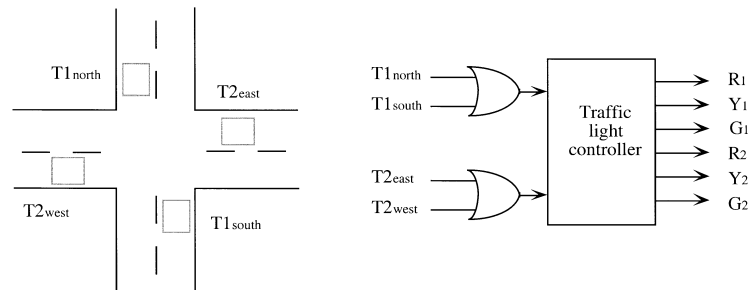
Flip-flops 13-9

Dual D flip-flop



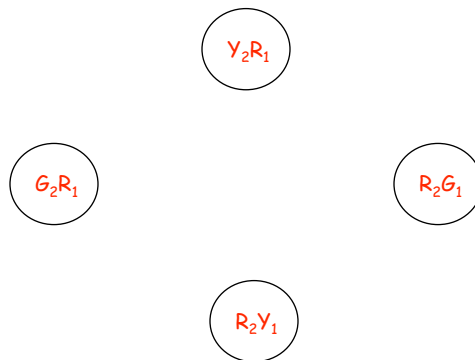
Flip-flops 13-10

Synthesis of sequential circuits



Flip-flops 13-11

Identify the required internal states



Flip-flops 13-12

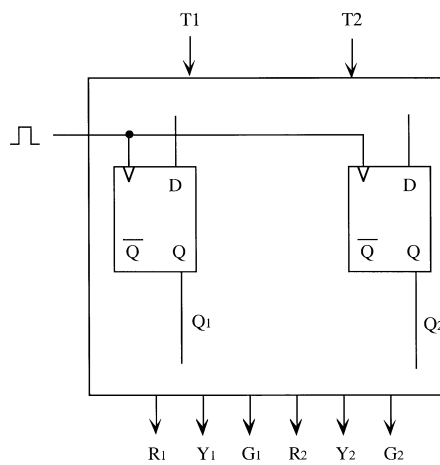
Determine required memory*

CODE			T2	T1	T2	T1	T2	T1	T2	T1	OUTPUTS					
Q2	Q1		0	0	0	1	1	1	1	0	R1	Y1	G1	R2	Y2	G2
G2	R1	0	0													
Y2	R1	0	1													
R2	G1	1	1													
R2	Y1	1	0													

*Assign codes to states and produce state and output tables.

Flip-flops 13-13

Select flip-flops and develop control levels



Flip-flops 13-14

Develop Boolean expressions for controls*

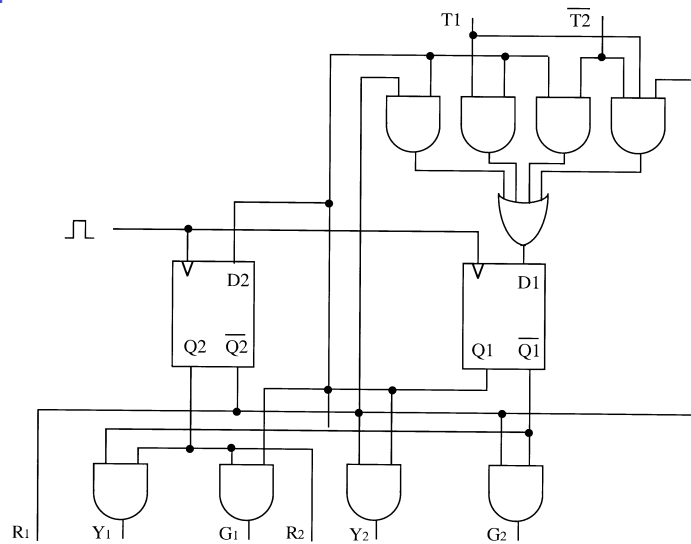
Q ₂	Q ₁	T ₂	T ₁	T ₂	T ₁	T ₂	T ₁	Q ₂	Q ₁	T ₂	T ₁	T ₂	T ₁	T ₂	T ₁	
		0	0	0	1	1	1	0		0	0	0	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	

$D_2 = Q_1$
 $D_1 = \overline{Q_2}Q_1 + Q_1\overline{T_2} + Q_1T_1 + \overline{Q_2}\overline{T_2}T_1$

*And outputs

Flip-flops 13-15

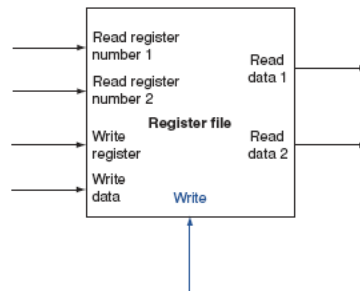
Implement circuit



Flip-flops 13-16

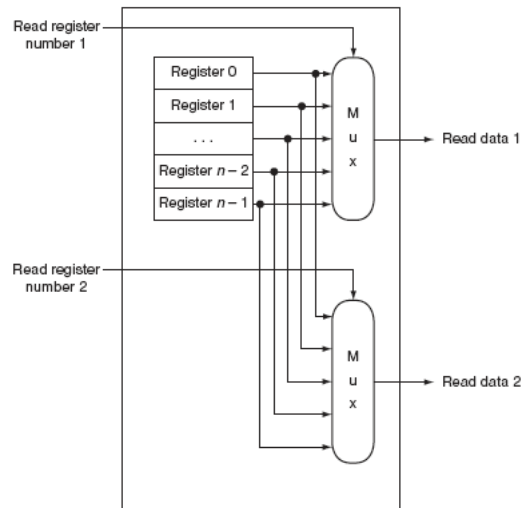
Register files

- o A **register file** consists of a set of registers that can be read and written by supplying a register number.
- o Register files with two read ports will be important later in the semester when we build our MIPS machine.



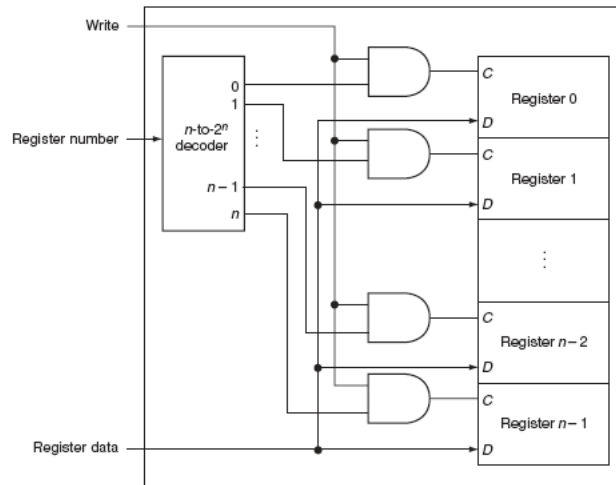
Flip-flops 13-17

Implementation of two read ports



Flip-flops 13-18

Implementation of write port



Flip-flops 13-19