Registers, Counters, Memory Units- III

Assist. Prof. Dr. Özge ÖZTİMUR KARADAĞ

ALKÜ

Previously

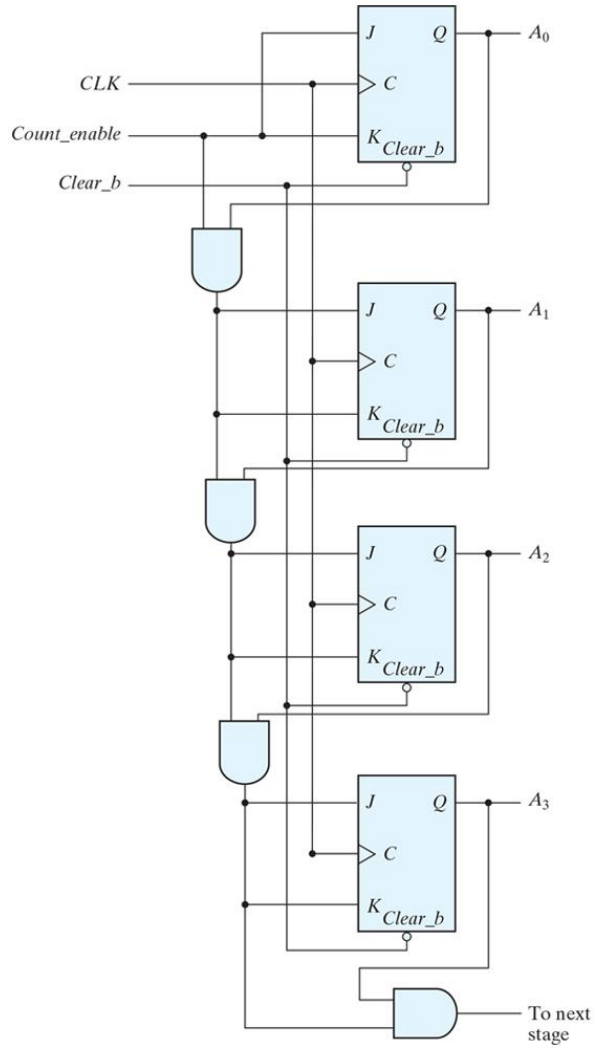
- Registers
  - Register with parallel load
  - Shift registers
- Counters
  - Ripple Counters

Synchronous Counters

- Clock pulses are applied to the inputs of all flip-flops.
- A common clock triggers all flip-flops simultaneously, rather than one at a time in succession as in a ripple counter.
- The decision whether a flip- flop is to be complemented is determined from the values of the data inputs.

Binary Counter

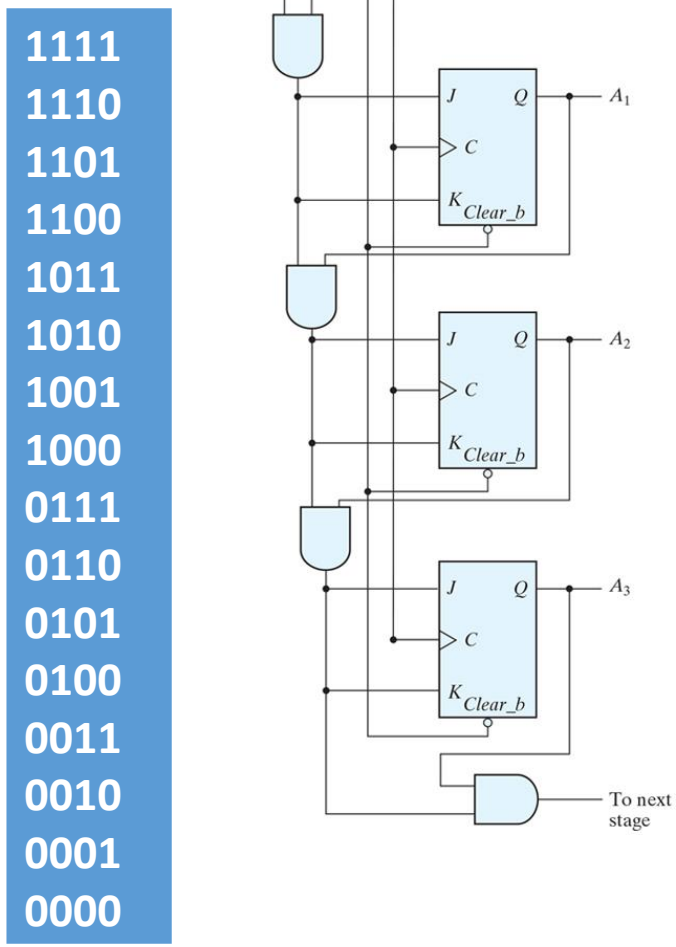
- Simple – no need to go through the logic design process.
- The flip-flop in the least significant position is complemented with every pulse. J=K=1
- A flip-flop in any other position is complemented when all the bits in the lower significant positions are equal to 1.
- Ex: If the current state of a 4 bit synchronous counter is $A_3A_2A_1A_0$=0011 then the next state will be:
  - 0100



4 bit synchronous binary counter

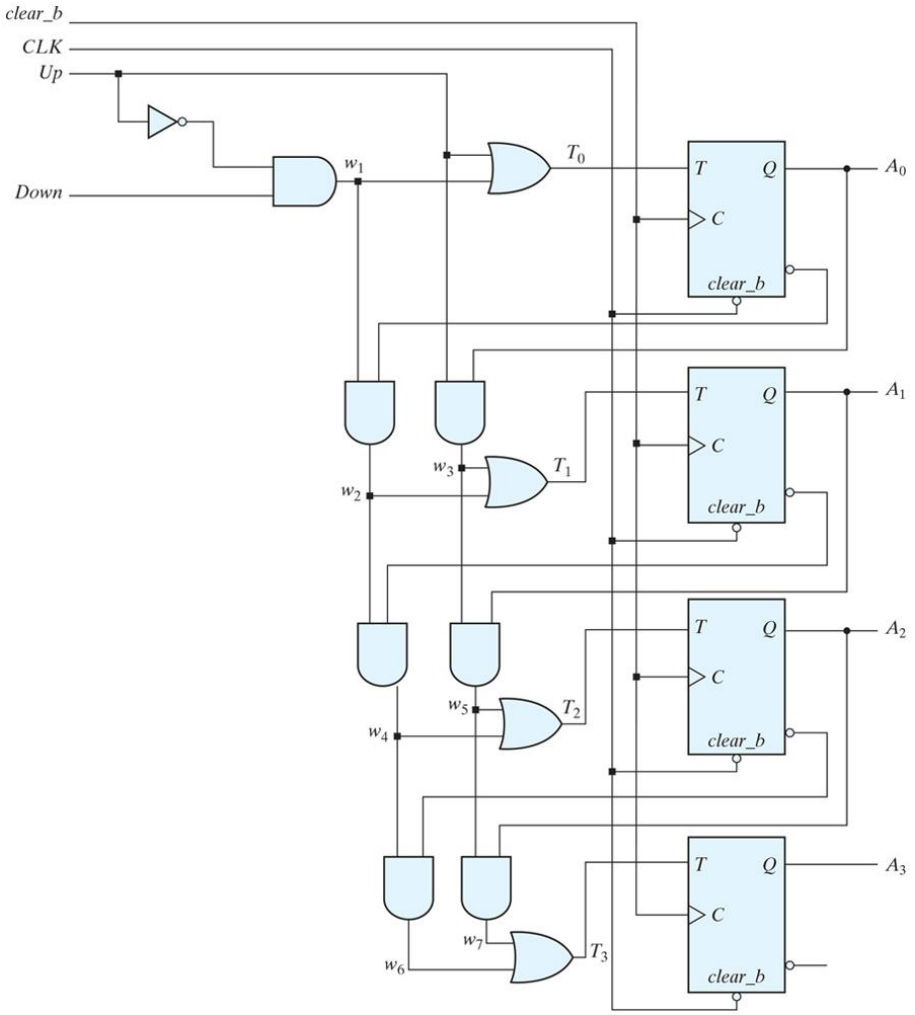
Binary Down Counter

- A synchronous countdown binary counter goes through the binary states in reverse order; from 1111 down to 0000 and back to 1111 to repeat the count.
- The flip-flop in the least significant position is complemented with every pulse.
- A flip-flop in any other position is complemented when all the bits in the lower significant positions are equal to 0.
- Ex: : If the current state of a 4 bit synchronous counter is $A_3A_2A_1A_0$=1100 then the next state will be:
  - 1011
- In this figure, if Q' instead of Q is fed to the AND Gates, then the counter will count down.



Binary Up-Down Counter

- When UP=1 T inputs willl come from the Q outputs of lower order flip-flop, and the counter will count up.
- When Down=1 and UP=0 T inputs will come from the Q' outputs of lower order flip-flop, and the counter will count down.
- When Down=1 and Up=1 the counter will count up.



BCD Counter

T Flip-Flop

T	Q(t + 1)	
0	$Q(t)$	No change
1	$Q'(t)$	Complement

- Binary Coded Decimal
  - 12= 1100 (binary representation)
  - 12 = 0001 0010 (BCD)
- Doesn't have a special pattern. Excitation Table:
- Y output is 1 when the counter's current state is 1001 and it is used to enable the counter for the higher digit.

Present State				Next State				Out put	Flip-flop Inputs			
Q8	Q4	Q2	Q1	Q8	Q4	Q2	Q1	y	TQ8	TQ4	TQ2	TQ1
0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	0	1
0	0	1	1	0	1	0	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	0	1
0	1	0	1	0	1	1	0	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	0	1
1	0	0	1	0	0	0	0	1	1	0	1	1

BCD Counter

Present State				Next State				Out put	Flip-flop Inputs			
Q8	Q4	Q2	Q1	Q8	Q4	Q2	Q1	y	TQ8	TQ4	TQ2	TQ1
0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	0	1
0	0	1	1	0	1	0	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	0	1
0	1	0	1	0	1	1	0	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	0	1
1	0	0	1	0	0	0	0	1	1	0	0	1

	00	01	11	10
00				
01			1	
11	X	X	X	X
10		1	X	X

$TQ_8=Q_8Q_1+Q_4Q_2Q_1$

	00	01	11	10
00			1	
01			1	
11	X	X	X	X
10			X	X

$TQ_4=Q_2Q_1$

	00	01	11	10
00		1	1	
01		1	1	
11	X	X	X	X
10			X	X

$TQ_2=Q_8'Q_1$

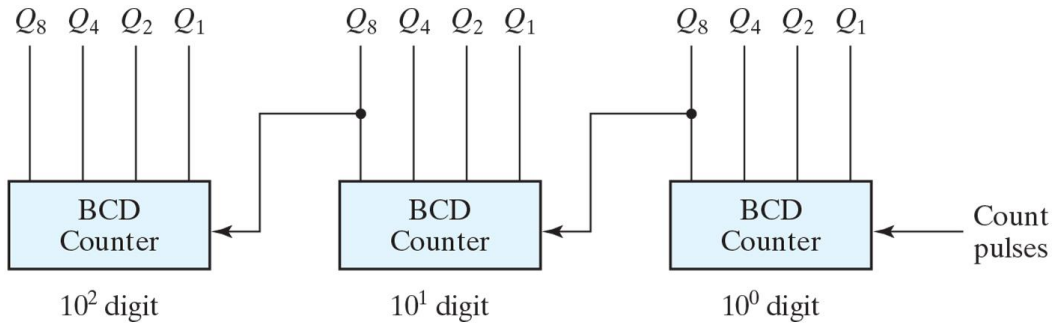
$TQ_1=1$

	00	01	11	10
00				
01				
11	X	X	X	X
10		1	X	X

$y=Q_8Q_1$

BCD Counter

- Synchronous BCD counters can be cascaded to form a counter for decimal numbers of any length.
- How can the following cascaded BCD ripple counter be converted to a synchronous BCD counter?



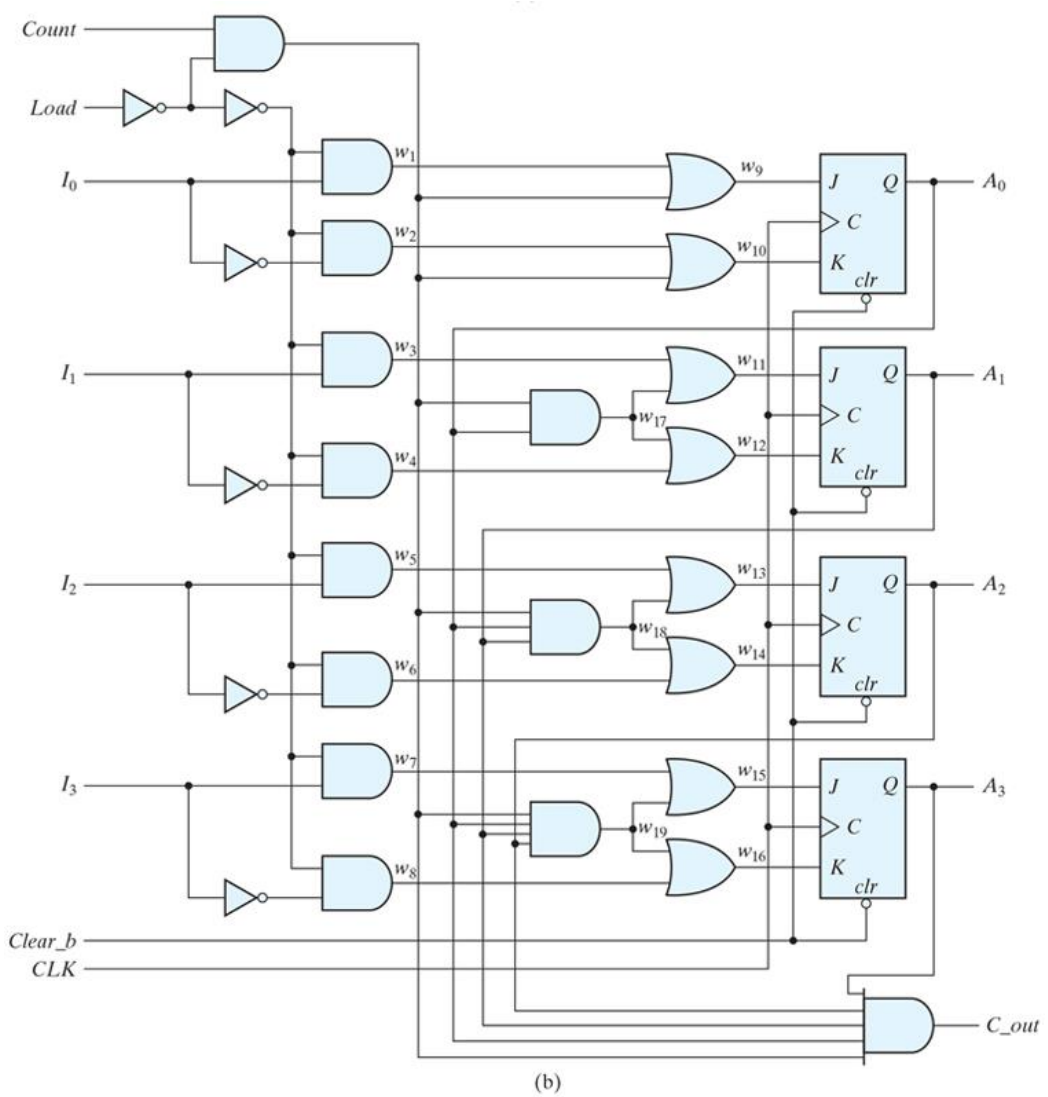
- output y must be connected to the count input of the next-higher significant decade.

Binary Counter with Parallel Load

- In order to load an initial value to the counter → parallel load

- Uyarma Tablosu

Clear_b	CLK	Load	Count	Function
0	X	X	X	
1	↑	1	X	
1	↑	0	1	
1	↑	0	0	

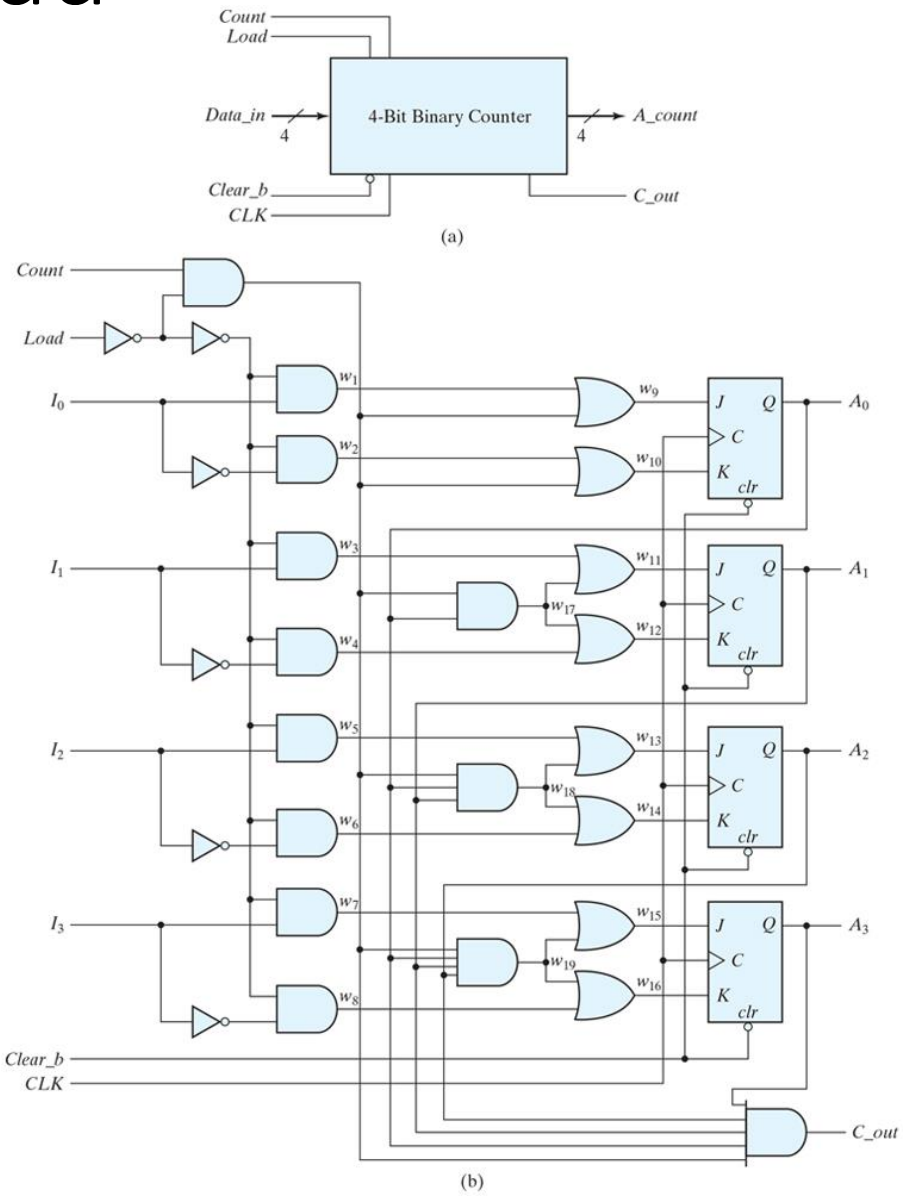


(b)

Binary Counter with Parallel Load

Clear_b	CLK	Load	Count	Function
0	X	X	X	Clear to 0
1	↑	1	X	Load inputs
1	↑	0	1	Count next binary state
1	↑	0	0	No change

- 4 bit counter is available as MSI.
- Ex: for a 16 bit counter, 4 number of this counter is required. The carry out is connected to the count input of the higher order bit.



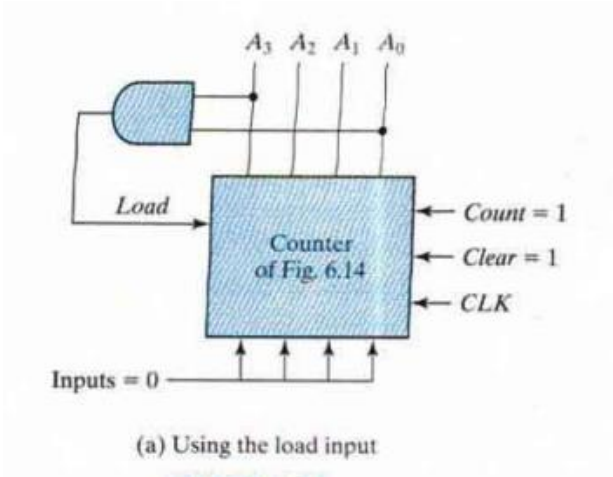
Binary Counter with Parallel Load

- A counter with a parallel load can be used to generate any desired count sequence.
- A Mod-N counter is a counter which counts any sequence of N numbers.
- Ex:
  - 4 bit counter is a mod-16 counter.
  - A BCD counter is a mod-10 counter.

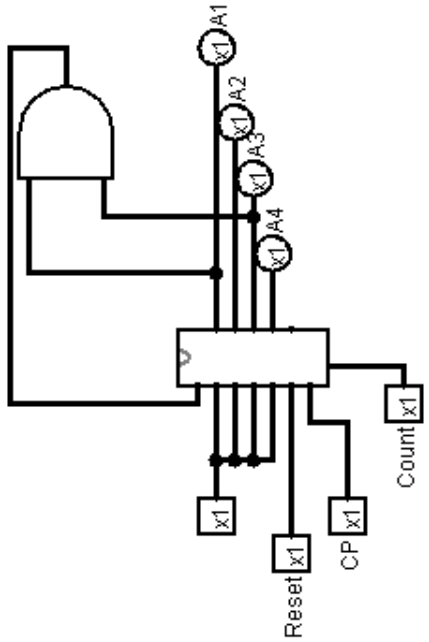
Example

0000
0001
0010
0011
0100
0101

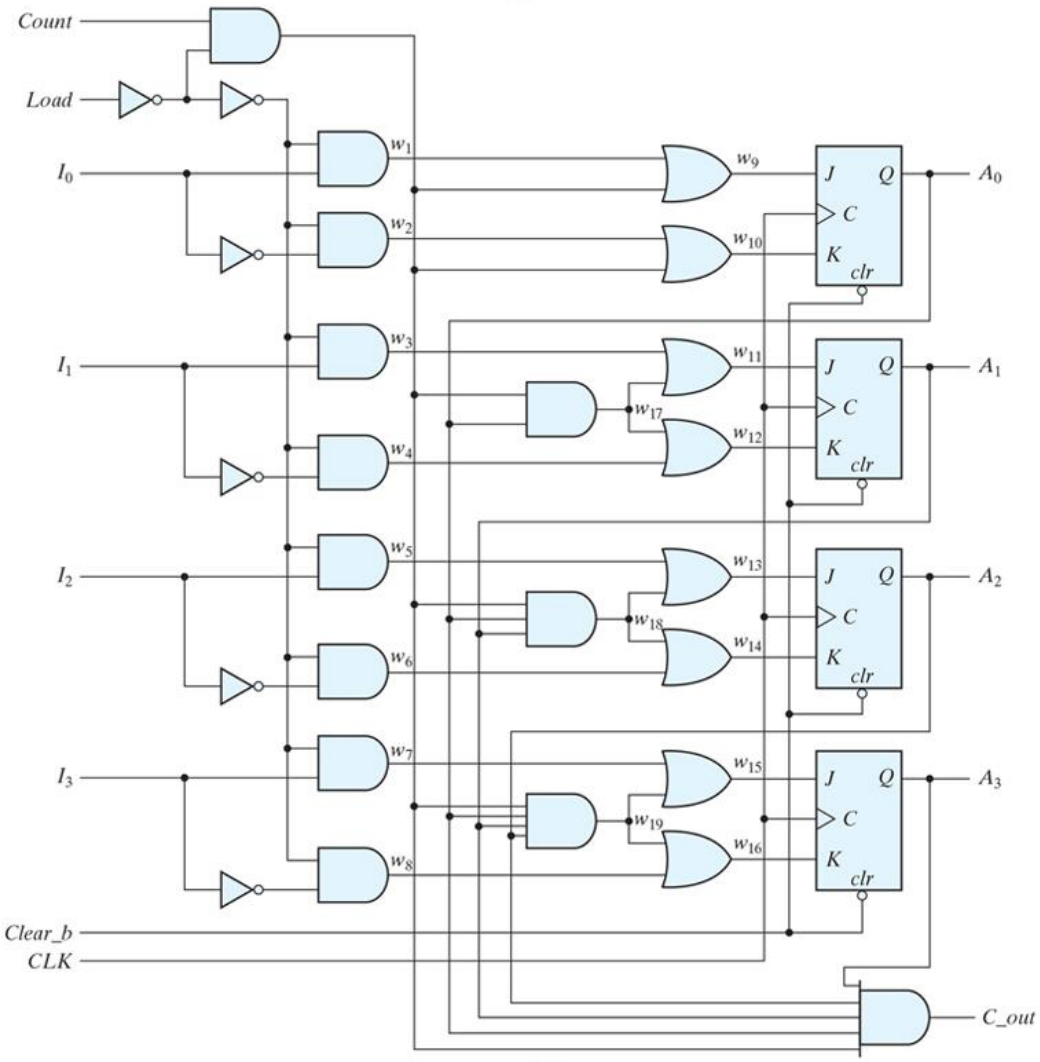
- Design mod-6 counter using the given circuit.
- For a count sequence of 0,1,2,3,4,5



(a) Using the load input



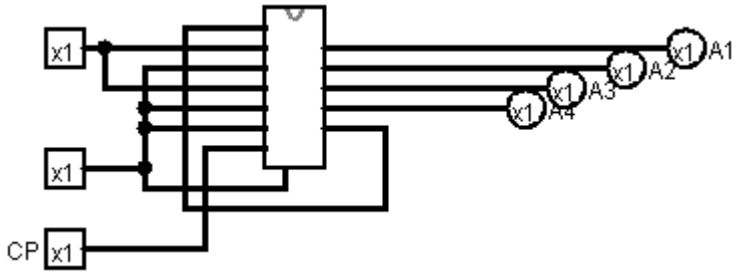
- Reset=1, Count=1, Inputs=0



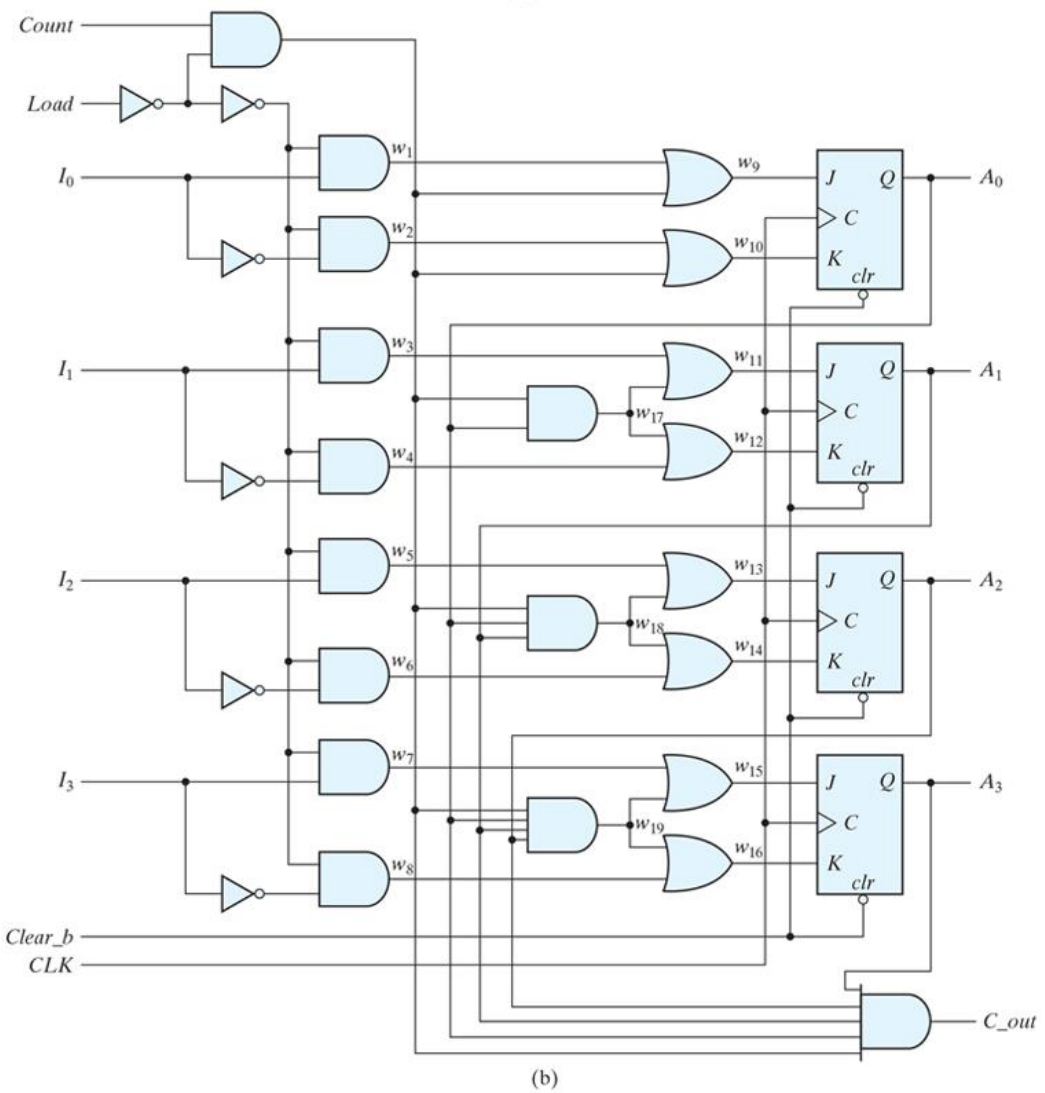
(b)

Example

- Design mod-6 counter using the given circuit.
- For a count sequence of 10,11,12,13,14,15:

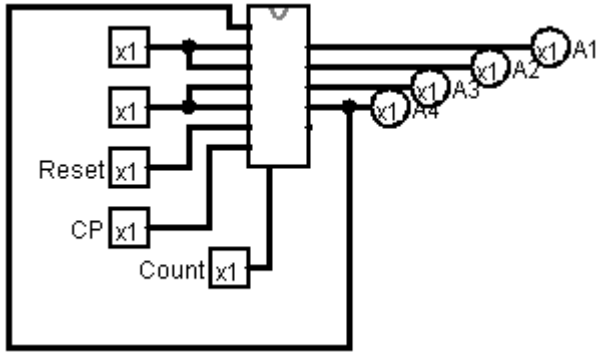


- Count=1, Reset=1, Load= C_out

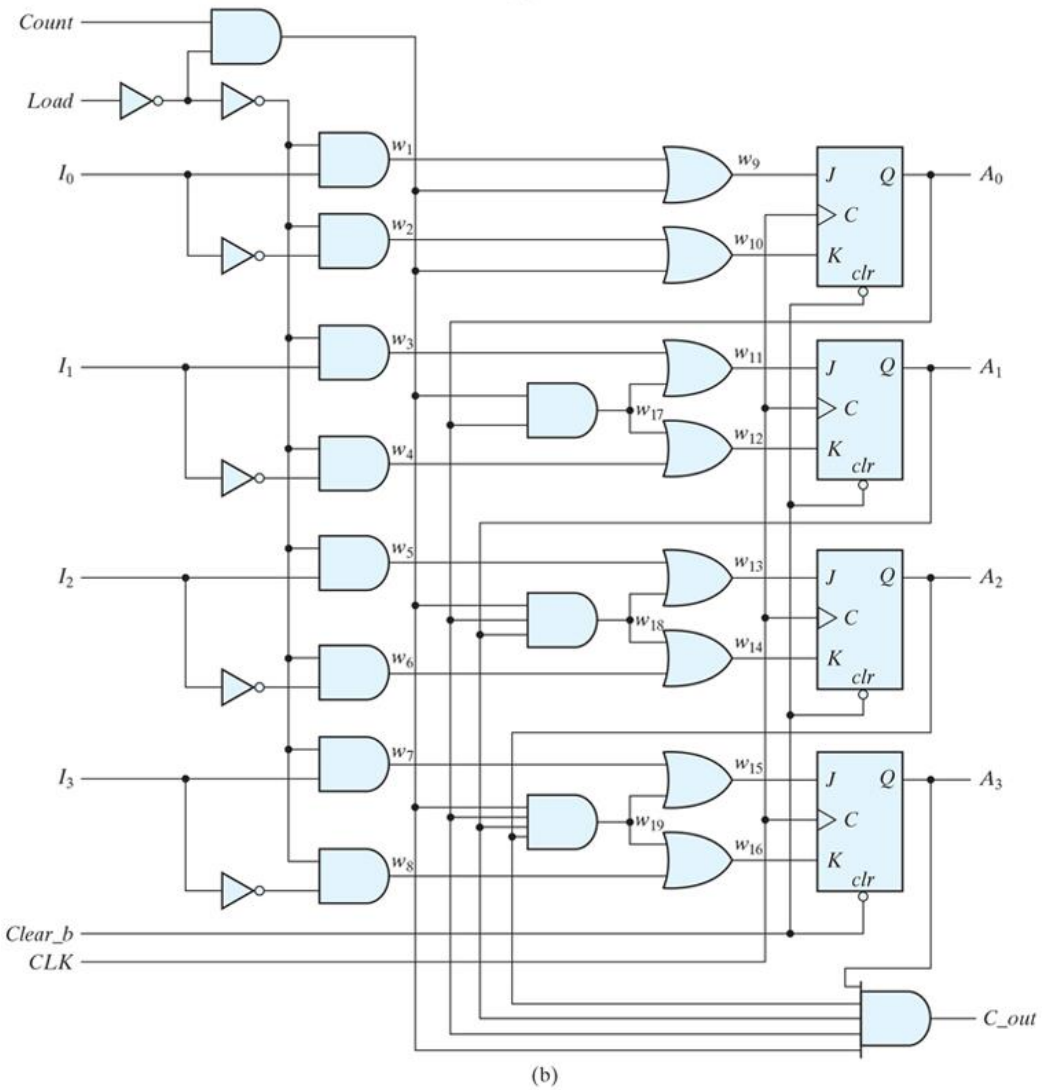


Example

- Design mod-6 counter using the given circuit.
- For a count sequence of 3,4,5,6,7,8:



- Reset=1, Count=1, Inputs=0

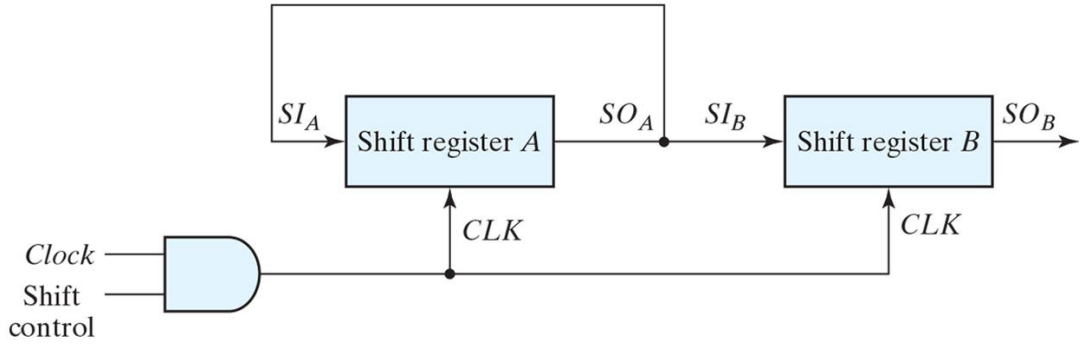


Timing Sequence

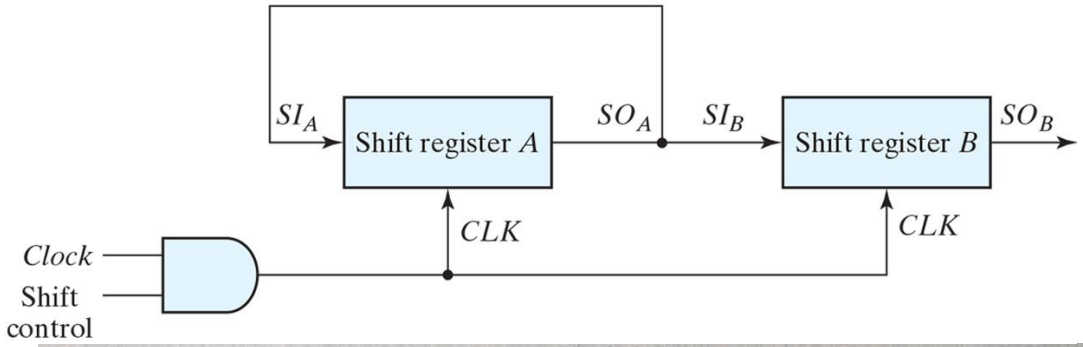
- In a digital system, the control unit generate timing signals to determines the order of (timing of) other units.
- Timing signals can be generated by means of counters and shift registers.

Timing Sequence

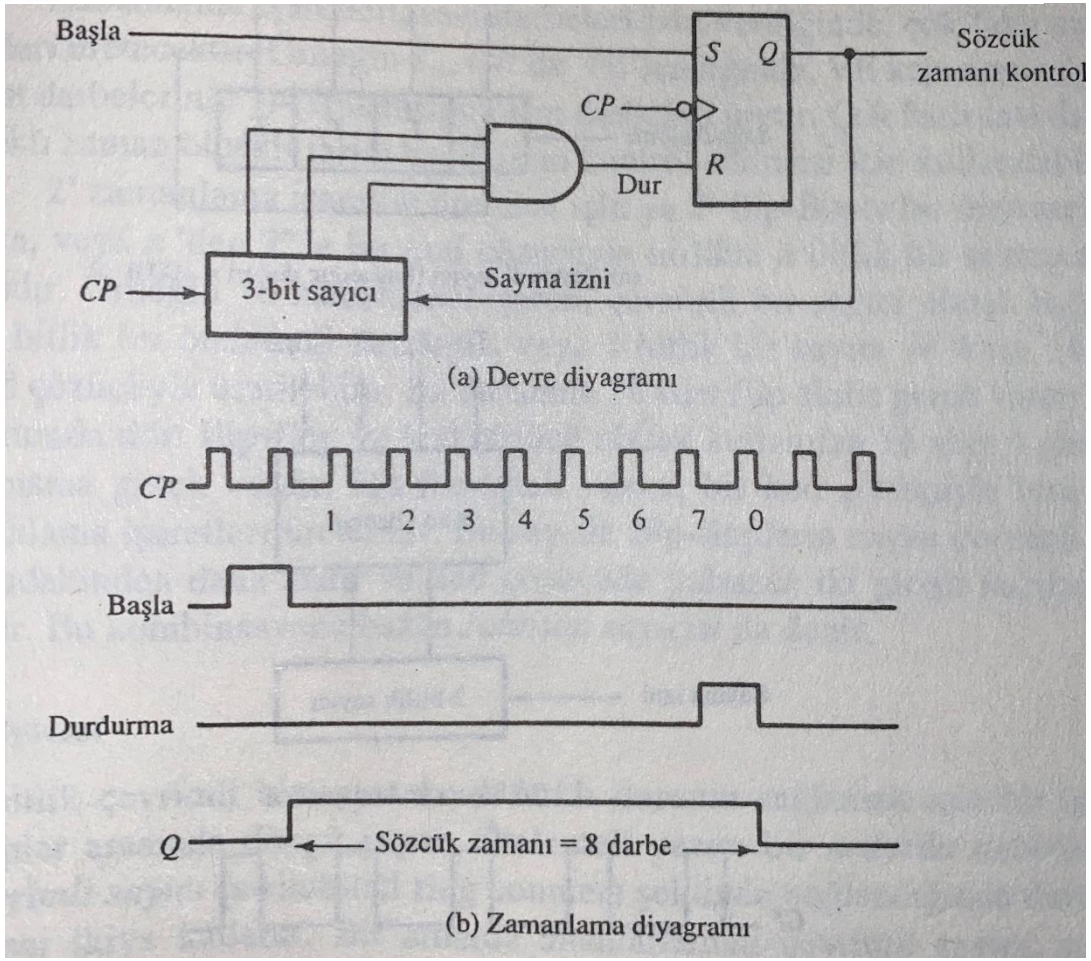
- Generating word time signal:
  - Serial transfer
    - The control unit should generate a word time signal which is equal to the number of bits in a shift register. Word time signal can be generated by means of a counter which counts the number of pulses.



Timing Sequence



- Assume that the word time signal remains constant for 8 clock pulses.
- A counter for this purpose is given in the figure.
  - Initial state of the 3 bit counter: 000,
  - Start signal sets the flip-flop, Q=1.
  - The output of the flip-flop generates the word time control and enables the counter.
  - After the 8 clock pulses are counted, the flip-flop is reset, Q=0
  - Timing diagram for the circuit:



Timing Sequence

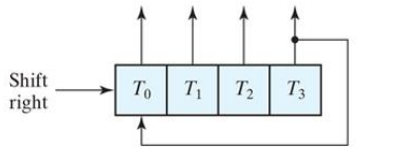
- Timing Signals:
  - Shift Registers
  - Counter with a decoder

- Ring counter: A ring counter is a circular shift register with only one flip-flop being set at any particular time; all others are cleared. The single bit is shifted from one flip-flop to the next to produce the sequence of timing signals.

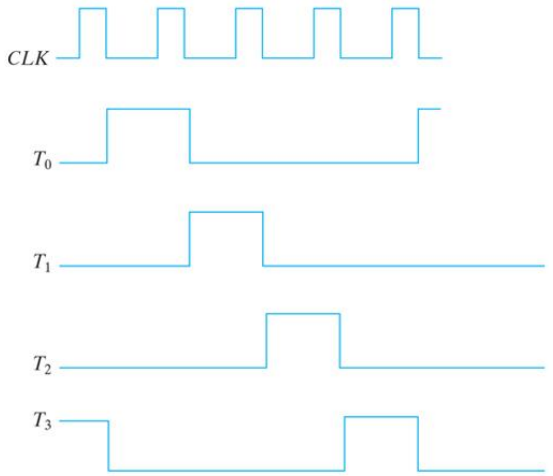
Timing Sequence

- Timing Signals
  - 4 bit shift register which is designed as a ring counter:
    - In order to generate 2^n number of timing signals either 2^n number of shift registers or an n bit counter with and $nx2^n$ decoder.
    - Ex: In order to generate 16 timing signals:
      - 16 bit shift register

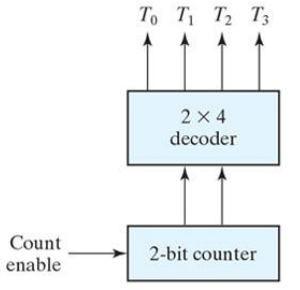
      or
      - 4 bit counter and a 4x16 decoder



(a) Ring-counter (initial value = 1000)



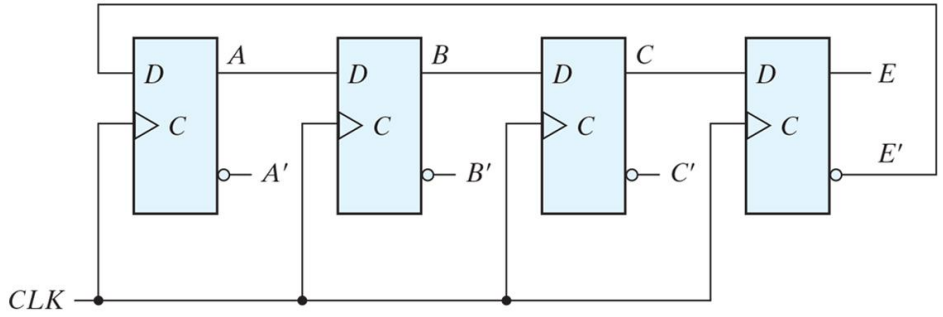
(b) Sequence of four timing signals



(c) Counter and decoder

Timing Sequence

- A k-bit ring counter circulates a single bit among the flip-flops to provide *k* distinguishable states.
- The number of states can be doubled if the shift register is connected as a *switch-tail* ring counter.
- A switch-tail ring counter is a circular shift register with the complemented output of the last flip -flop connected to the input of the first flip-flop.
- Starting from a cleared state the switch-tail ring counter goes through a sequence of eight states.
- In general, a k-bit switch-tail ring counter will go through a sequence of *2k* states.
  - Starting from all 0's each shift operation inserts 1's from the left until the register is filled with all 1's. In the next sequences, 0's are inserted from the left until the register is again filled with all 0's.



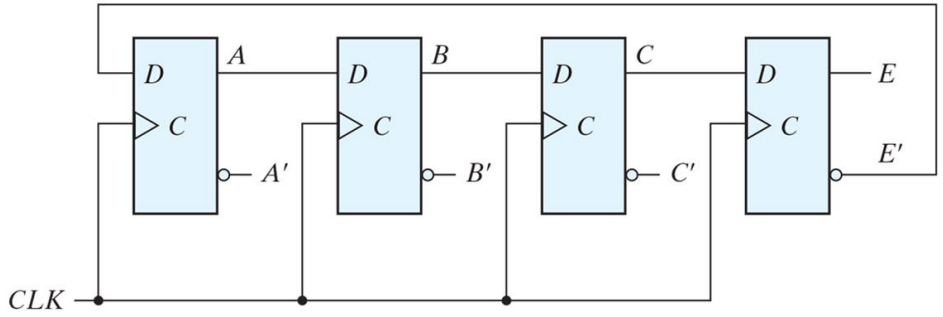
(a) Four-stage switch-tail ring counter

Sequence number	Flip-flop outputs A	B	C	E	AND gate required for output
1	0	0	0	0	$A'E'$
2	1	0	0	0	AB'
3	1	1	0	0	BC'
4	1	1	1	0	CE'
5	1	1	1	1	AE
6	0	1	1	1	$A'B$
7	0	0	1	1	$B'C$
8	0	0	0	1	$C'E$

(b) Count sequence and required decoding

Timing Sequence

- Johnson Counter
  - A Johnson counter is a k-bit switch-tail ring counter with *2k* decoding gates to provide outputs for 2k timing signals.
  - The decoding of a k-bit switch-tail ring counter to obtain 2k timing signals follows a regular pattern.



(a) Four-stage switch-tail ring counter

Sequence number	Flip-flop outputs				AND gate required for output
	A	*B*	*C*	*E*	
1	0	0	0	0	$A'E'$
2	1	0	0	0	AB'
3	1	1	0	0	BC'
4	1	1	1	0	CE'
5	1	1	1	1	AE
6	0	1	1	1	$A'B$
7	0	0	1	1	$B'C$
8	0	0	0	1	$C'E$

(b) Count sequence and required decoding

References

- Morris Mano, Digital Design, 2nd edition.