COEN 312 DIGITAL SYSTEMS DESIGN - LECTURE NOTES

Concordia University

Chapter 6: Registers and Counters

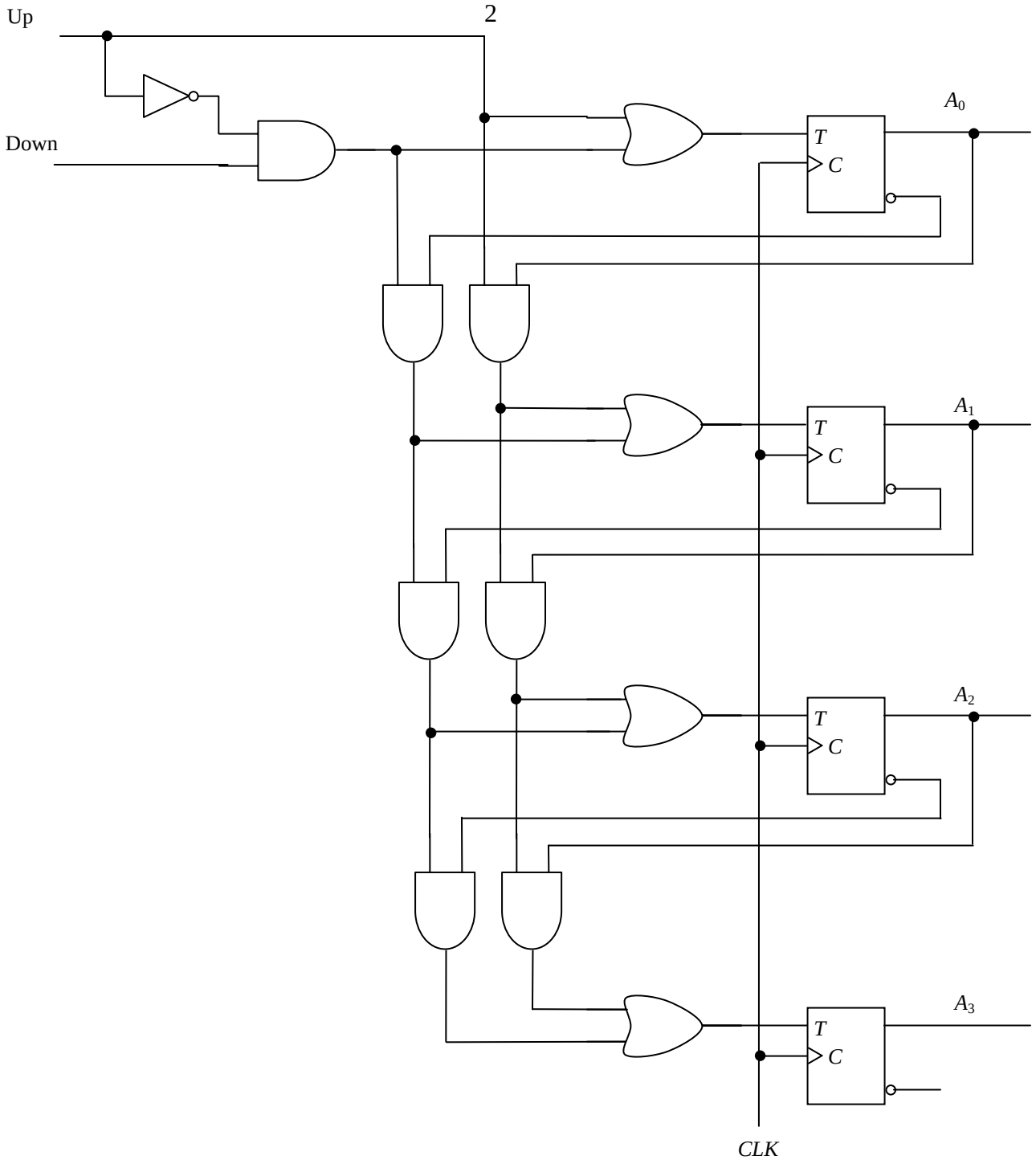
NOTE: For more examples and detailed description of the material in the lecture notes, please refer to the main textbook:
Digital Design 3rd Edition, By Morris Mano, Publisher Prentice Hall, 3rd Edition
All examples used in the lecture notes are from the above reference.

Up-Down Binary Counters

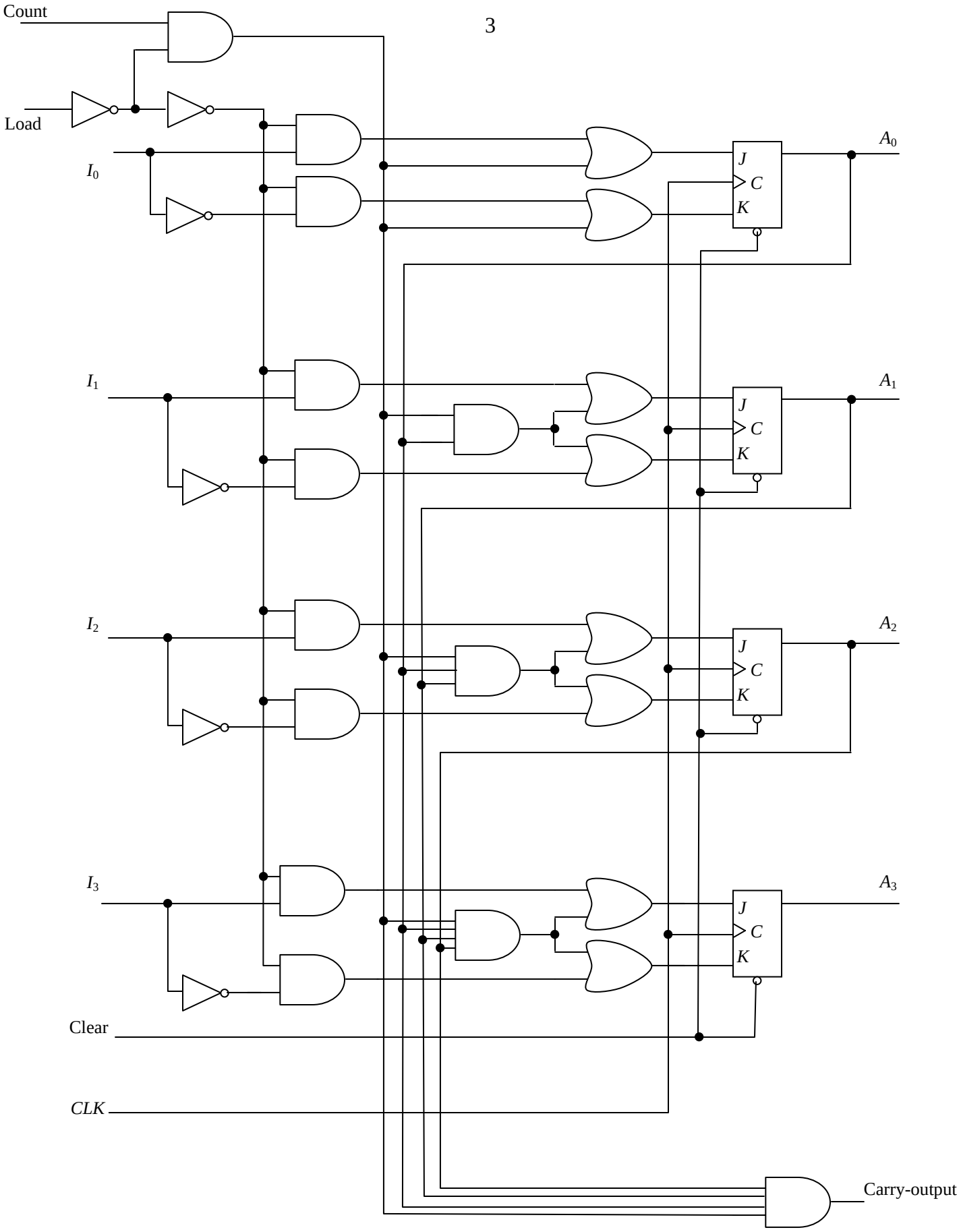
- The state table of the four-bit up counter with *T* flip-flops is given below.

	Present State				Next State				Flip-Flop Inputs			
No.	Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	T_3	T_2	T_1	T_0
0	0	0	0	0	0	0	0	1	0	0	0	1
1	0	0	0	1	0	0	1	0	0	0	1	1
2	0	0	1	0	0	0	1	1	0	0	0	1
3	0	0	1	1	0	1	0	0	0	1	1	1
4	0	1	0	0	0	1	0	1	0	0	0	1
5	0	1	0	1	0	1	1	0	0	0	1	1
6	0	1	1	0	0	1	1	1	0	0	0	1
7	0	1	1	1	1	0	0	0	1	1	1	1
8	1	0	0	0	1	0	0	1	0	0	0	1
9	1	0	0	1	1	0	1	0	0	0	1	1
10	1	0	1	0	1	0	1	1	0	0	0	1
11	1	0	1	1	1	1	0	0	0	1	1	1
12	1	1	0	0	1	1	0	1	0	0	0	1
13	1	1	0	1	1	1	1	0	0	0	1	1
14	1	1	1	0	1	1	1	1	0	0	0	1
15	1	1	1	1	0	0	0	0	1	1	1	1

- This table shows that a bit changes when all of its less significant bits are equal to one (this is what one intuitively expects in an up counter).
- Similarly, one can find the state table of a down counter, where a bit changes when all of its less significant bits are equal to zero.
- One can design an up-down counter with an up and down control inputs, as shown in the following figure.



- It can be easily verified that when the external input “Up” is equal to 1, then no matter what the other external input, i.e. “Down” is, the circuit will operate as an up-counter.
- When the external input “Down” is equal to 1 and the external input “Up” is equal to zero, then the circuit operates as a down-counter.
- If both external inputs “Up” and “Down” are equal to zero, then the output of the flip-flops remain unchanged.
- A binary counter with parallel load is shown in the following figure.


Count
Load
I_0
I_1
I_2
I_3
J
C
K
A_0
A_1
A_2
A_3
Clear
CLK
Carry-output

- When the external input "Load" is equal to 1, then no matter what "Count" is, the input lines I_0, I_1, I_2 and I_3 will appear in the output of the flip-flops A_0, A_1, A_2 and A_3, respectively.
- When, on the other hand, "Count" is equal to 1 and "Load" is equal to zero, then the circuit will operate as an up-counter.
- When both external inputs "Load" and "Count" are equal to zero, then the flip-flop outputs remain unchanged.

BCD Counter

- A BCD up counter is a digital system that counts from 0 to 9 as a decimal digit, and returns back to 0 to repeat the counting.
- A BCD counter can be built in a way similar to the binary counter.
- The state table can be obtained as shown below.

	Present State				Next State				Output	Flip-Flop Inputs			
No.	Q_8	Q_4	Q_2	Q_1	Q_8	Q_4	Q_2	Q_1	y	T_8	T_4	T_2	T_1
0	0	0	0	0	0	0	0	1	0	0	0	0	1
1	0	0	0	1	0	0	1	0	0	0	0	1	1
2	0	0	1	0	0	0	1	1	0	0	0	0	1
3	0	0	1	1	0	1	0	0	0	0	1	1	1
4	0	1	0	0	0	1	0	1	0	0	0	0	1
5	0	1	0	1	0	1	1	0	0	0	0	1	1
6	0	1	1	0	0	1	1	1	0	0	0	0	1
7	0	1	1	1	1	0	0	0	0	1	1	1	1
8	1	0	0	0	1	0	0	1	0	0	0	0	1
9	1	0	0	1	0	0	0	0	1	1	0	0	1

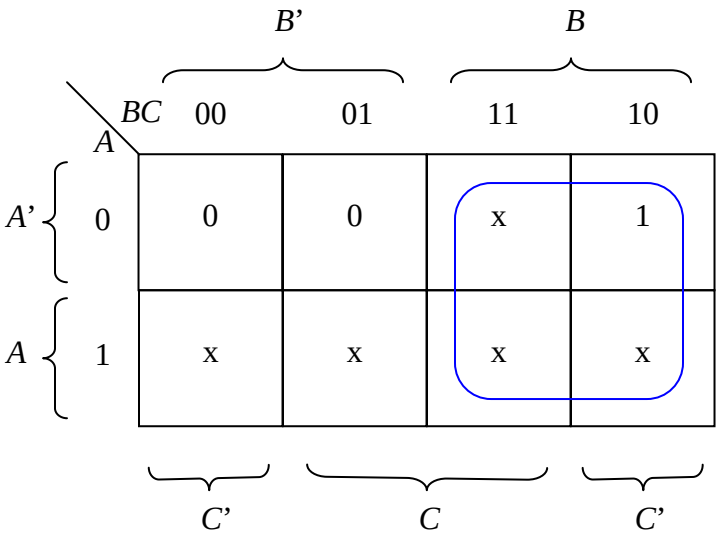
- The output is, in fact, a carry out, which indicates that the numbers 0 to 9 have been counted once and a new set of BCD codes are being counted.

Counters with unused States

- A sequential circuit with m states, where $2^{n-1} < m < 2^n$, will require n flip-flops, and as a result, the circuit will have $2^n - m$ unused states.
- For example, a counter with a prespecified sequence 000-001-010-100-101-110-000 has two unused states 011 and 111, as shown in the following state table.

Present State			Next State			Flip-Flop Inputs					
A	***B***	***C***	***A***	***B***	***C***	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	1	0	0	1	x	x	1	0	x
1	0	0	1	0	1	x	0	0	x	1	x
1	0	1	1	1	0	x	0	1	x	x	1
1	1	0	0	0	0	x	1	x	1	0	x

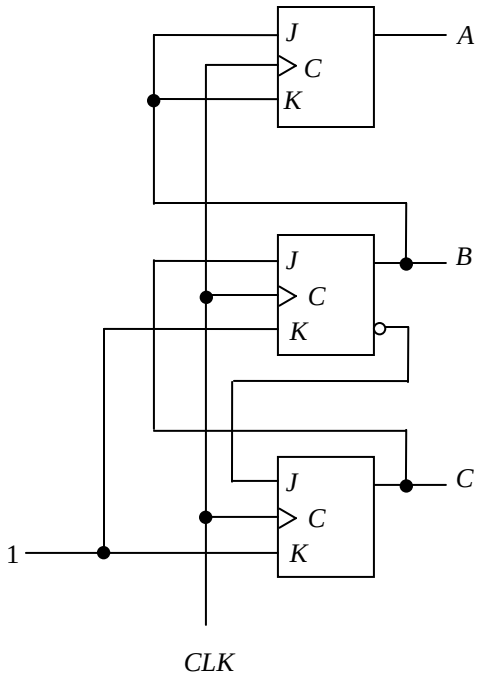
- To find the simplified flip-flop input expressions using K-map, the terms corresponding to the unused states 011 and 111 are considered don't-care conditions.
- For instance, the K-map corresponding to J_A is as follows.



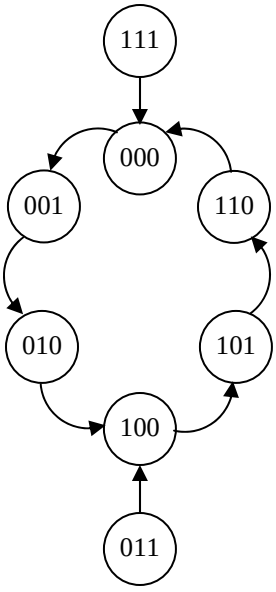
- The simplified Boolean expressions for the flip-flop inputs are as follows:

$$J_A = B \qquad K_A = B$$
$$J_B = C \qquad K_B = 1$$
$$J_C = B' \qquad K_C = 1$$

- This leads to the following logic diagram for this system:

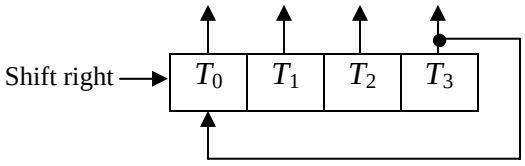


- From the logic circuit, one can find out what happens if the circuit starts from an unused state.
- This can be easily figured out from the state diagram which is obtained below.

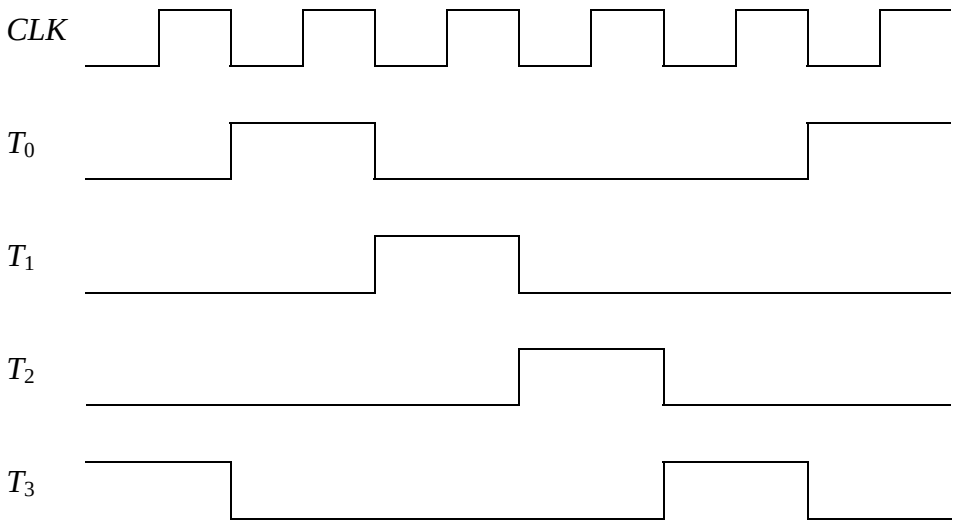


Ring Counter

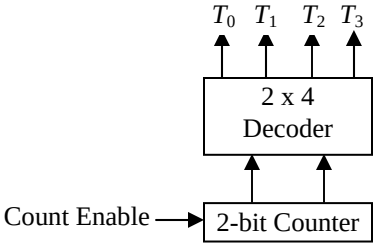
- A ring counter is a register with circular shift, where only one flip-flop output is equal to one, and all other ones are zero.
- A ring counter can be used to generate the timing signal for a circuit, where, for example, only one component of the system should get a set input and all other ones should get a zero input.
- An example of a 4-bit ring counter is given below.



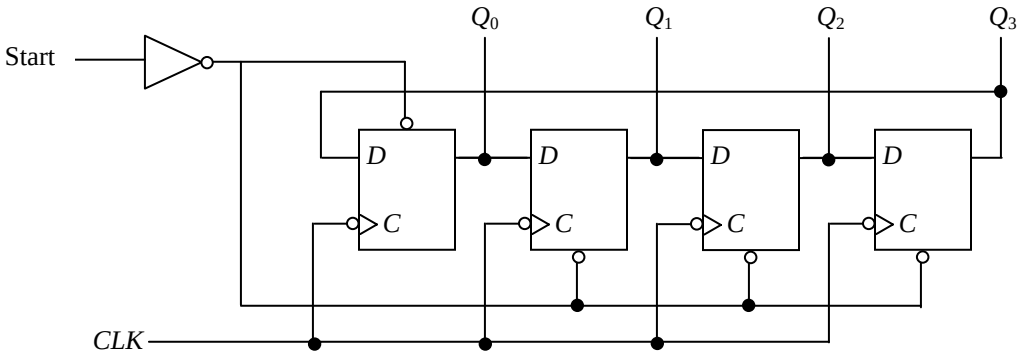
- A typical timing diagram of the above counter is as follows:



- A ring counter with 2^n bits (or 2^n timing signals) can be implemented using a *n*-bit binary counter and a *n*-to-2^n-line decoder.
- For example, the above ring counter can be implemented as follows:



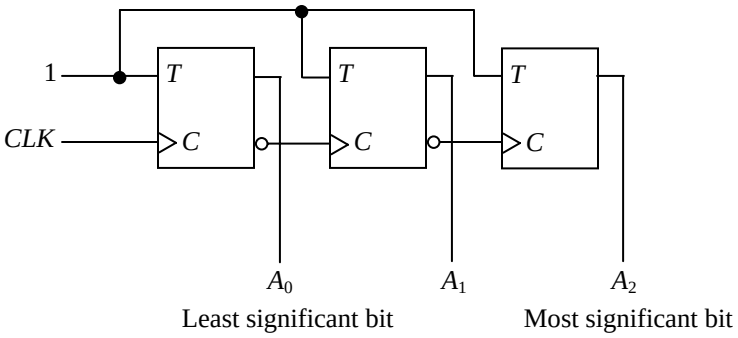
- The following circuit is a 4-bit ring counter with a Start line.



- When the Start line is equal to one, the left-most flip-flop is set and all other flip-flops are reset, as required for a ring counter.

Asynchronous Counter or Ripple Counter

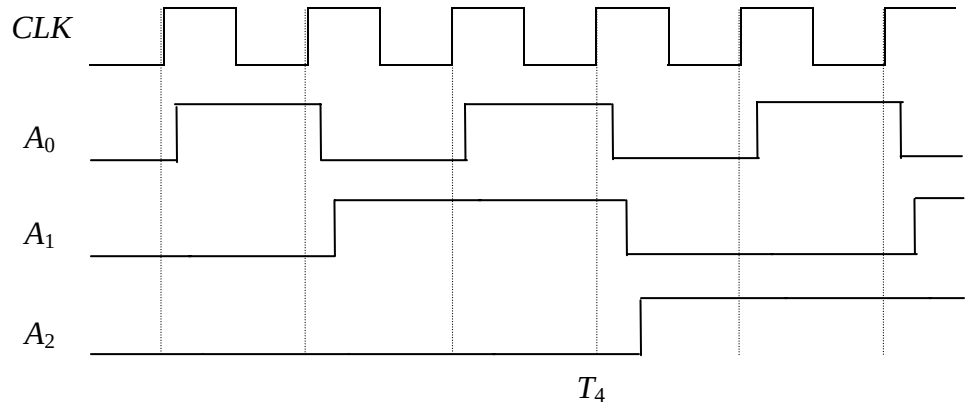
- Example: A three-bit ripple up-counter with *T* flip-flops is given below.



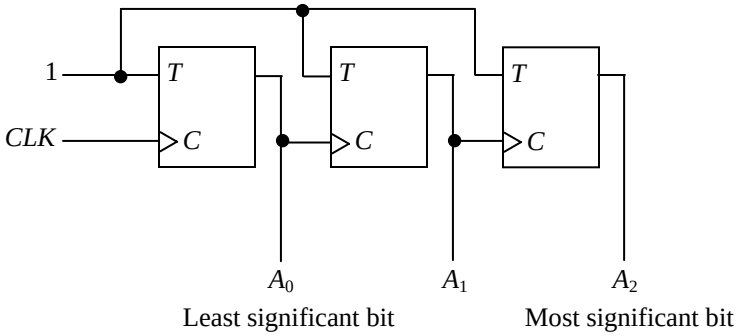
- A_0 toggles once after each clock edge.
- The counting sequence is 0-1-2-3-4-5-6-7-0-…, as shown below:

A_2	A_1	A_0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

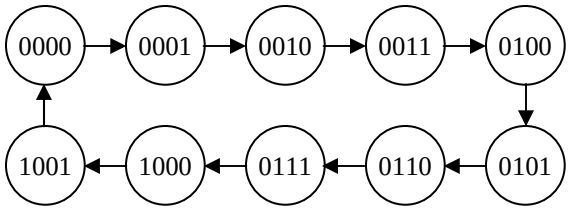
- The timing diagram of the ripple counter given above, is as follows:



- As it can be seen from the above diagram, in reality because of the propagation delay, the counting will have some transition problem for a short period of time after the clock edge. For instance, right after T_4, we will have 011-010-000-100 (instead of 000-100).
- One can also use a *D* flip-flop to build a ripple counter, by connecting the input of each flip-flop to the *Q*' of the same flip-flop.
- A three bit ripple down-counter with *T* flip-flops is given below.



BCD Ripple Counter



- It can be easily verified that:
- Q_1 changes after each clock cycle.
- Q_2 complements each time Q_1 changes from 1 to 0 while Q_8 is 0.
- Q_2 remains at 0 while Q_8 is 1.
- Q_4 complements each time Q_2 changes from 1 to 0.
- Q_8 remains at 0 while Q_2 or Q_4 is 0.
- When Q_2 and Q_4 both become equal to 1, then Q_8 complements as soon as Q_1 changes from 1 to 0.
- Q_8 changes to 0 next time that Q_1 changes from 1 to 0.

Reference:

[1] Digital Design 3rd Edition, By Morris Mano, Publisher Prentice Hall, 3rd Edition.