

Chapter 7

Registers & Counters

© Luis Entrena, Celia López,
Mario García, Enrique San Millán
Universidad Carlos III de Madrid

Contents

1. Registers

- Registers serial input, parallel-serial output
- Registers parallel output, parallel-serial output
- Universal shift register

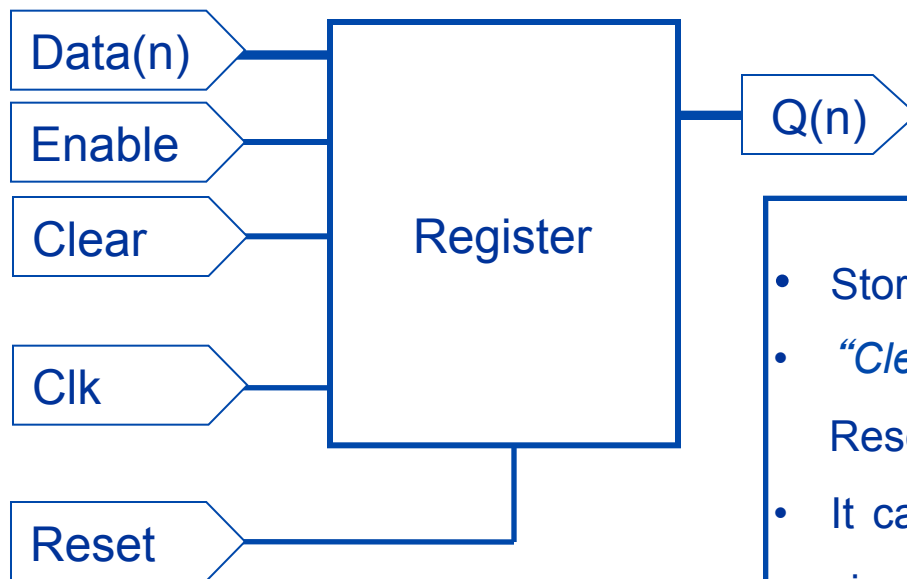
2. Counters

- **Synchronous Counters**
 - Concept of synchronous counter.
 - Synthesis as a Finite State Machine with T-flip-flops.
 - Up-down counter
 - Counters with Load input, Carry-In, Enable and Carry-Out.
 - Applications with synchronous counters: sequencers.
- **Counters based on shift registers**
 - Ring-counter. Johnson counter.

Register

“Digital circuit with two basic functions: data storing and data movement” (Floyd)

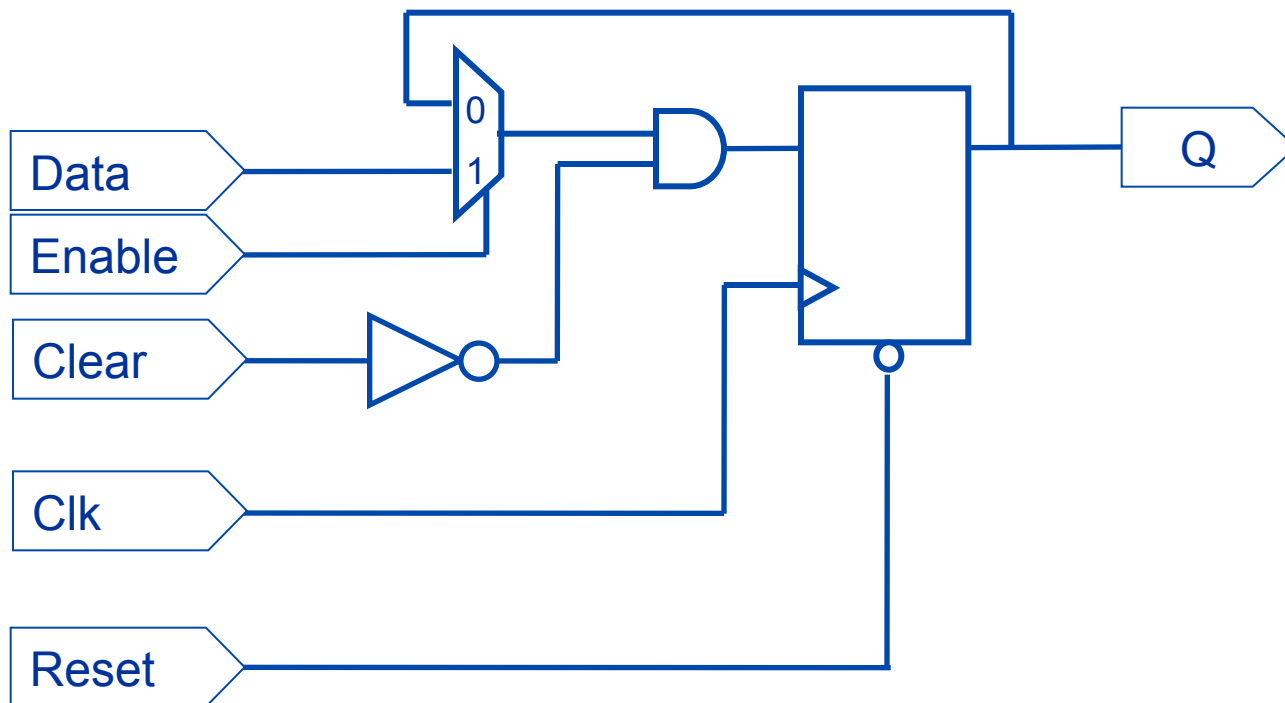
It is a collection of two or more D-type flip-flops with a common input. It is used for storing a number of related bits, such as a byte (8 bits).



- Stores data in the active edge of Clk
- “Cleans” contents *in the active level of* Reset
- It can have synchronous Enable and Clear signals

Register (1 bit)

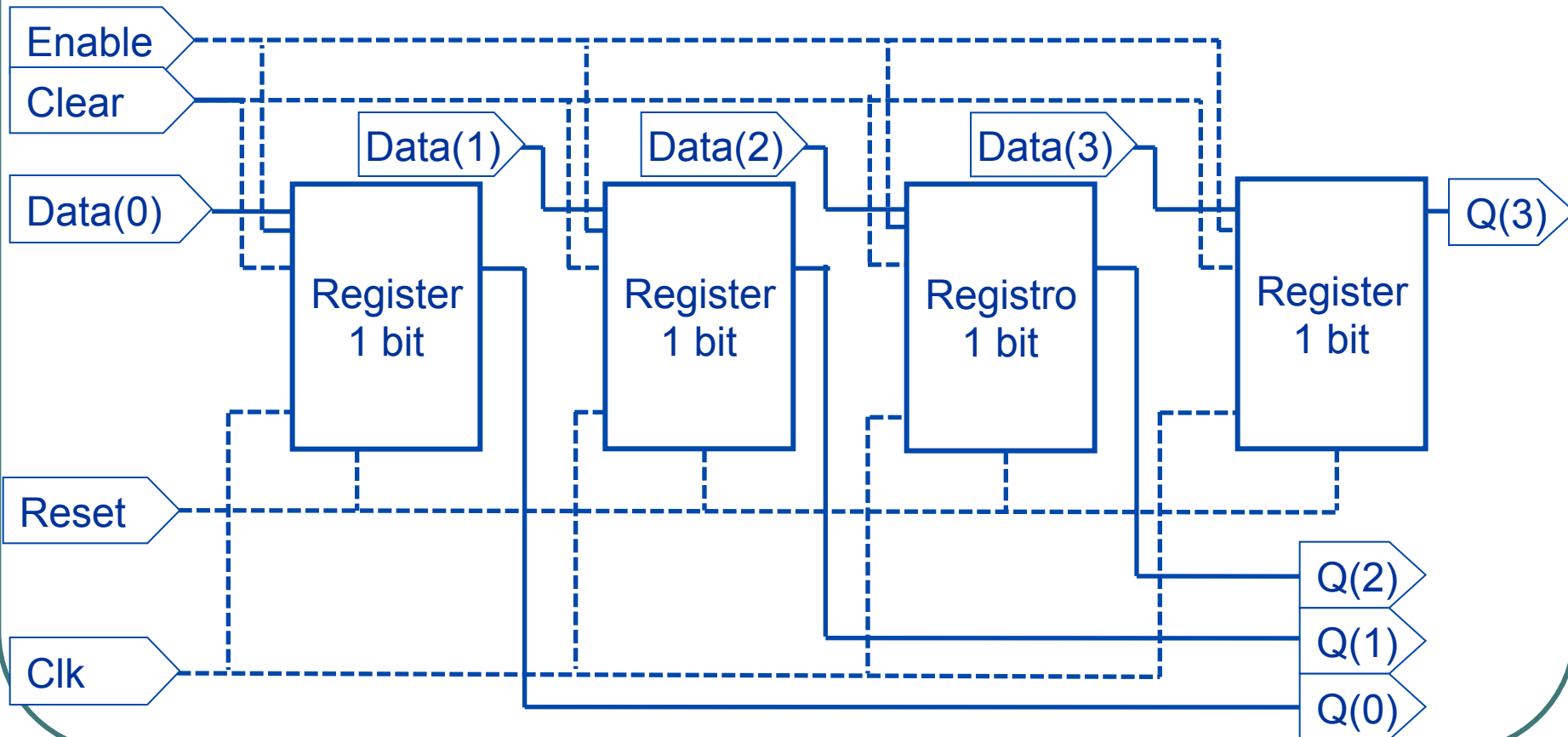
Schematic



Register (4 bits)

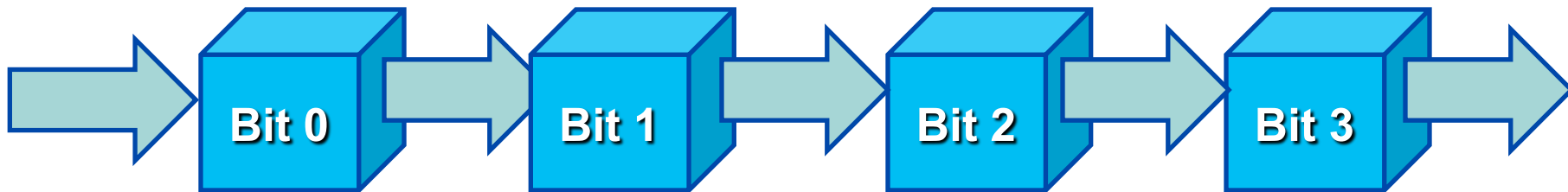
Parallel input / Parallel output

Schematic



Shift Register

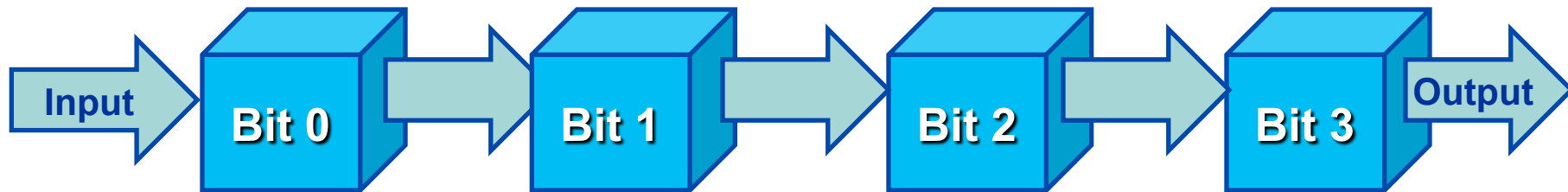
It is a register for storing and shifting the information



- Registers data in active edges of Clk. It also shifts the bits.
- “Cleans” contents *in the active level of* Reset
- It can have synchronous Enable and Clear signals

Shift Register

Serial input and output

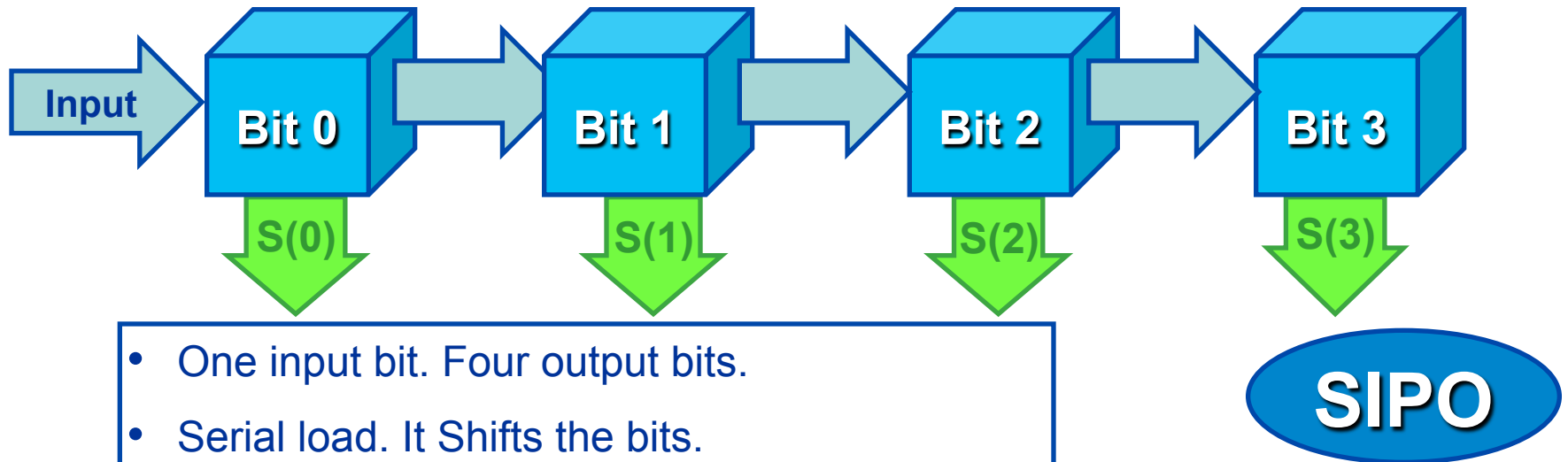


- One input bit . One output bit.
- Serial load. It shifts the bits.
- 4 Clk cycles for loading a data.
- 4 Clk cycles for reading a data.

SISO

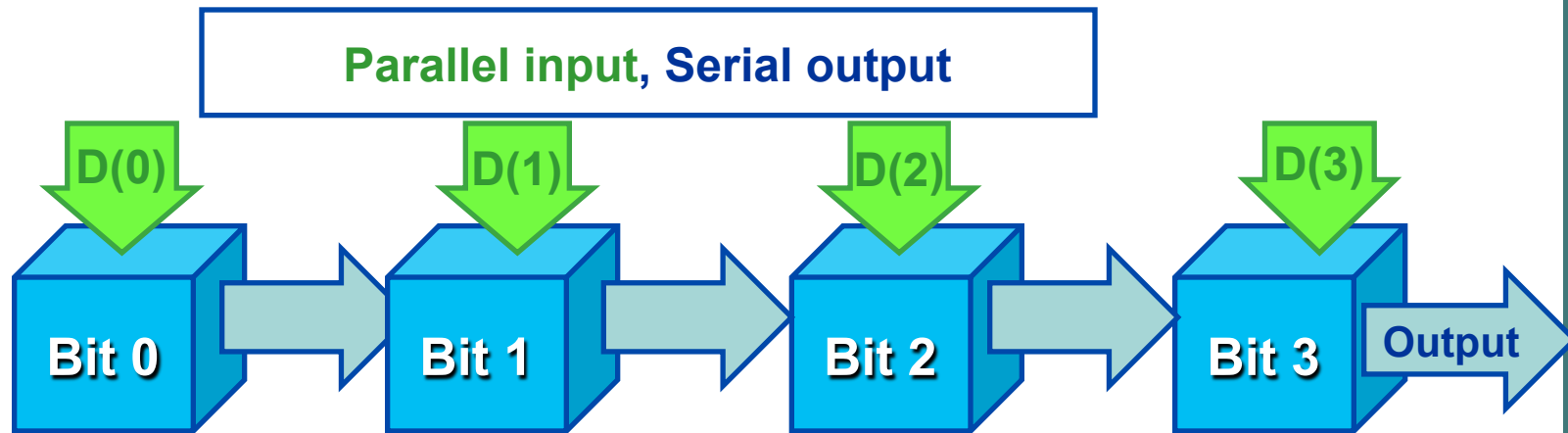
Shift Register

Serial input, Parallel output



- One input bit. Four output bits.
- Serial load. It Shifts the bits.
- Four Clk cycles for loading a data.
- One Clk cycle for reading a data.

Shift Register



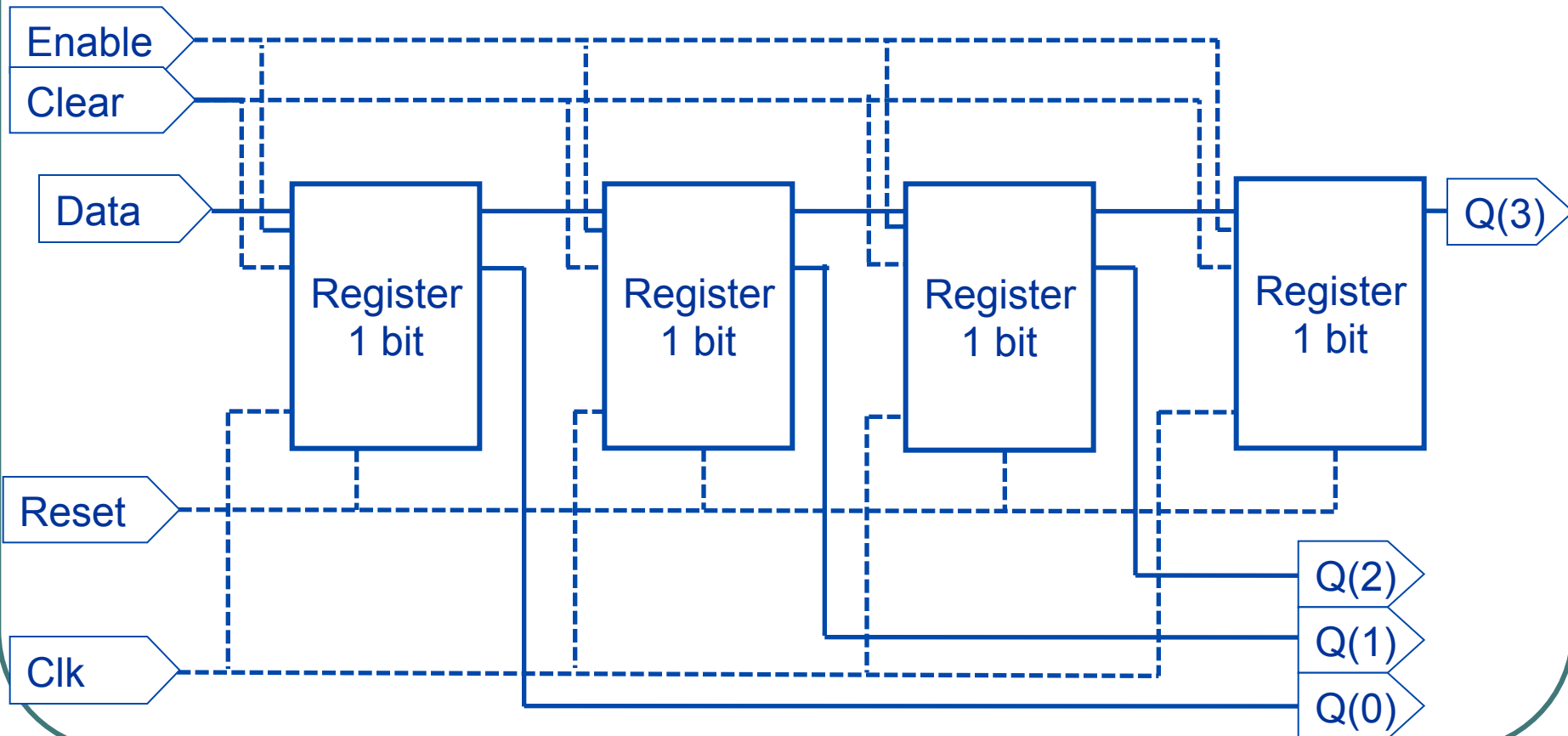
- Four input bits. One output bit.
- Parallel load. Serial output. It shifts the bits.
- 1 Clk cycle for storing a data.
- 4 Clk cycles for reading a data.

PISO

Shift Register

Serial input/ Parallel-Serial Output

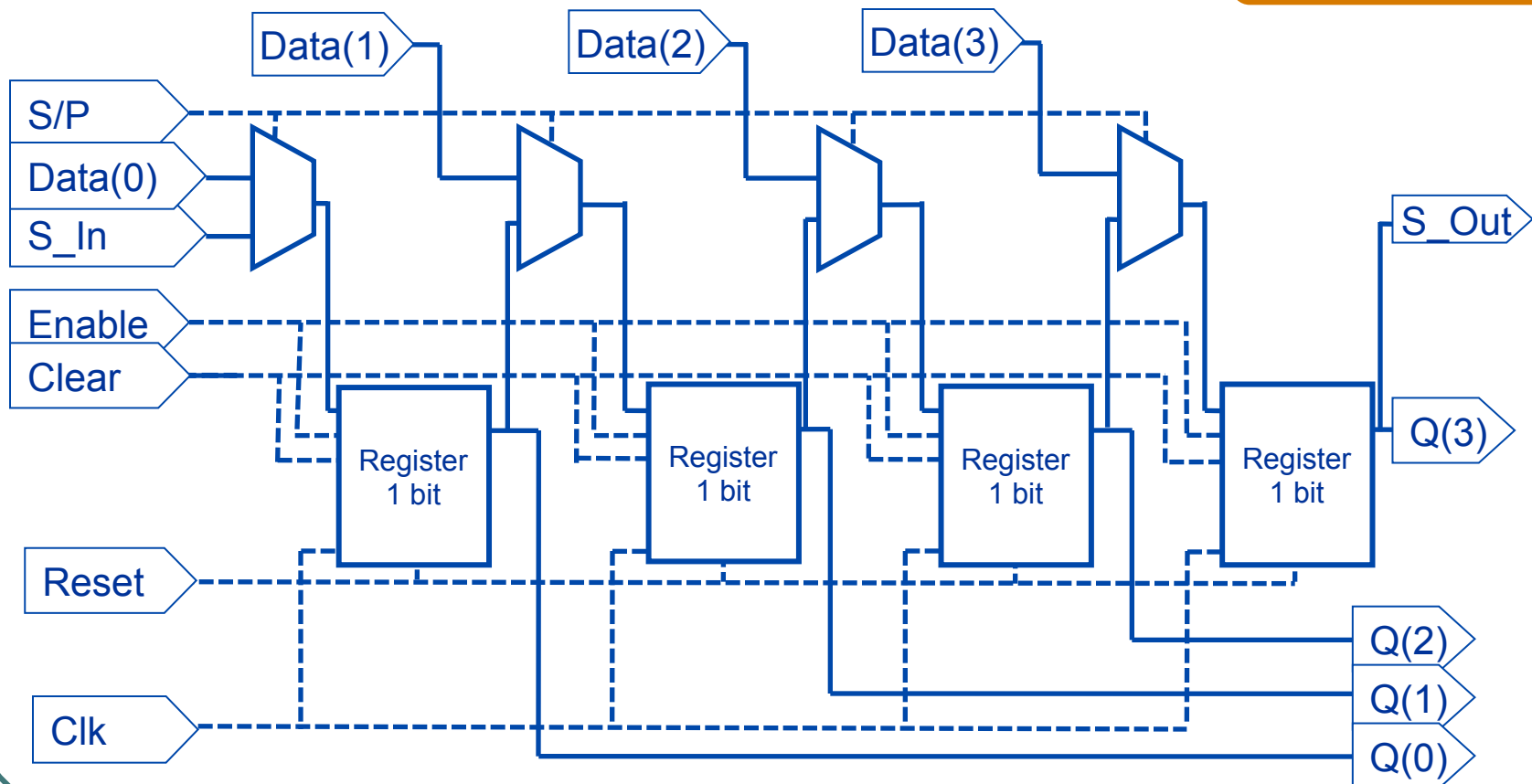
Schematic



Shift Register

Serial-parallel input / Serial-parallel output

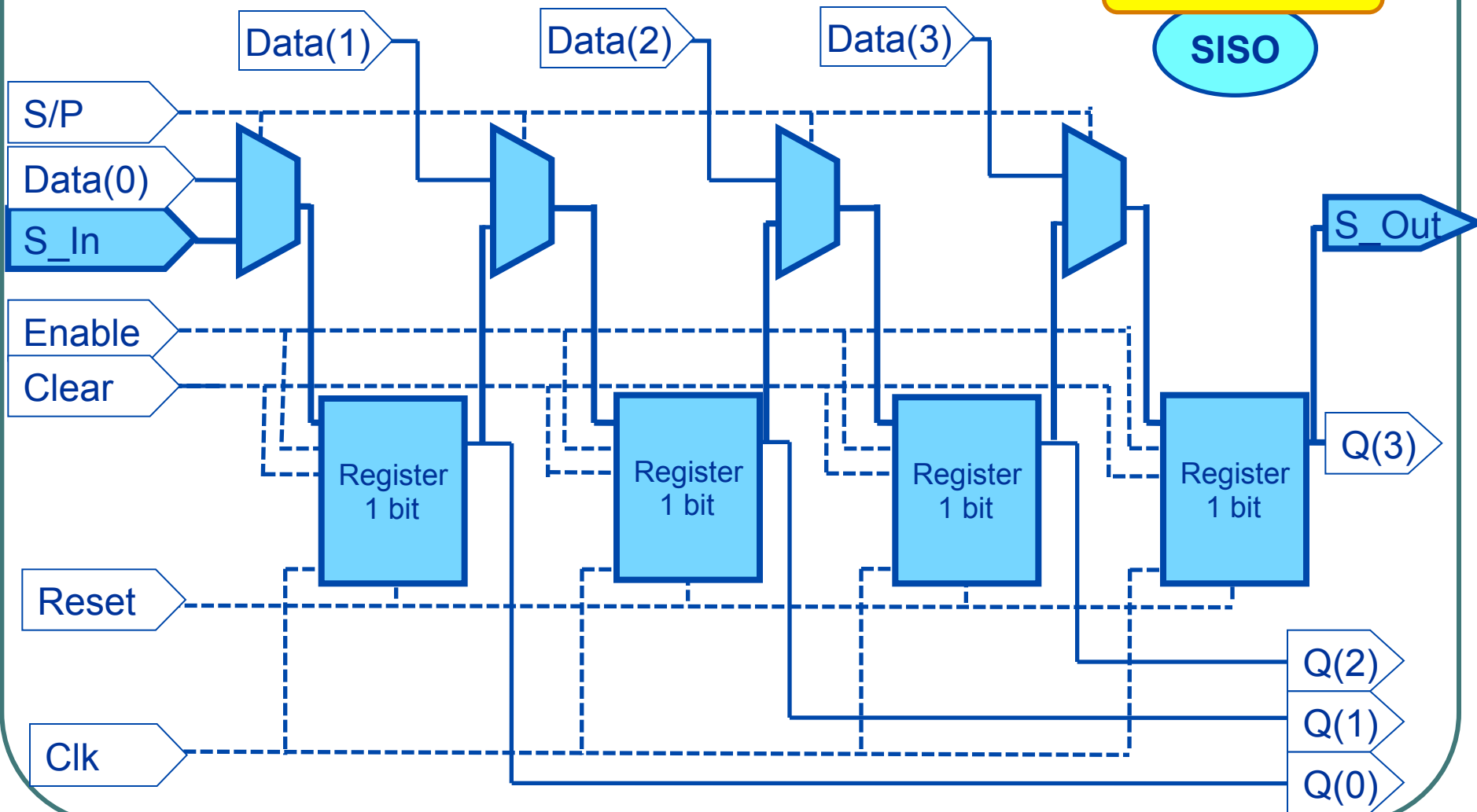
Schematic



Shift Register

Schematic

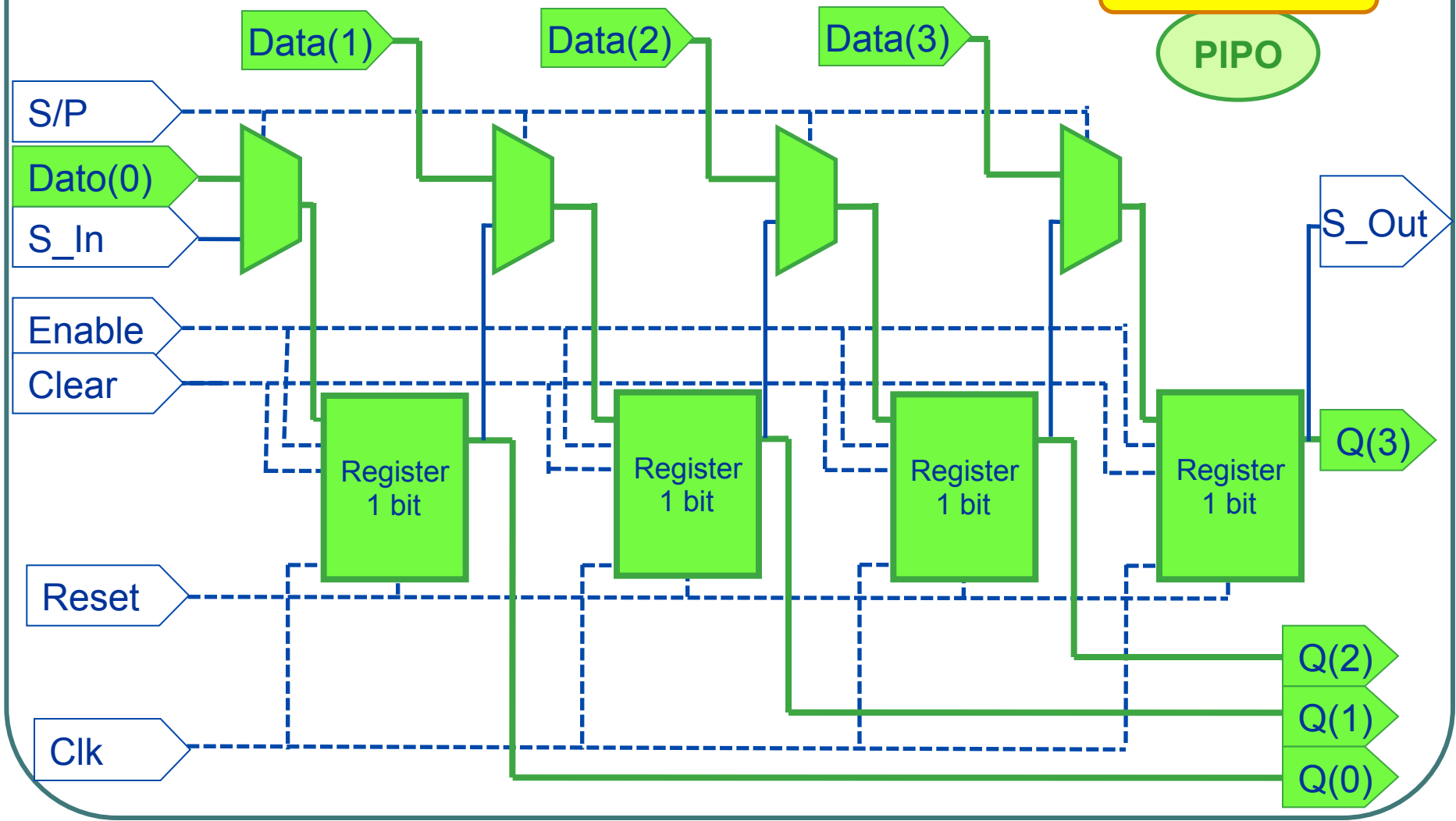
SISO



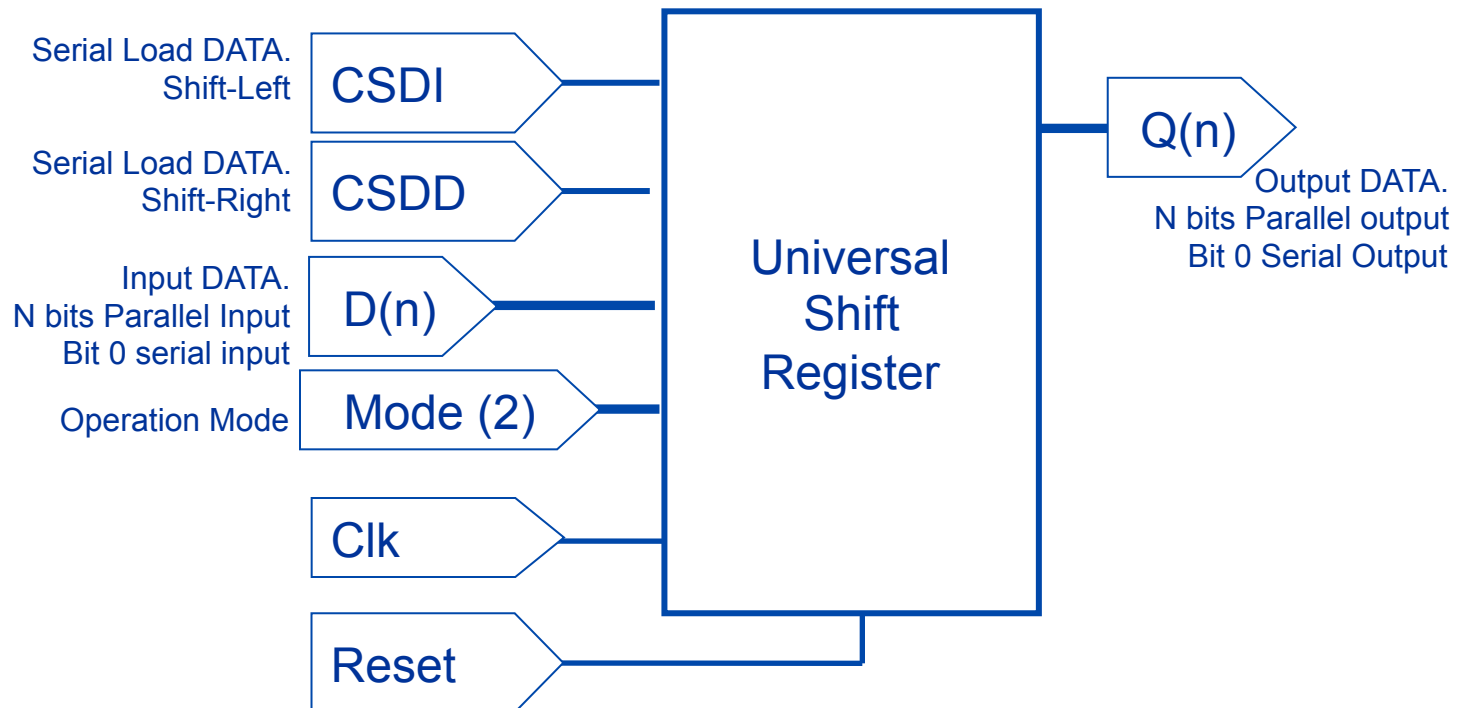
Shift Register

Schematic

PIPO



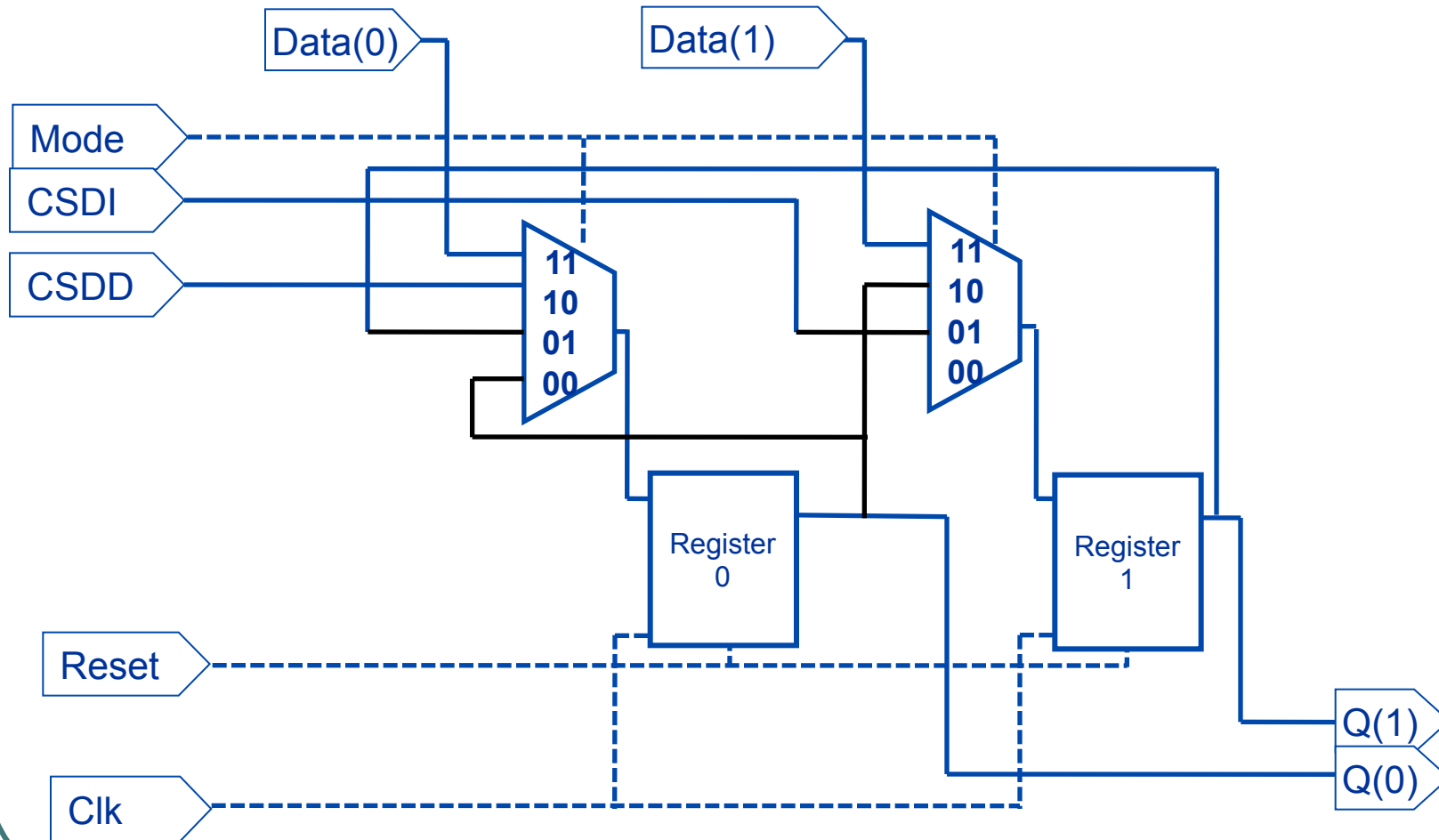
Universal Shift Register



Shift register with serial / parallel input. Data can be left or right shifted.

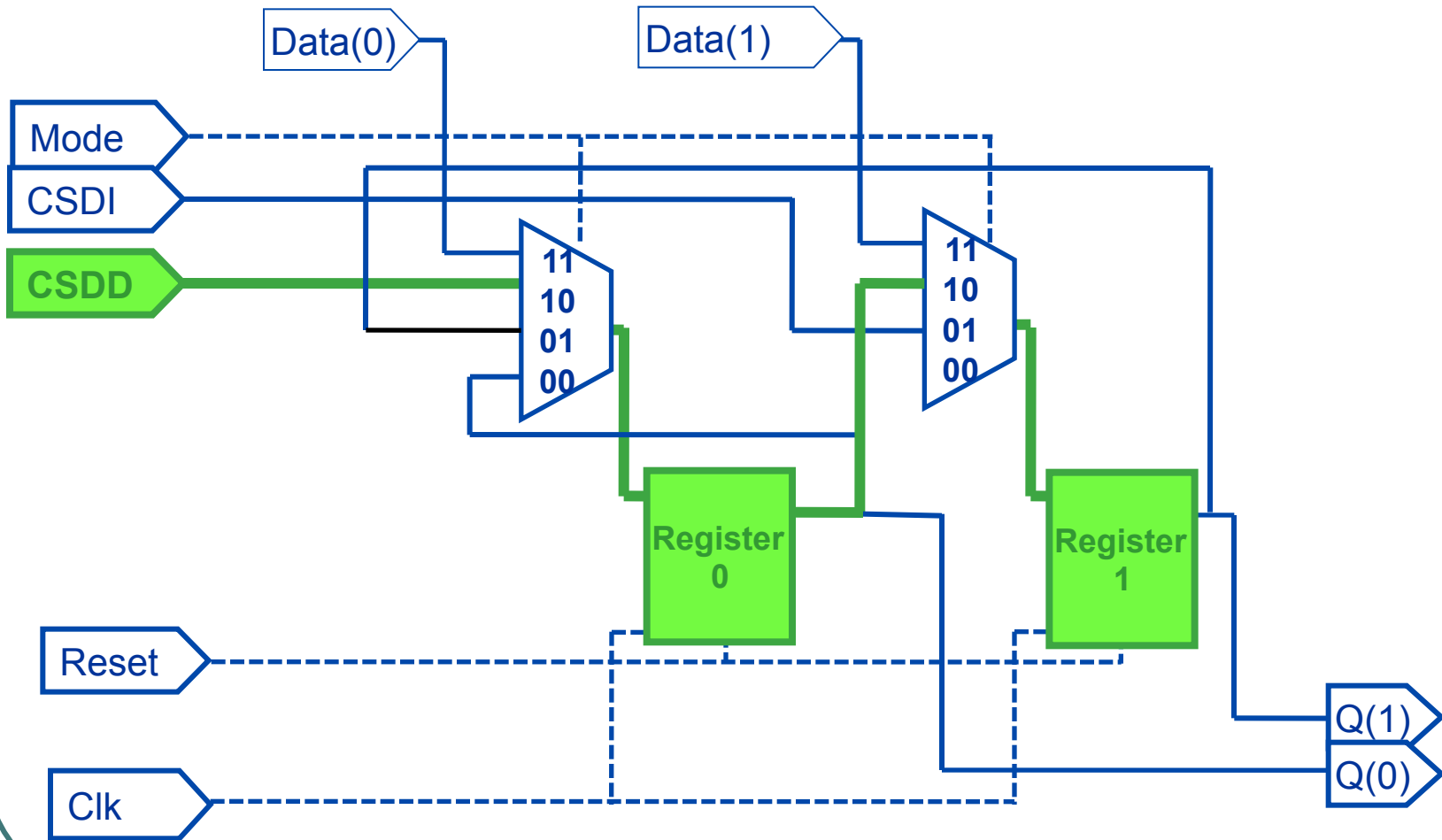
Universal Shift Register

Schematic



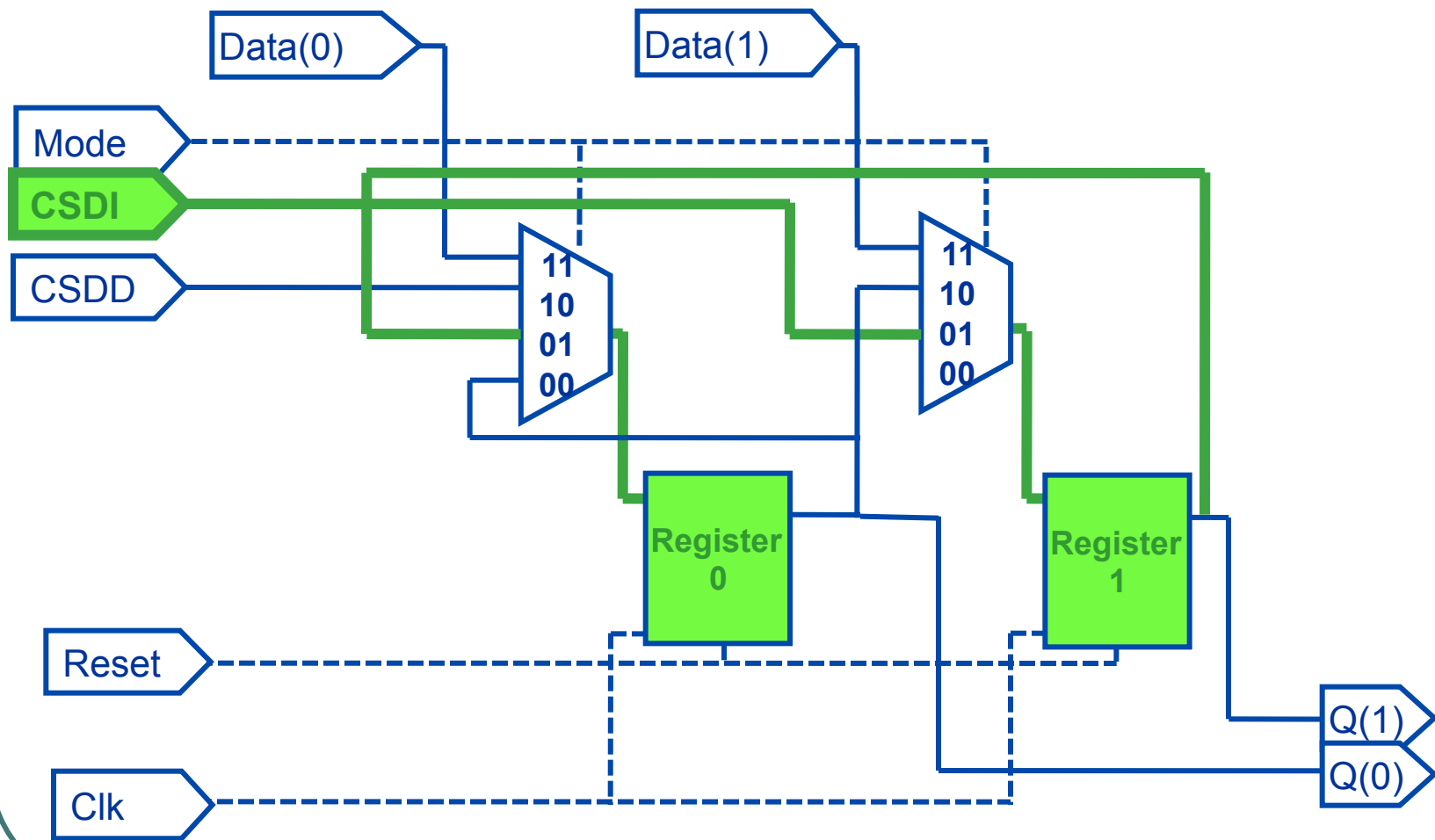
Universal Shift Register

Schematic



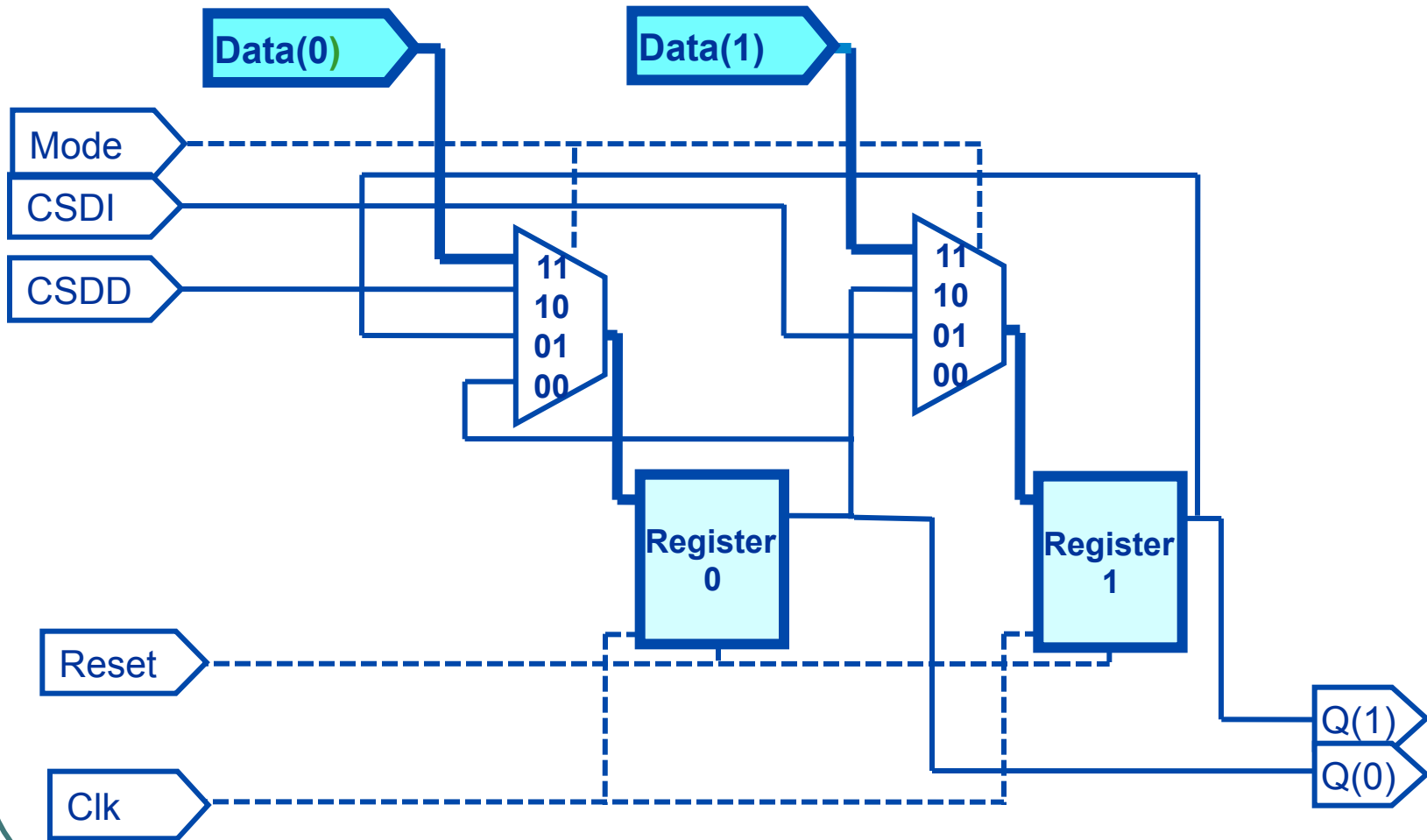
Universal Shift Register

Schematic



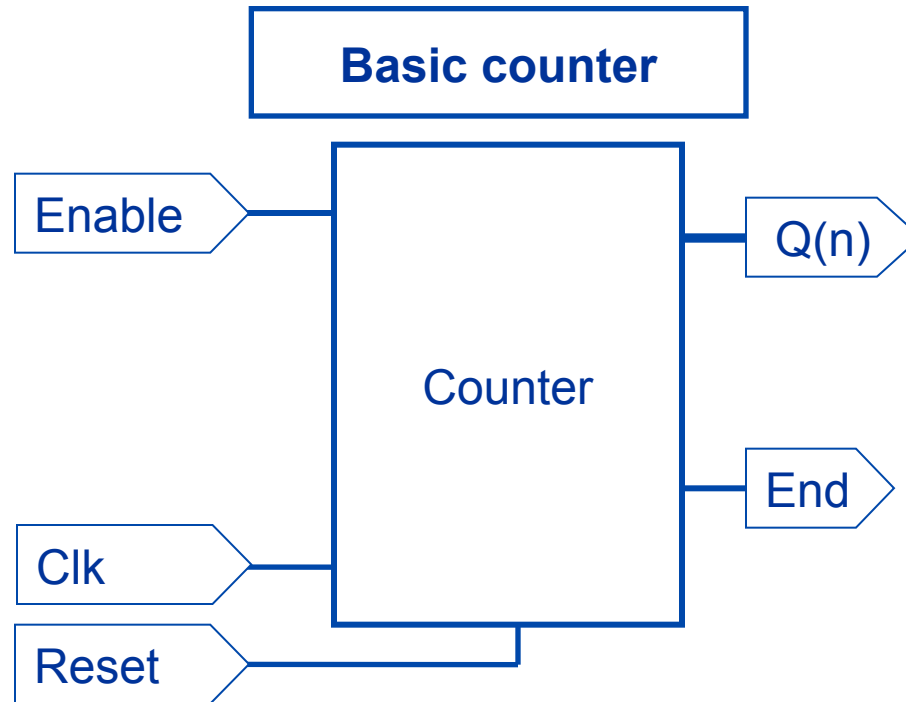
Universal Shift Register

Schematic



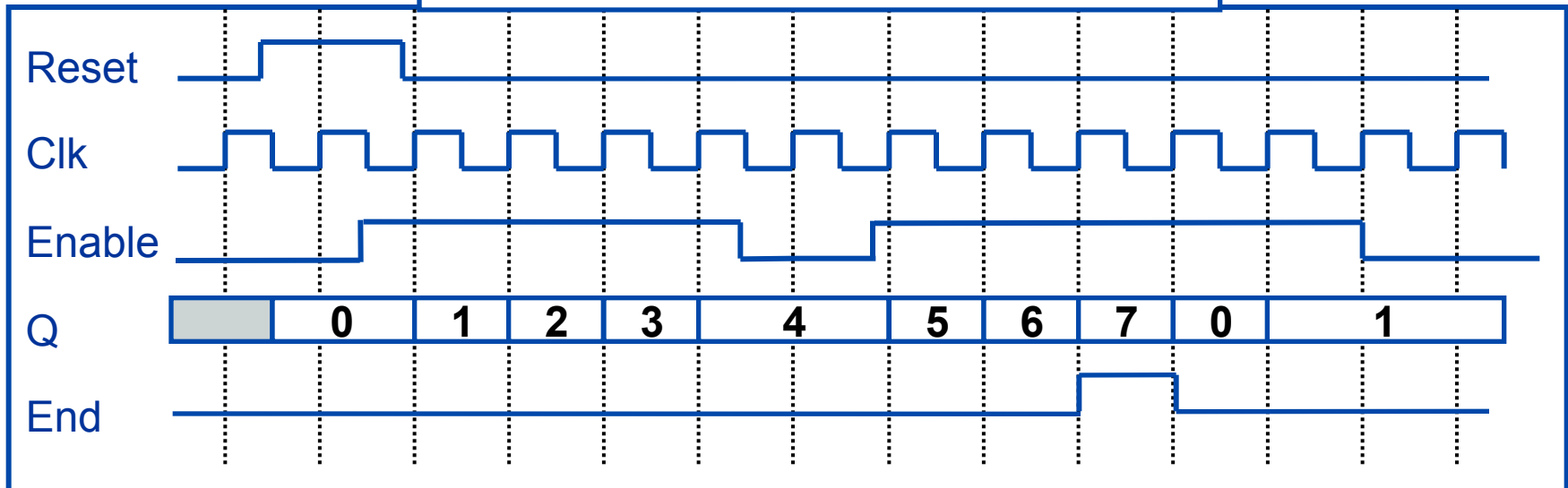
Synchronous Counters

Synchronous Counters



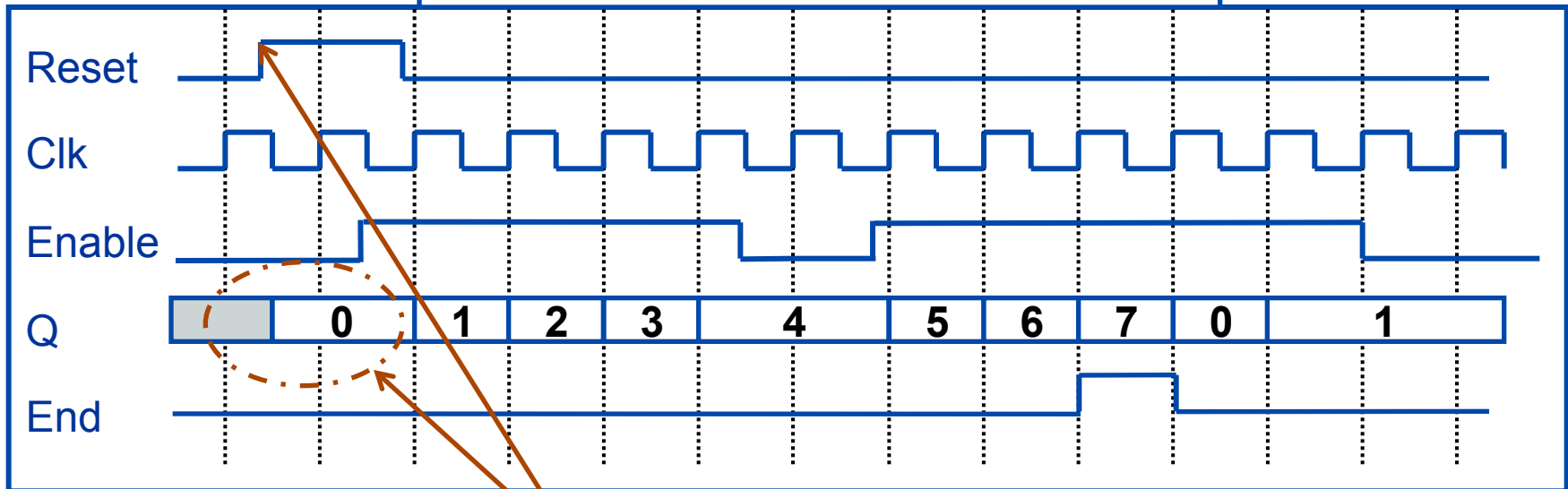
Synchronous Counters

Functional cronogram



Synchronous Counters

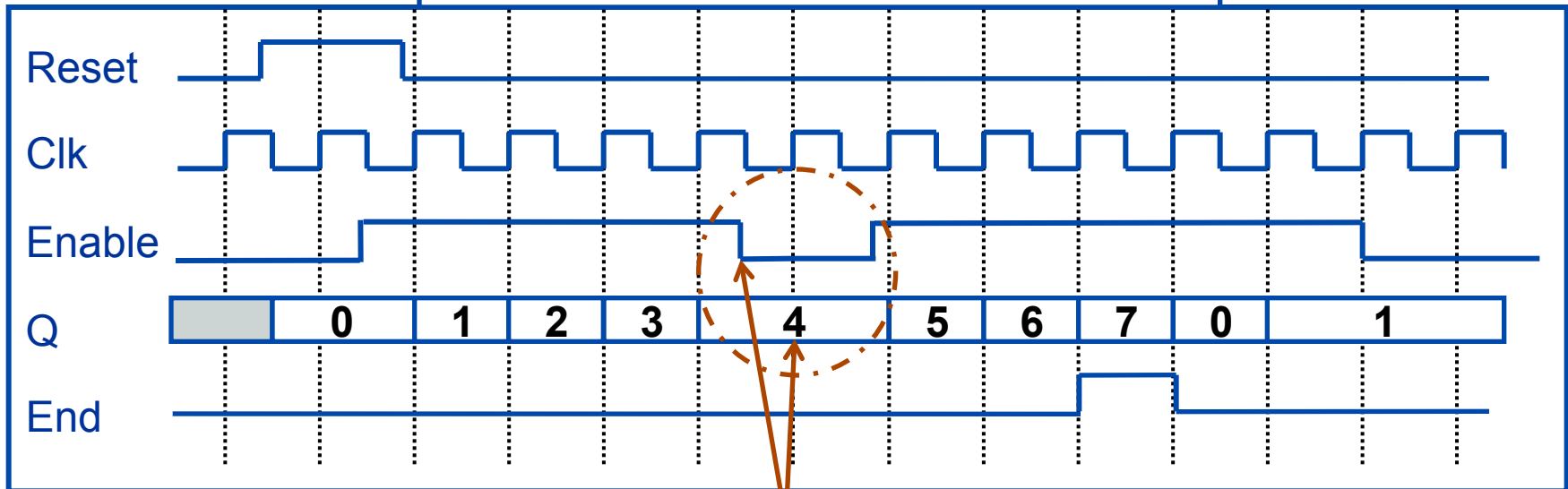
Functional cronogram



Asynchronous initialization

Synchronous Counters

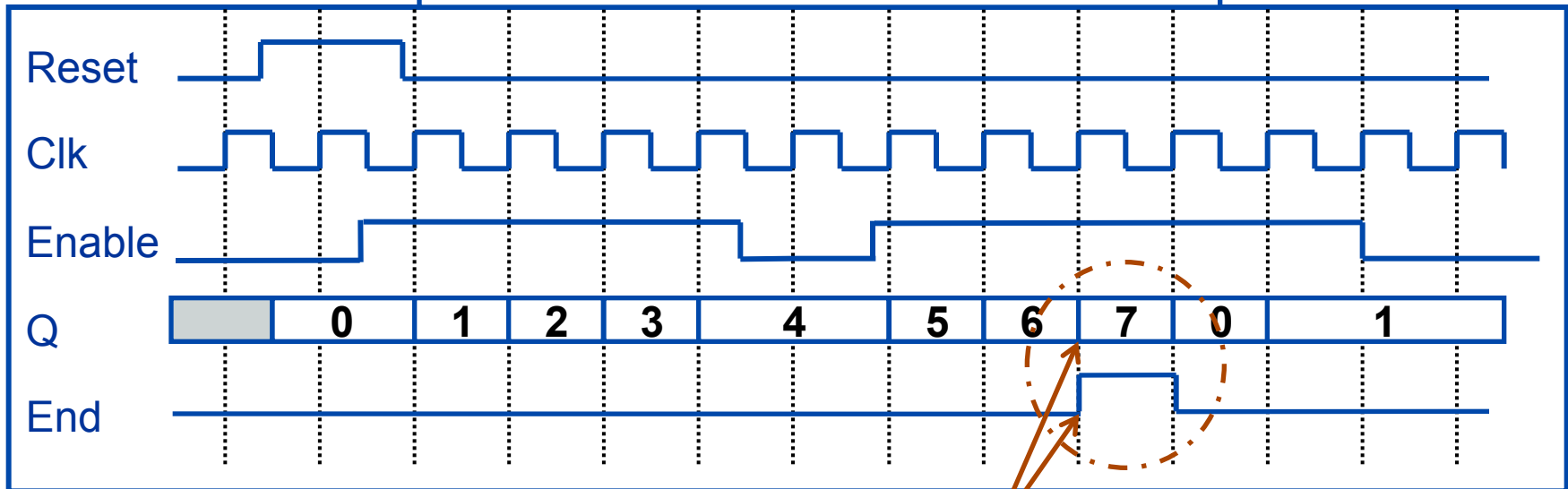
Functional cronogram



Stop of counting

Synchronous Counters

Functional cronogram



End of counting

Synchronous Counters as FSM

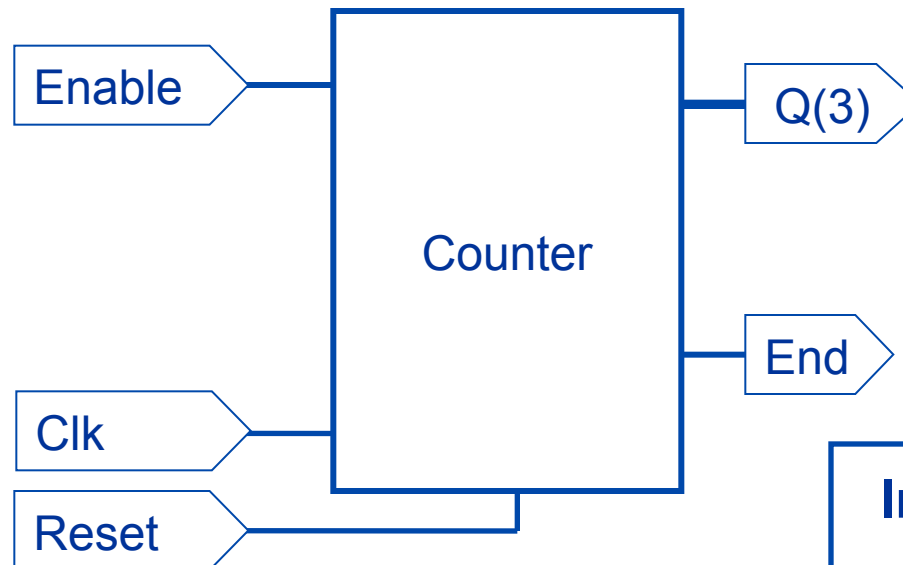
- 1. Inputs & Outputs*
- 2. State Diagram. State assignment. Flip-flops*
- 3. Transitions table*
- 4. Optimization*
- 5. Schematic*

Synchronous Counters as FSM

1. Input & Outputs

Counter
Natural Binary
0-7

Moore

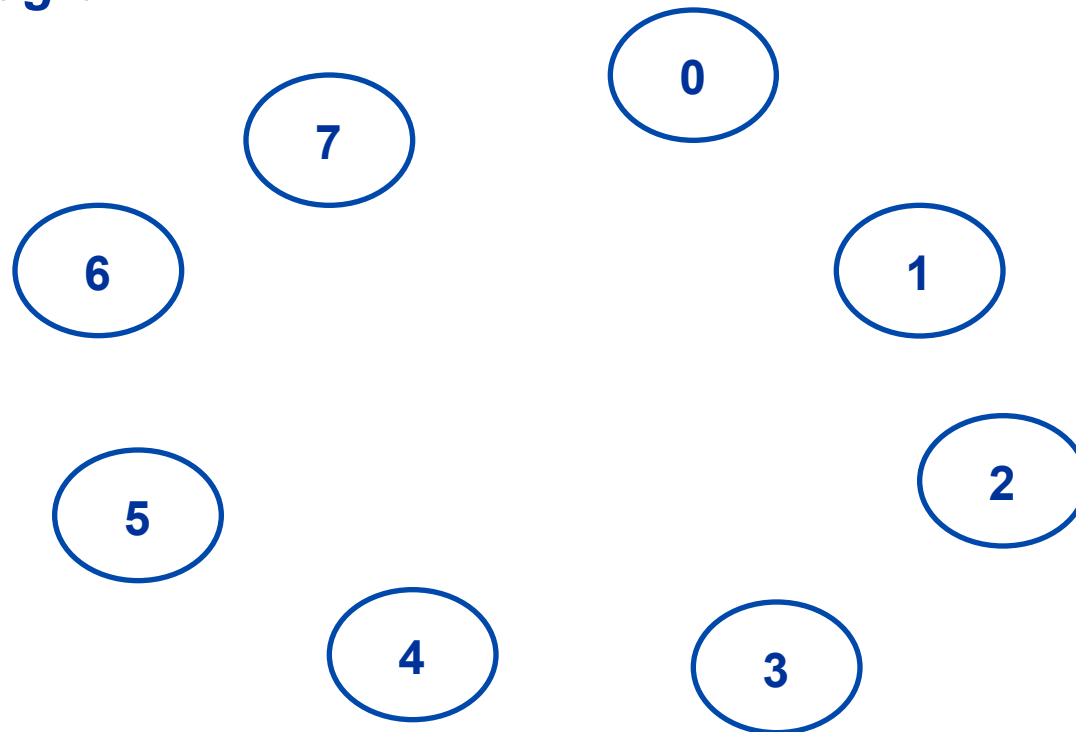


Inputs__Outputs

Ena__Q-E

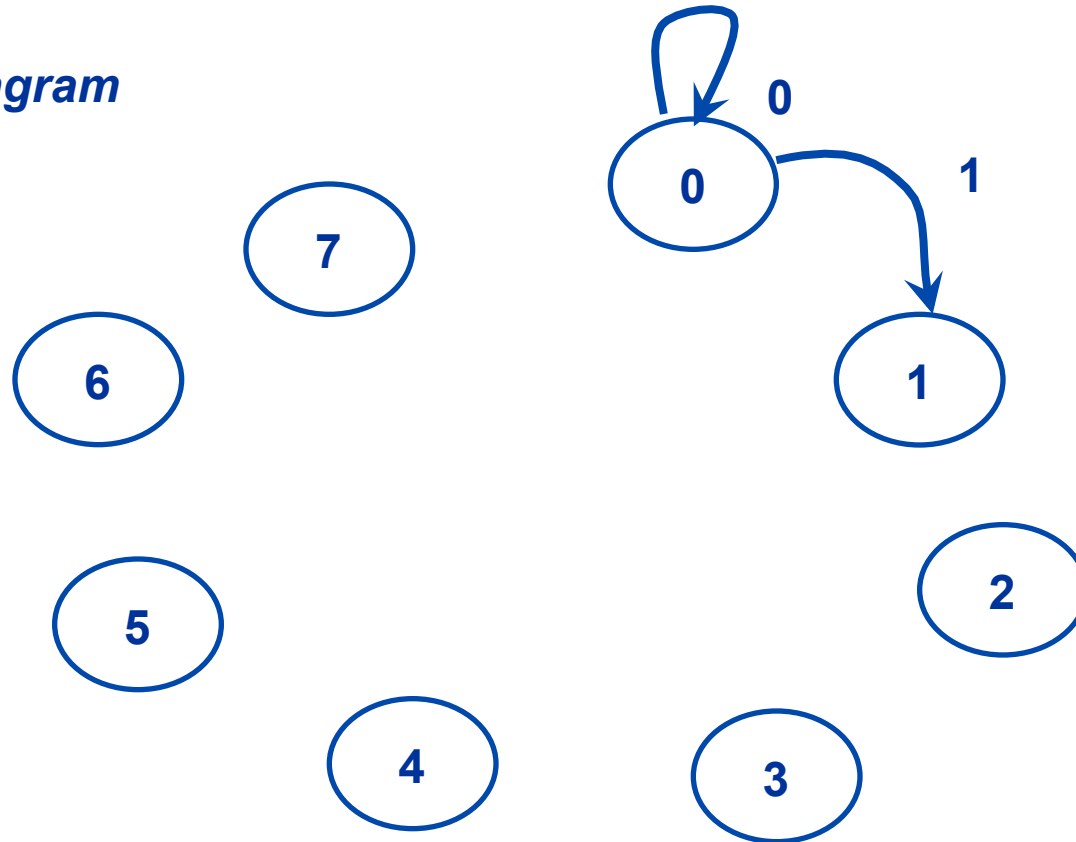
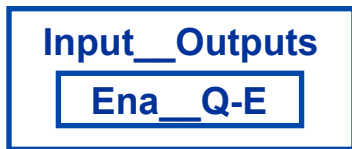
Synchronous Counters as FSM

2. State Diagram



Synchronous Counters as FSM

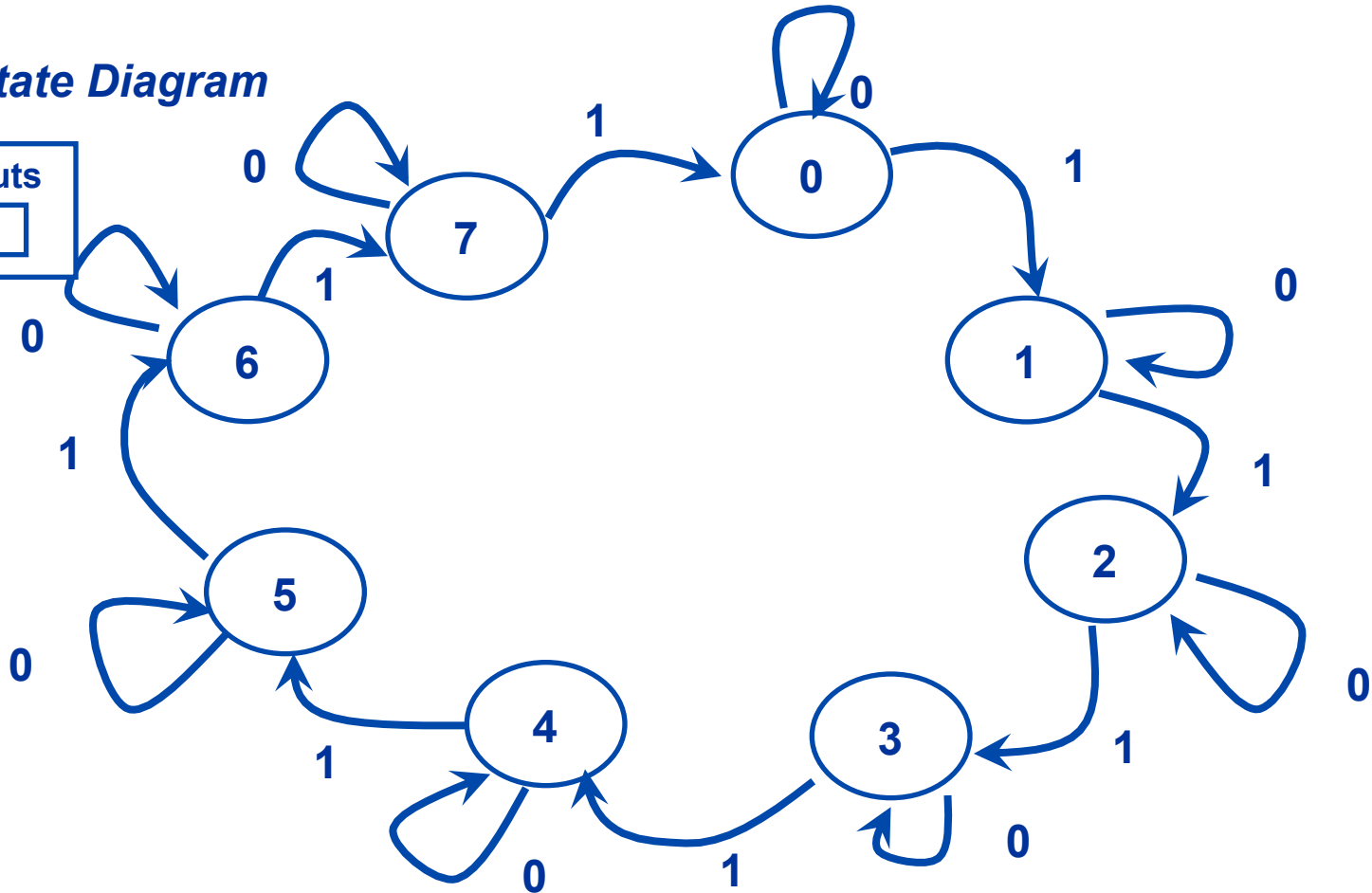
2. State Diagram



Synchronous Counters as FSM

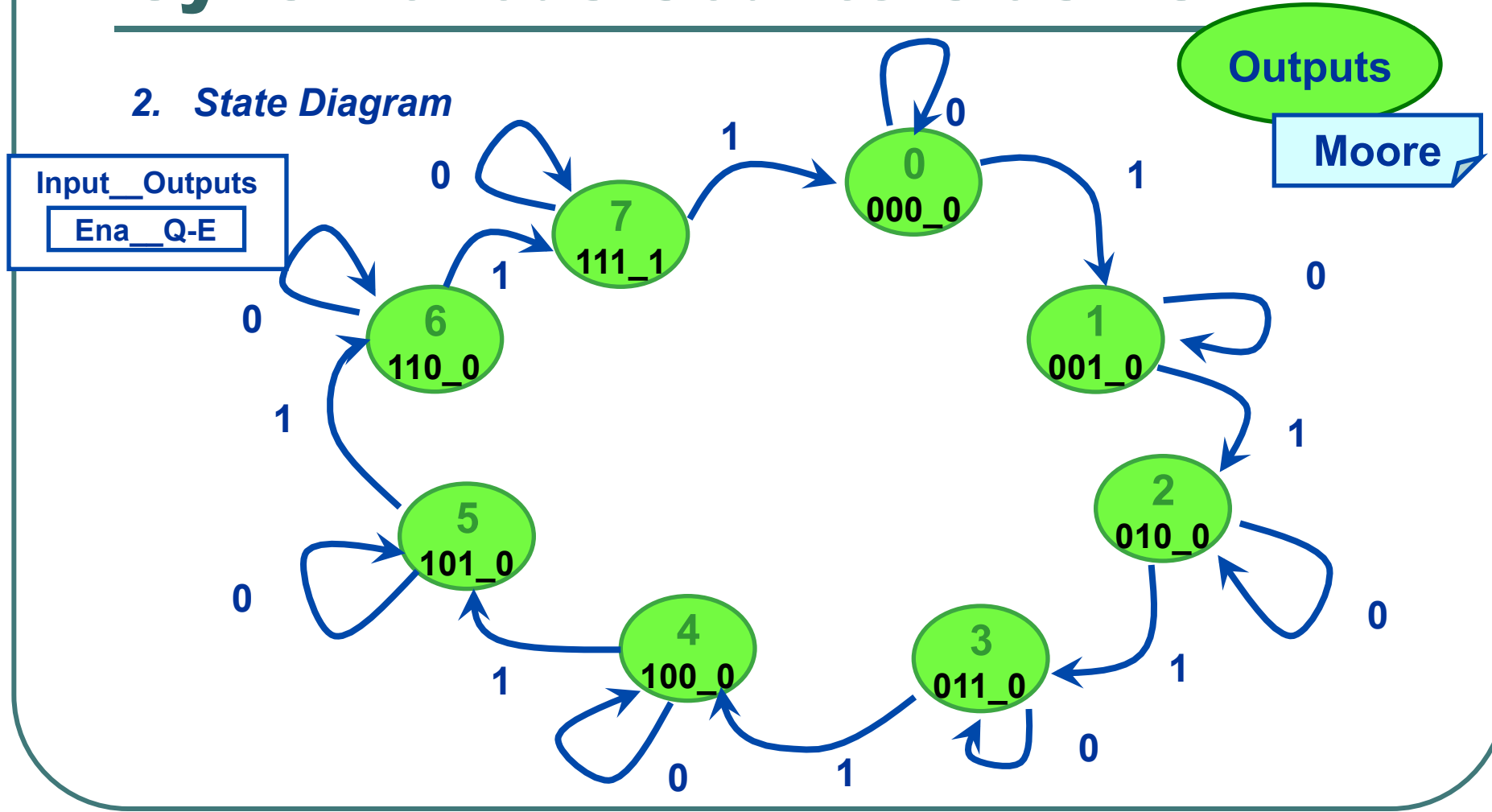
2. State Diagram

Input	Outputs
Ena	Q-E



Synchronous Counters as FSM

2. State Diagram



Synchronous Counters as FSM

2. State assignment

7 states → 3 flip-flops

State_Coding

0 →	000
1 →	001
2 →	010
3 →	011
4 →	100
5 →	101
6 →	110
7 →	111

Synchronous Counters as FSM

3. Transitions Table (I)

State	Input	State'	Input_T	Outputs
Q2Q1Q0	Enable		T2T1T0	Q - End
000	0	000	000	000-0
000	1	001	001	000-0
001	0	001	000	001-0
001	1	010	011	001-0
010	0	010	000	010-0
010	1	011	001	010-0
011	0	011	000	011-0
011	1	100	111	011-0
100	0	100	000	100-0
100	1	101	001	100-0
101	0	101	000	101-0
101	1	110	011	101-0

Synchronous Counters as FSM

3. Transitions Table(I)

State	Input	State'	Input_T	Outputs
Q2Q1Q0	Enable		T2T1T0	Q - End
110	0	110	000	110-0
110	1	111	001	110-0
111	0	111	000	111-1
111	1	000	111	111-1

Synchronous Counters as FSM

4. Optimization

T2

Q0 En Q2Q1	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	0	0	1	0
10	0	0	0	0

T1

Q0 En Q2Q1	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	0	0	1	0
10	0	0	1	0

Synchronous Counters as FSM

4. Optimization

T0

Q0 En / Q2Q1	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

End

Q0 En / Q2Q1	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	1	1
10	0	0	0	0

Synchronous Counters as FSM

4. Optimization

$$T2 = Q2 \cdot Q1 \cdot \text{Enable}$$

$$T1 = Q0 \cdot \text{Enable}$$

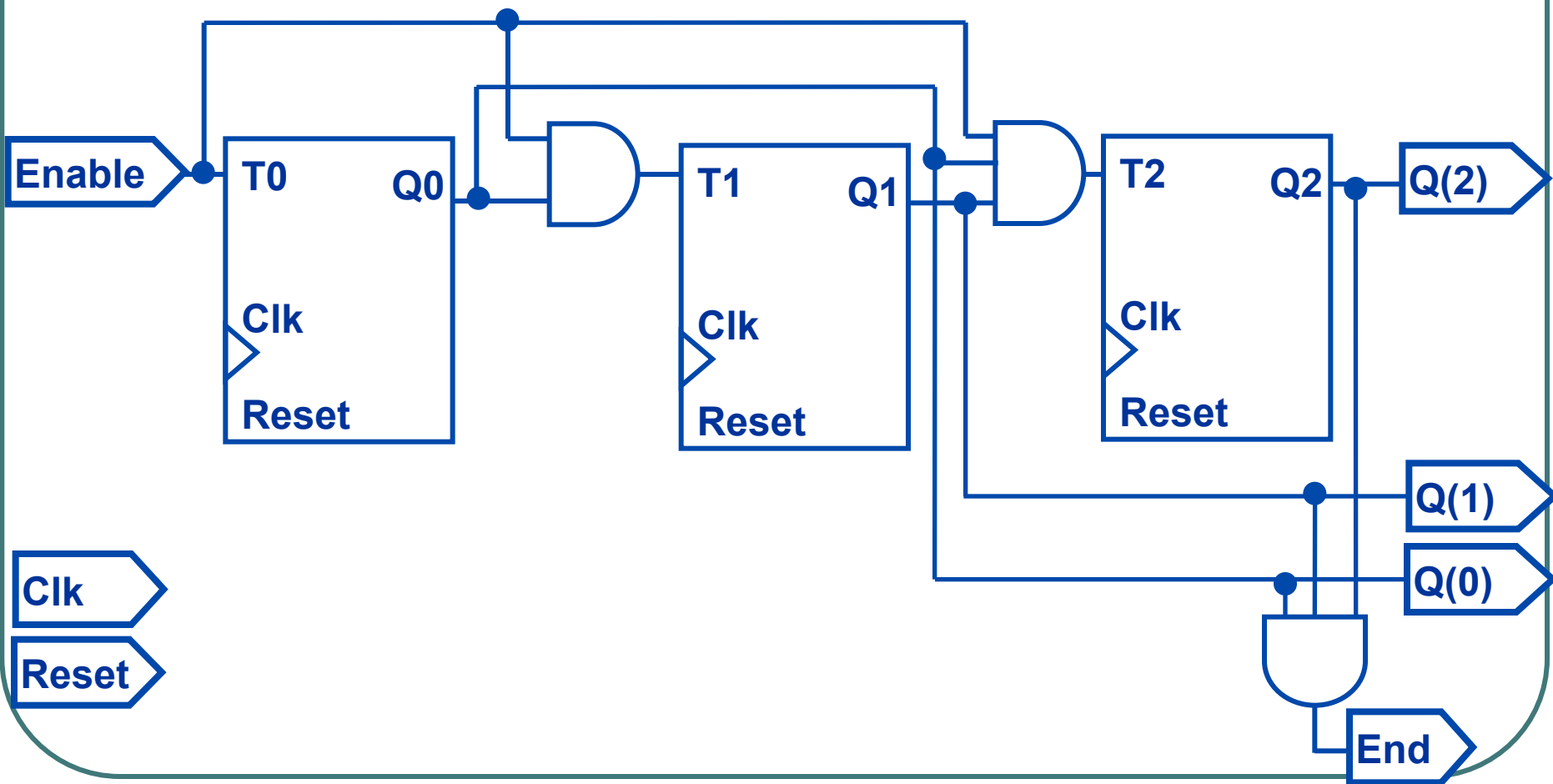
$$T0 = \text{Enable}$$

$$\text{End} = Q2 \cdot Q1 \cdot Q0$$

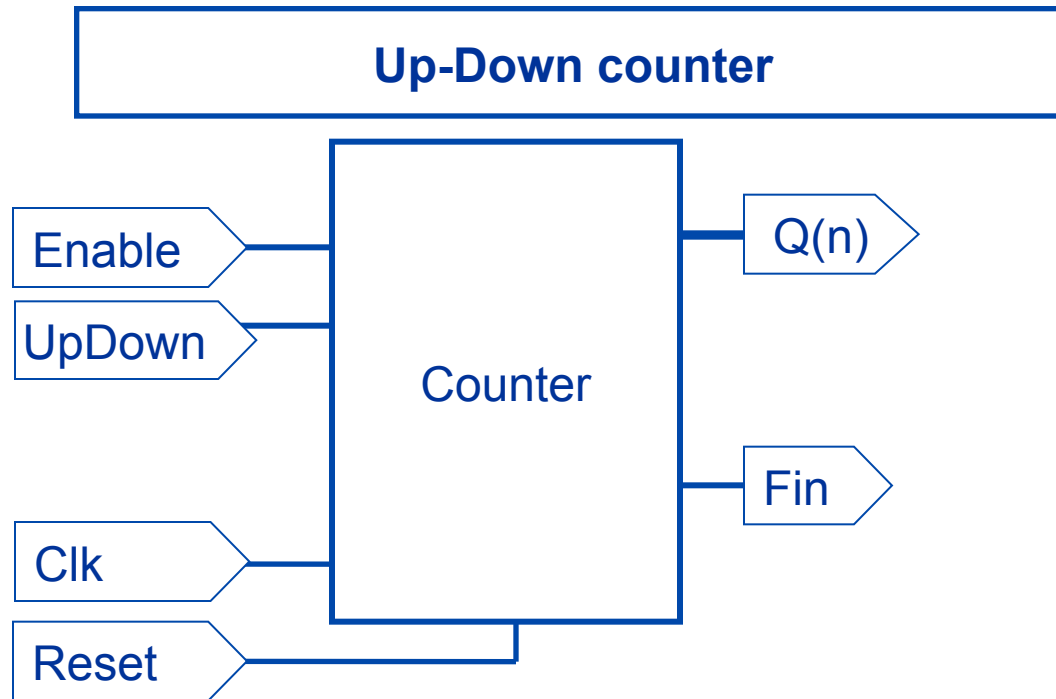
Synchronous Counters as FSM

Schematic

5. Schematic

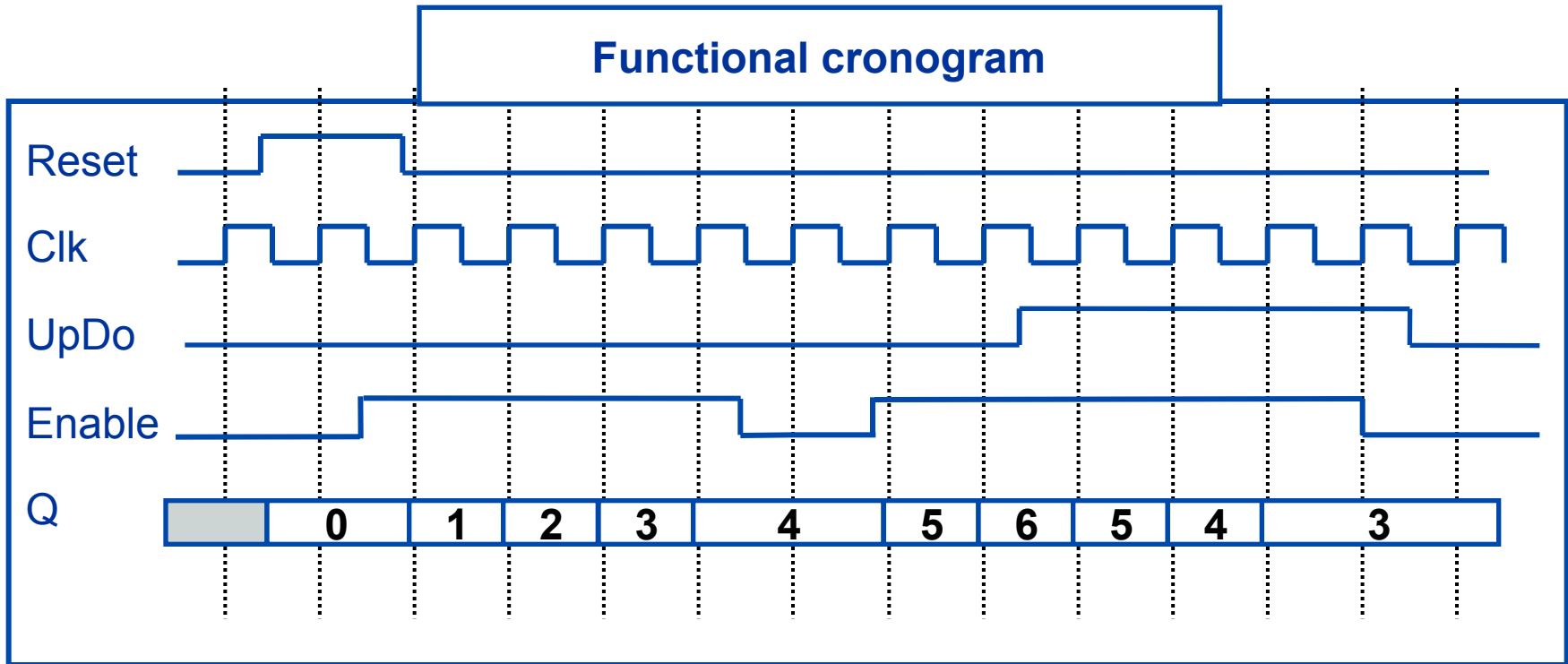


Synchronous Counters



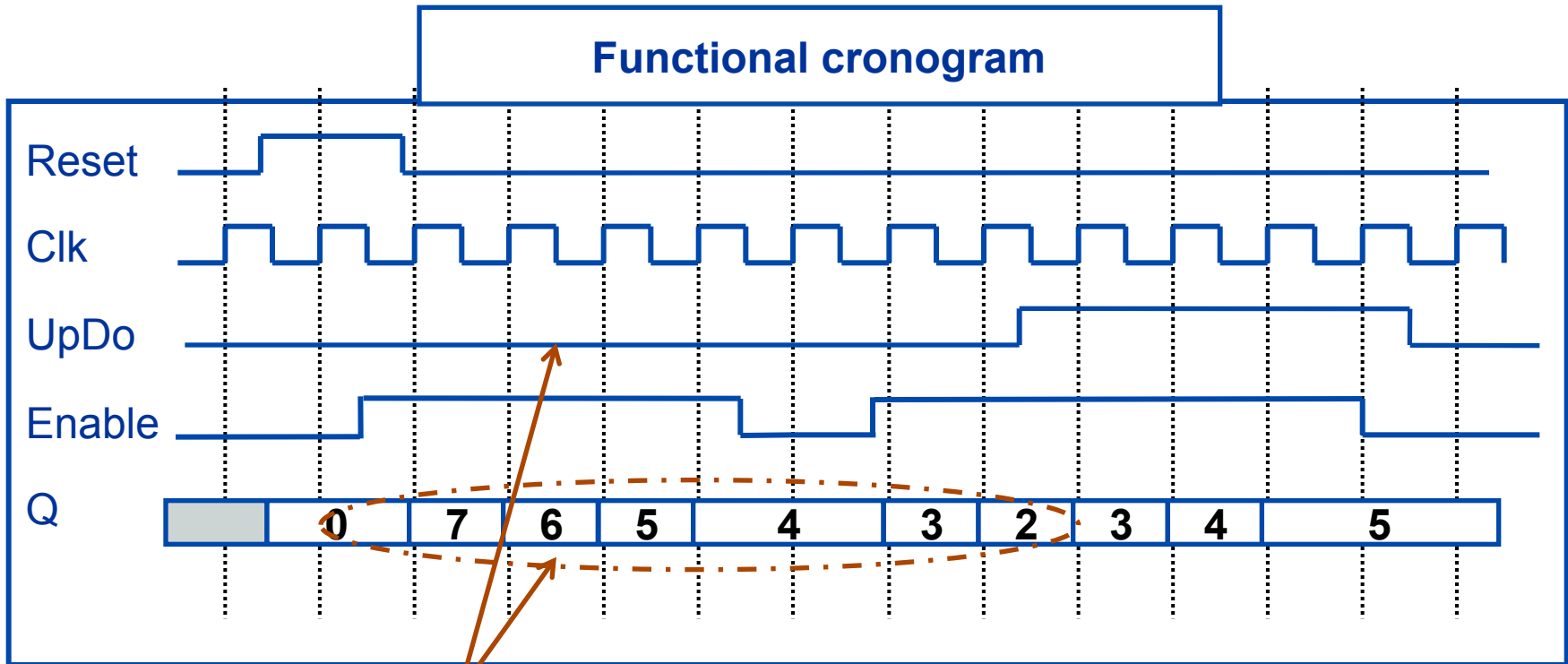
Synchronous Counters(Asc/Desc)

Functional cronogram



Synchronous Counters(Asc/Desc)

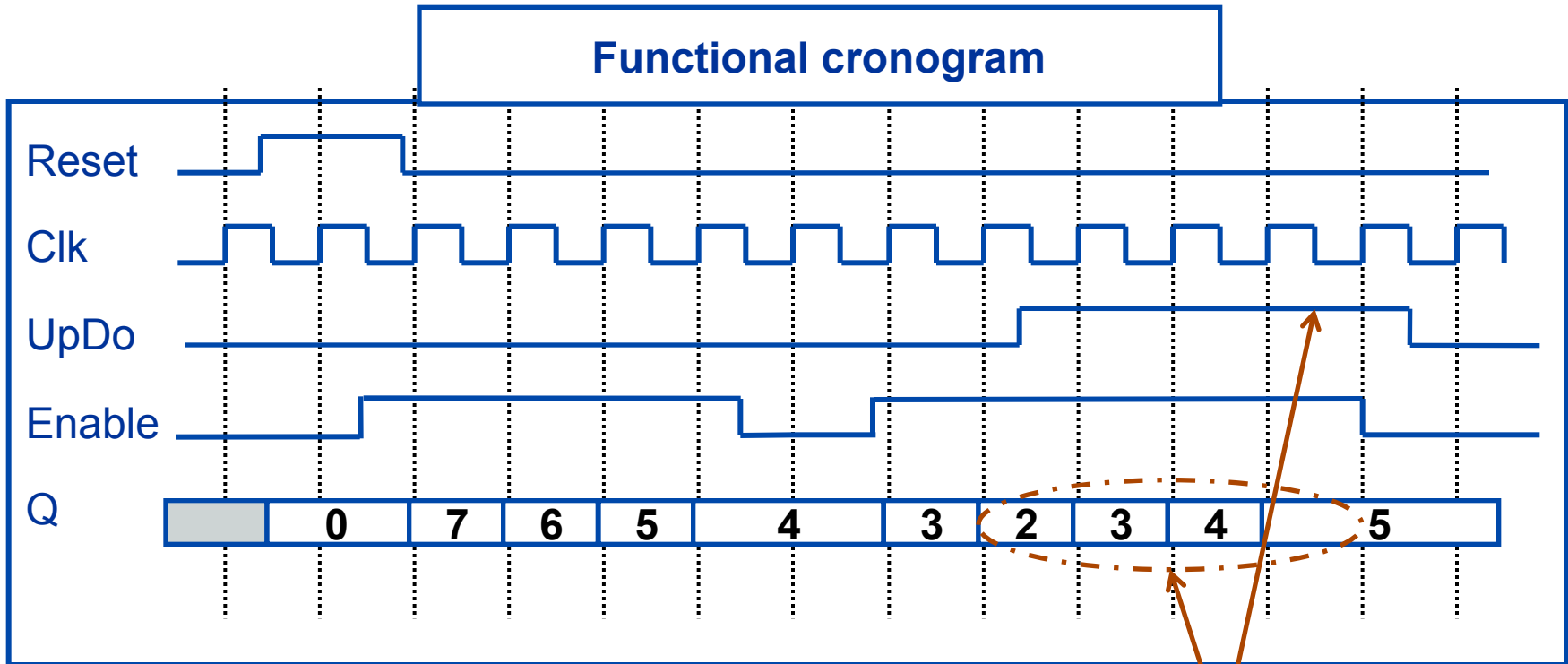
Functional cronogram



Descending count

Synchronous Counters(Asc/Desc)

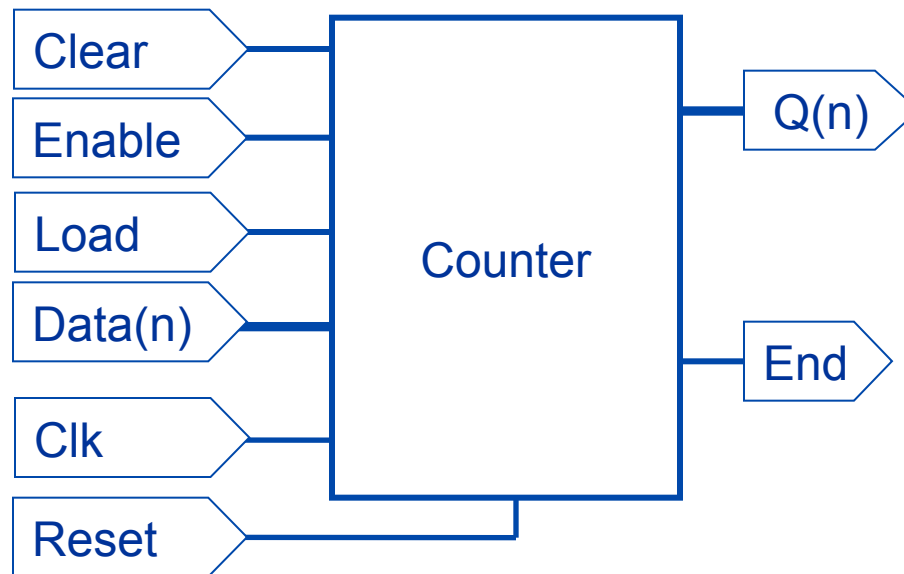
Functional cronogram



Ascending count

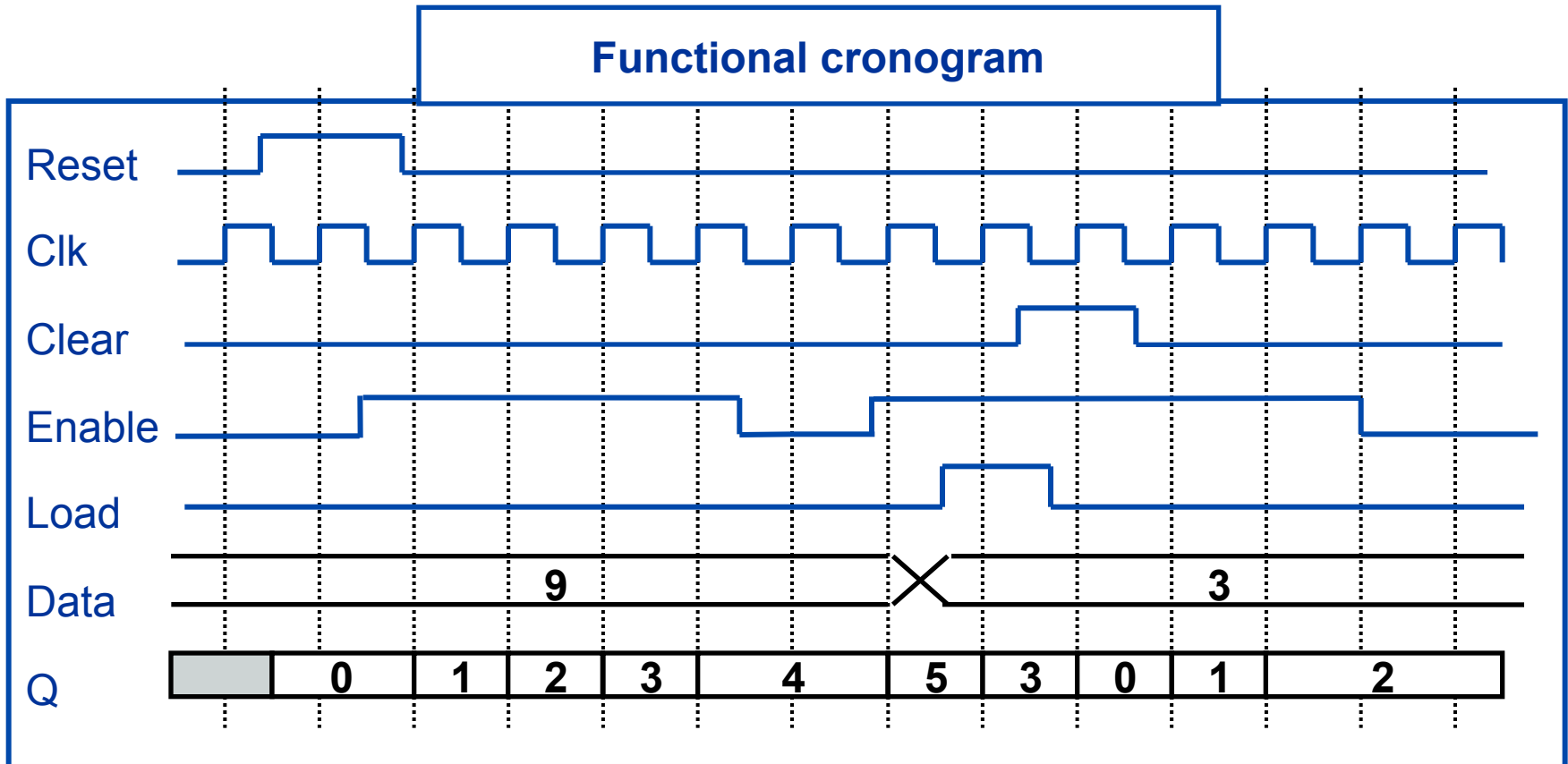
Synchronous Counters

Counter with Pre-Load and Synchronous clear

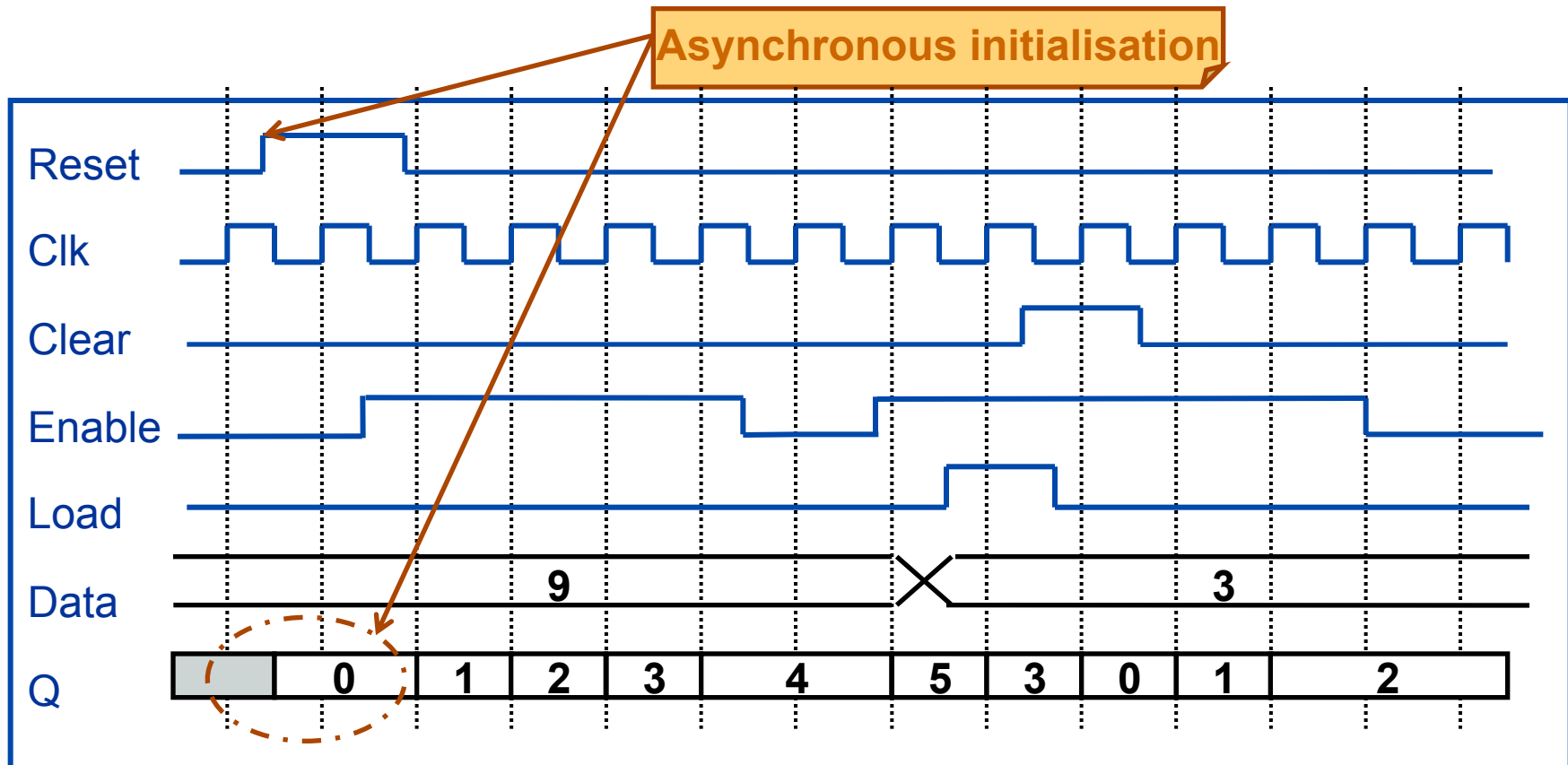


Synchronous Counters

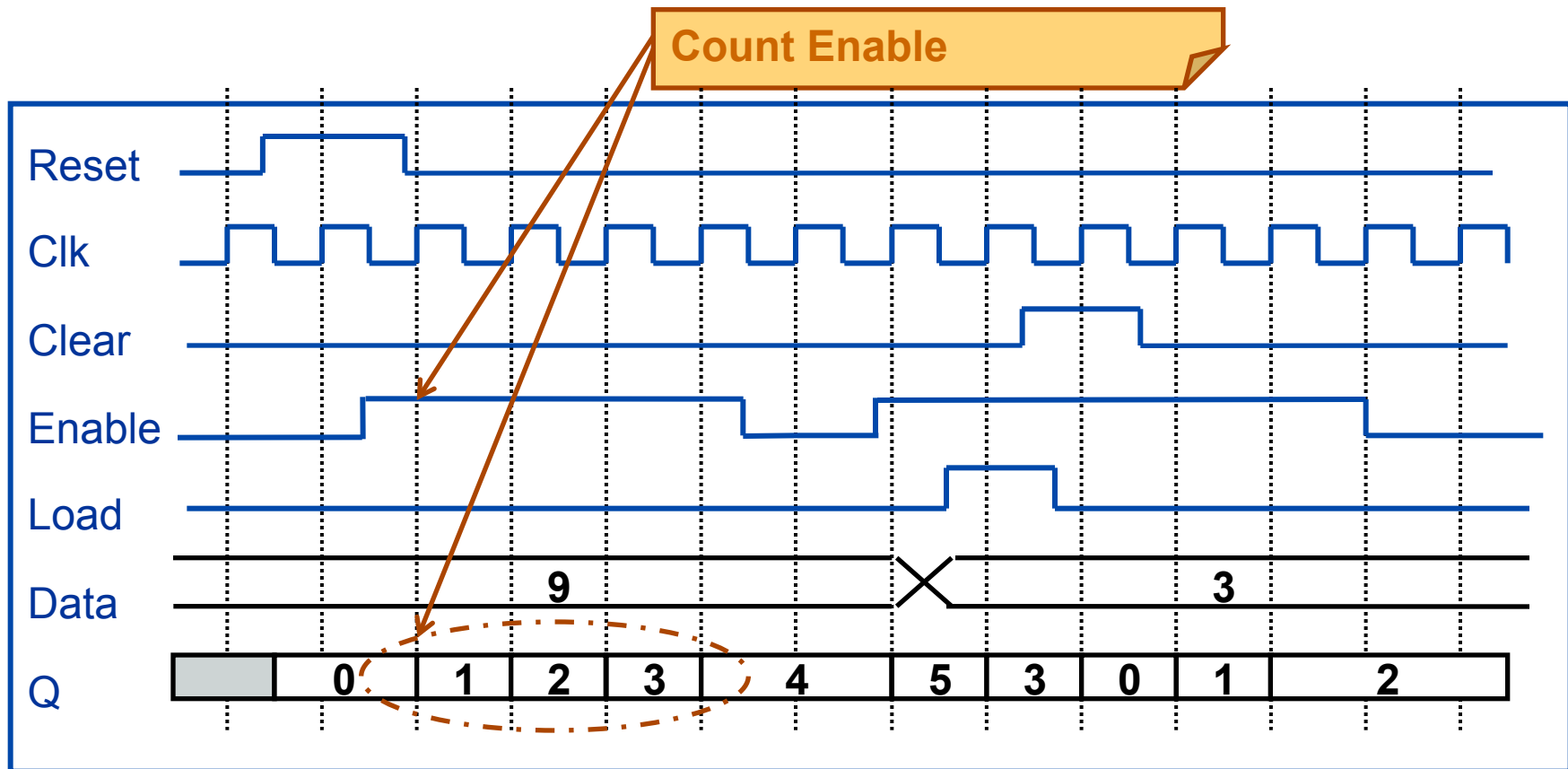
Functional cronogram



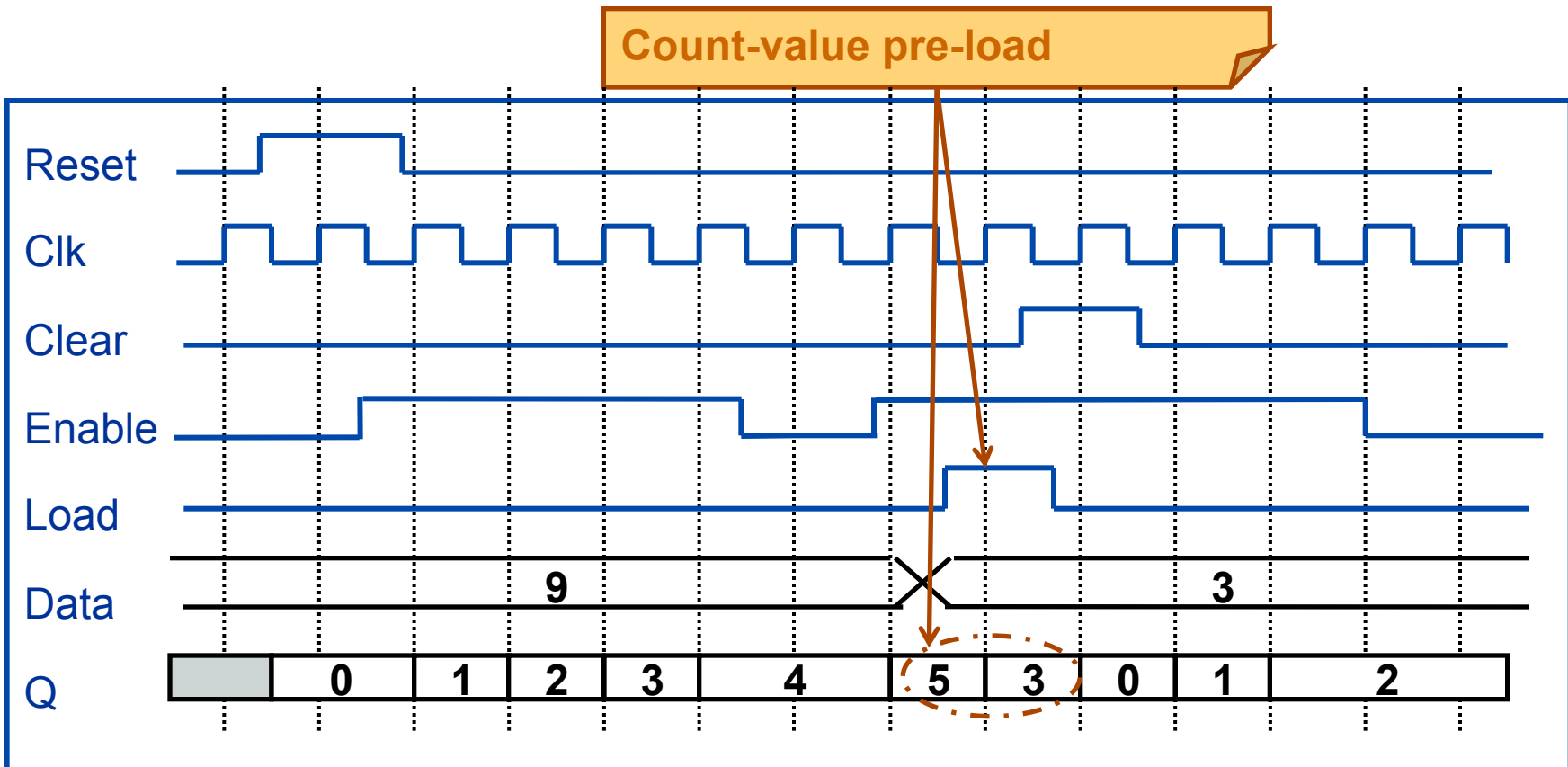
Synchronous Counters



Synchronous Counters

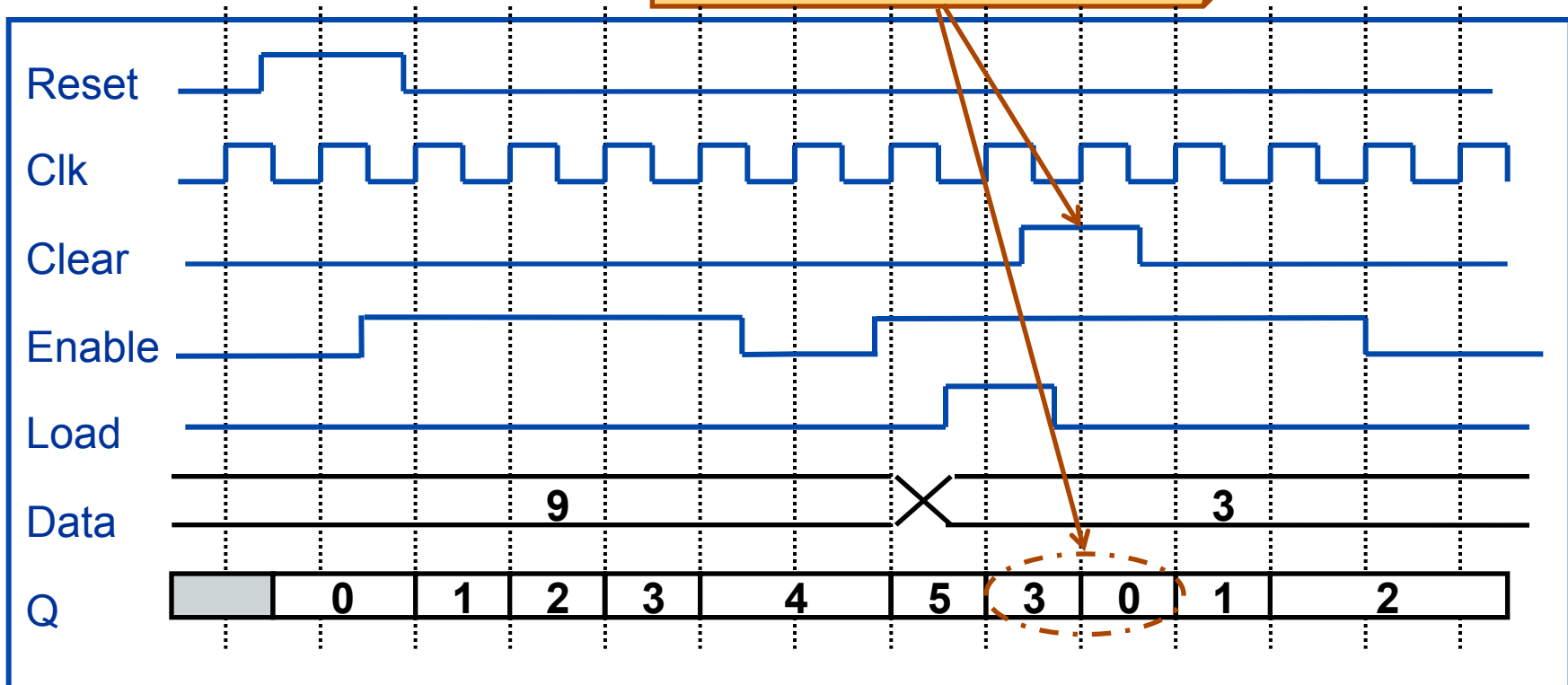


Synchronous Counters



Synchronous Counters

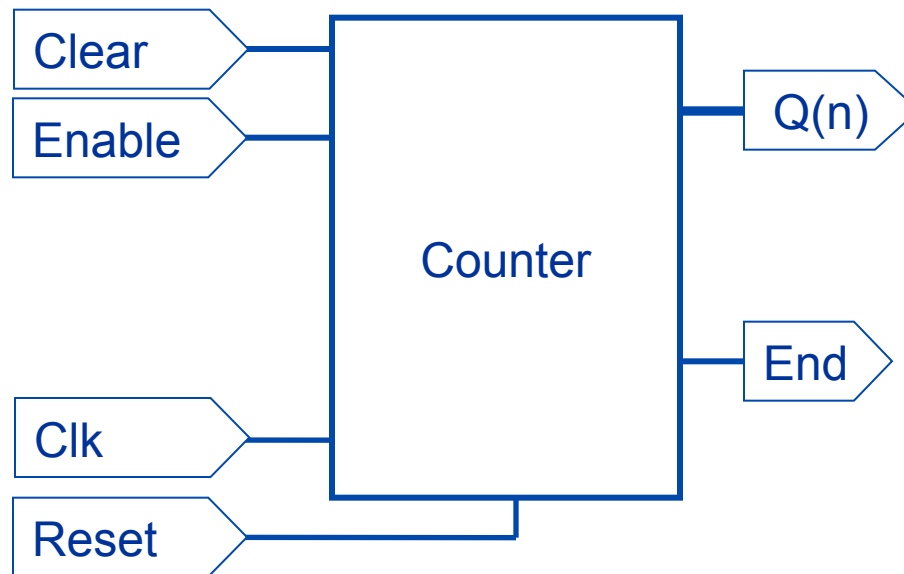
Synchronous initialisation



Synchronous Counters(FSM)

Counter with input Clear

Exercise



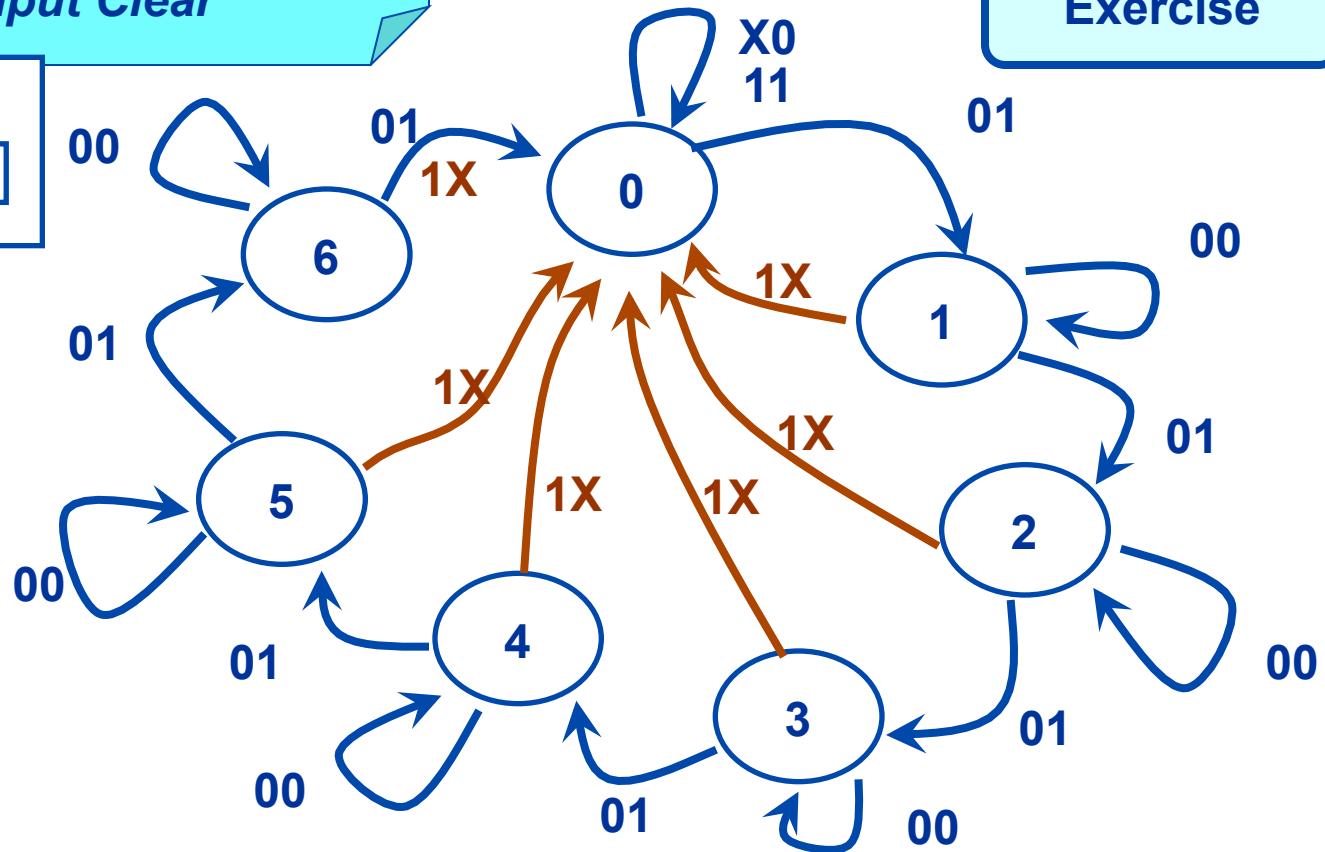
Synchronous Counters(FSM)

Counter with input Clear

Exercise

Inputs__Outputs

Clear-Ena__Q-End



Synchronous Counters(FSM)

Counter with input Clear

Exercise

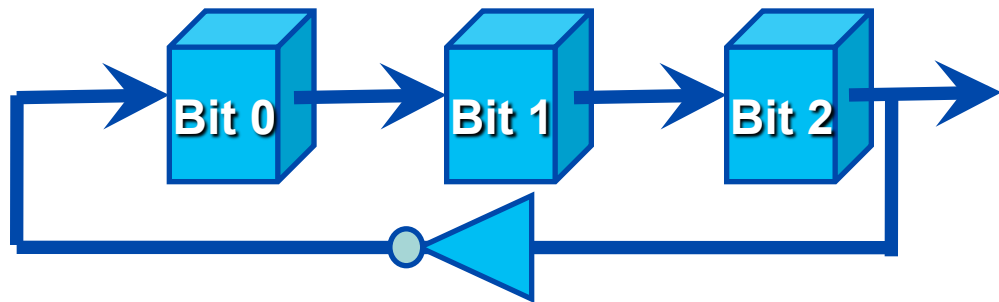


Applications with synchronous counters: sequencers

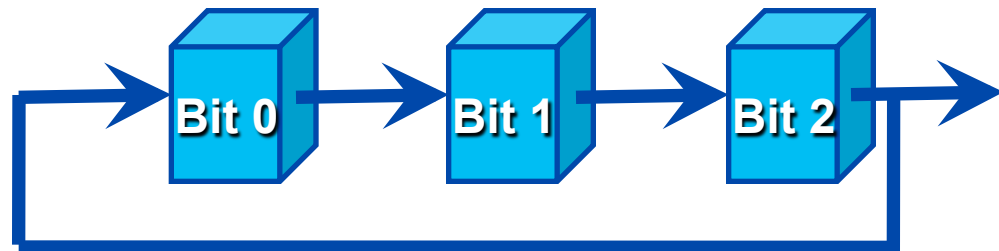
Counters based on Shift-Registers

“A counter based on a shift register is basically a shift register with a feedback from serial output to serial input, in such a way that special sequences are generated” (Floyd)

Johnson Counter



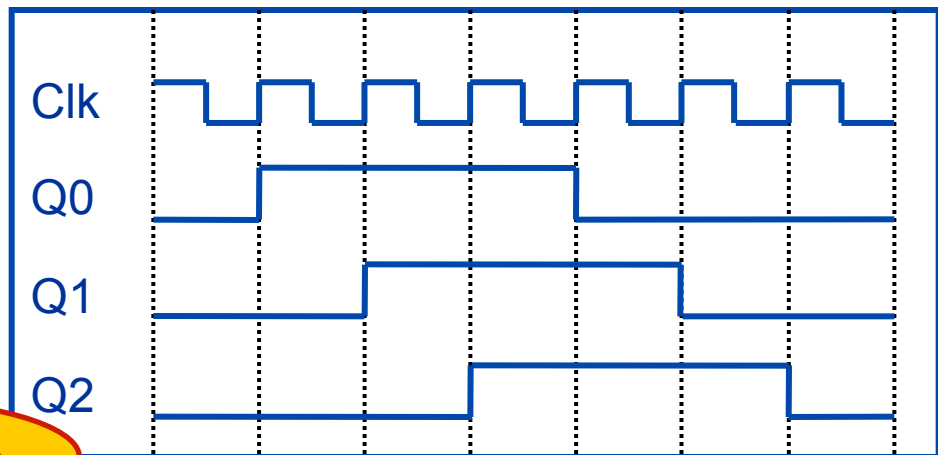
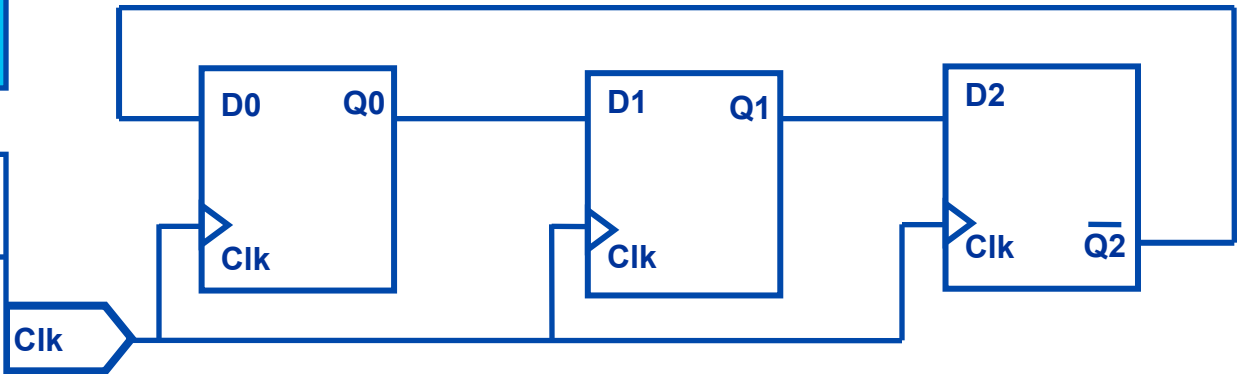
Ring Counter



Counters based on Shift-Registers

Johnson Counter

Cycle	Q2	Q1	Q0
0	0	0	0
1	0	0	1
2	0	1	1
3	1	1	1
4	1	1	0
5	1	0	0
6	0	0	0
7	0	0	1

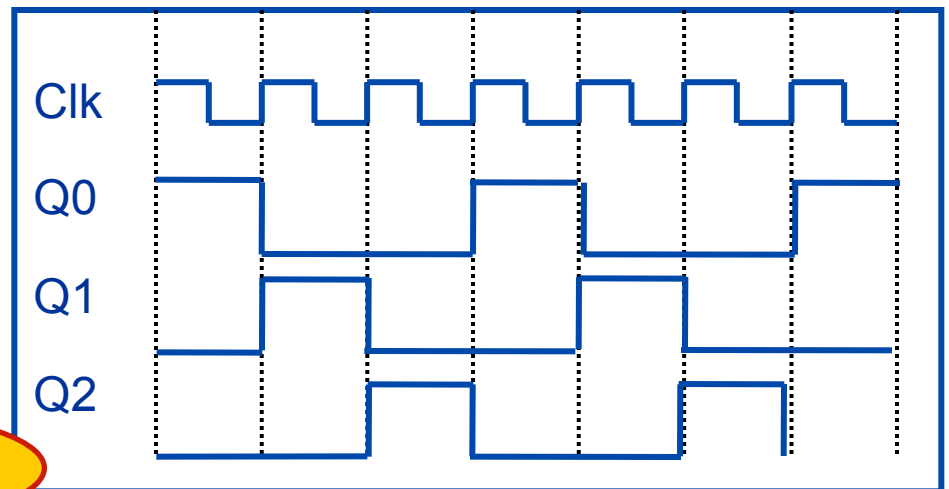
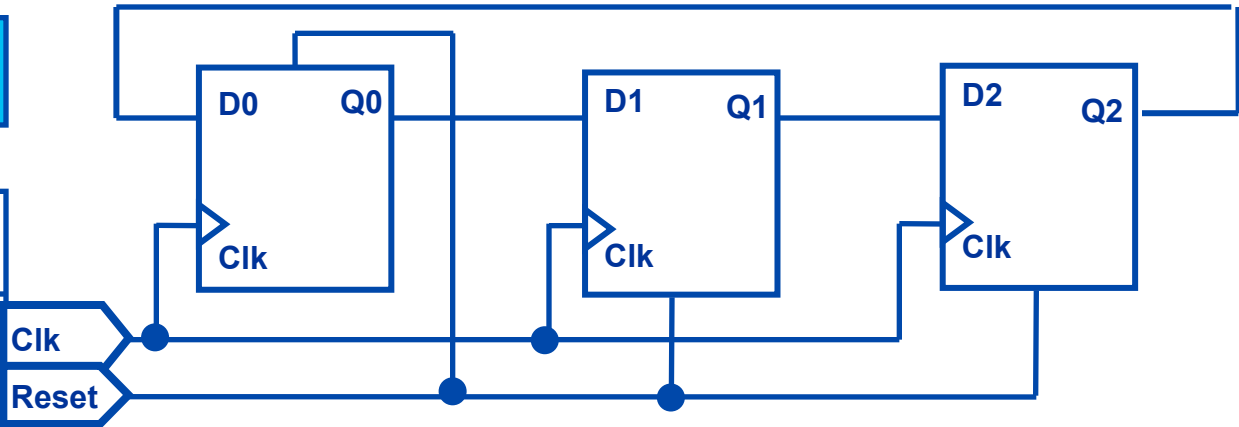


2n patterns

Counters based on Shift-Registers

Ring Counter

Cycle	Q2	Q1	Q0
0	0	0	1
1	0	1	0
2	1	0	0
3	0	0	1
4	0	1	0
5	1	0	0
6	0	0	1
7	0	1	0



n patterns