Victoria University of
Bangladesh

CSE-223 - Fall Exam - 2024

Digital Electronics & Pulse Technique

SM Mithon Hossen
212121003I

Q-01

Ans to the Qs No-1 (a)

A master-salve Flip-Flop is a type of sequential circuit that consists of two flip-flops connected in series- one acting as the Master and the other as the Slave. The master is triggered by the clock signal directly, while the save si is triggered by inverted clock signal. This setup helps in eliminating race conditions and ensure stable data transfer.

Working Principle

When the clock is HIGH, the master flip-flop is active and captures the input. When the clock gose LOW, the slave flip-flop the stored value from the master. This ensures that the output chanys only during the falling edge of the reducing the of glitches.

Example: Master-Slave Flip-Flop

A flip-flop can be implemented as a Master-Save Flip-Flop using two flip-flops.

p-02

The Master Flip-Flop takes inputs J and K
The Slave Flip-Flop follows the master but update on the falling edge of clock

Truth Table for JK Master-Slave Flip-Flop

J	K	Output (Q)
0	0	No Change
0	1	Reset (0)
1	0	Set (1)
1	1	Toggle

~~Em~~ Eliminates timing issues by using two clocked flip-flops.

~~Red~~ Reduces race conditions in circuits.

Ans to the Qs No 1 (b)

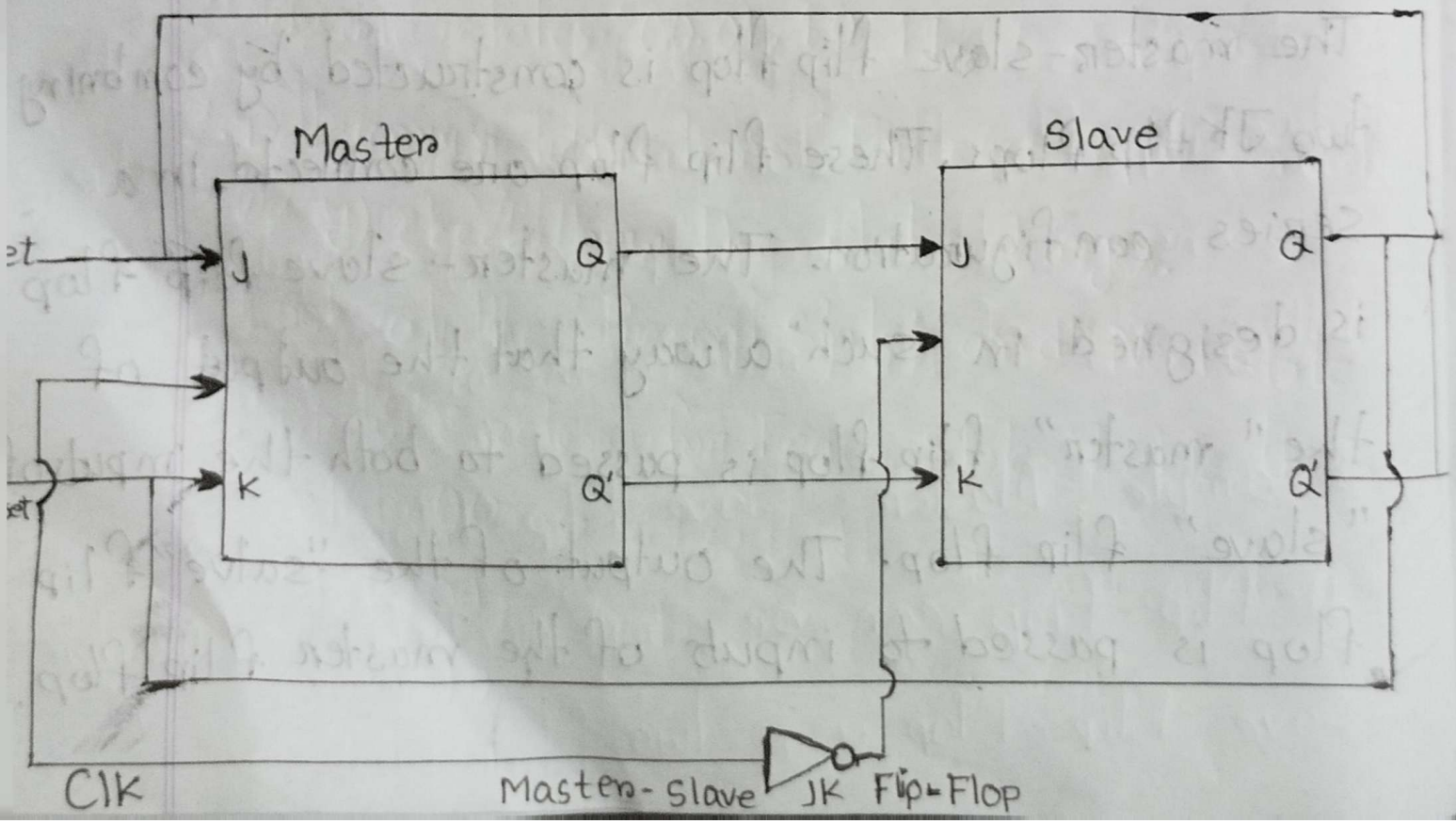
Master - Slave Jk Flip Flop

In "JK Flip Flop", when both the inputs and CLK set to 1 for a long time, then Q output toggle until the CLK is 1. Thus, the uncertain or unreliable output produces. This problem is referred to as a race-round conidition in Jk fip-flop and avoided ensuring that that CLK set to 1 only for very short time.

Explanation

The master-slave flip flop is constructed by combning two Jk flip flops. These flip flop are connecld, in a series configuration. The master-slave flip flop is designed in such a way that the output of the "master" flip flop is passed to both the inputs of "slave" flip flop. The output of the "salve" flip flop is passed to inputs of the master flip flop.

In "master-salve flip flop", apart from these two flip flops, an inverter or NOT gate is also used. For passing the inverted clock pulse to the "slave" flip flop, the ~~int~~ inverter is connected to the clocks pluse. In simple words, when CP set to false for "master", then CP is set to true for "slave", and when CP set to true for "master", then CP is set to false to "salve".



Working:

- When the clock pulse is true, the slave flip flop will be in the isolated state, and the system's state may be affected by the J and K inputs. The "slave" remains isolated until the CP is 1. When the CP set to 0, the master flip-flop pases the information to the slave flip-flop to obtain the output.

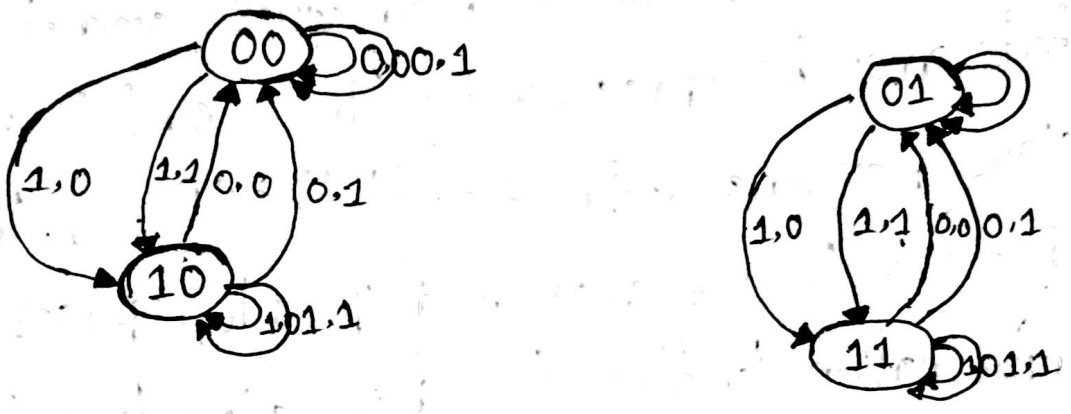
- The master flip flop responds first from the slave because the master flip flop is the positive level trigger, and the slave flip flop is the negative level trigger.

- The output Q' = 1 of the master flip flop is passed to the slave flip flop as an input J set to 0 and K set to 1. The clock forces the slave flip flop to work as reset and then the slave copies the master flip flop.

p-06

QS NO 2-A) draw the state diagram for the given table??

Ans to the Qs. No - 2 (a)



Steps to Draw the state Diagram:

1. Identify the states: The states are represented by (A, B) pairs.

   Possible states: 00, 01, 10, 11.

2. Determain Transitions: For each input condition (x, y) check how the state changes.

   Look at the "Present State" and "Next State" columns

3. Draw Circles for Feach State: Lable them as:

   State 00

   state 01

   state 10

   state 11

4. Draw Directed Arrows for Transitions:

If the present state (A, B) moves to another state (A, B) based on (x, y) draw an arrow.

Lable the arrow with the input condition that causes the ~~to~~ transition.

5. Indude Output Values!

Each transition can be labeled with "x, y/z" where is the output.

P-08

Ans to the Qs No 3 (a)

8-bit storage latch which feeds parallel data to an 8-bit shift register. Data can also be loaded serially. The shift register output, Q_H, is a 3-state output, allowing this device to be used in bus-oriented systems.

The HC589A directly interfaces with the SPI serial data port on CMOS MPUs and MCUs.

Features:

- Output Drive Capability: 15 LSTTL Loads.
- Outputs Directly Interface to CMOS, NMOS, and TTL.
- Operating Voltage Rang: 2.0 to 6.0 V
- Low Input Current: 1 μA
- High Noise Immunity Characteristic of CMOS Devices
- In Compliance with the Requirements Defined by JEDEC Standard No. 7A.

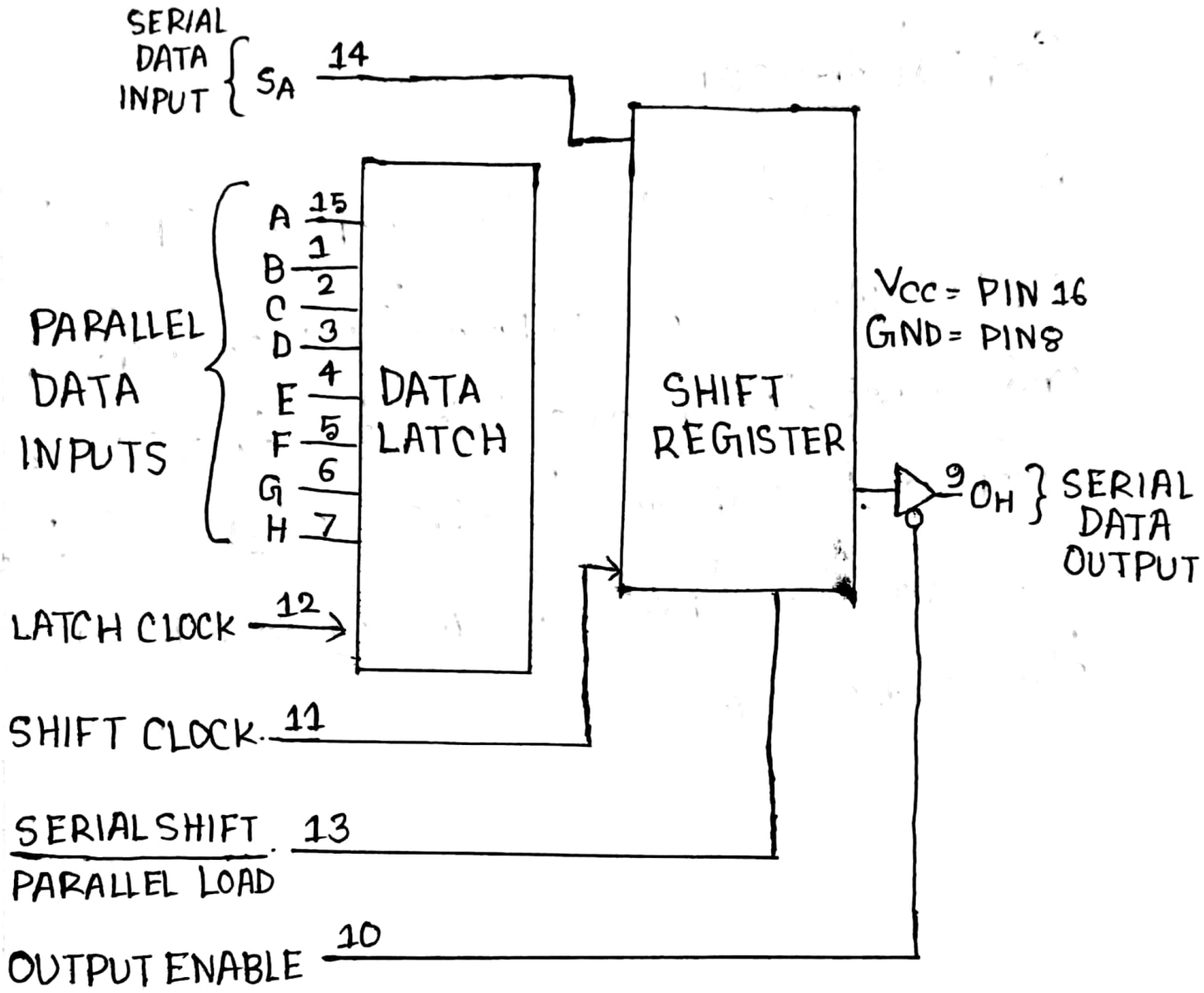


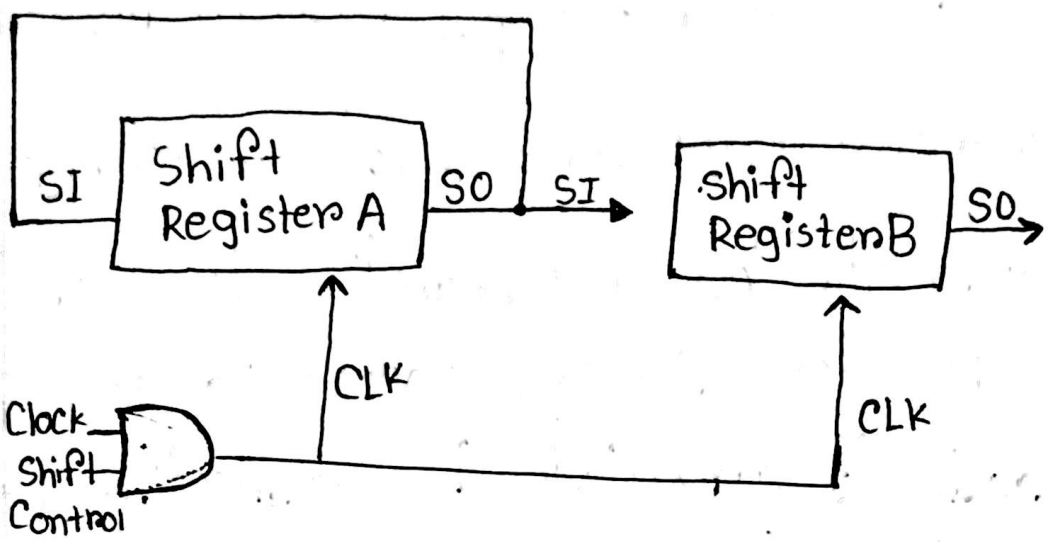
Figure 1. Logic Diagram

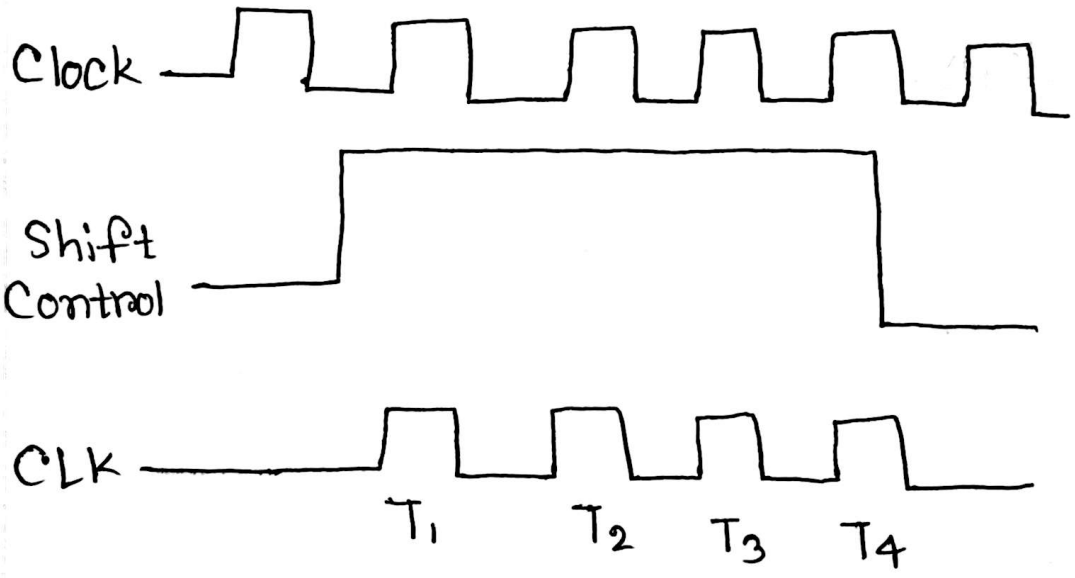
Ans to the Qs No 3 (b)

Serial Transfer:

A digital system is said to ~~people~~ operate in a serial Mode when information is transfred and multipulated one bit at a time. This is contrast to parallel transfer where all the bits of the register are transferred at the same time.

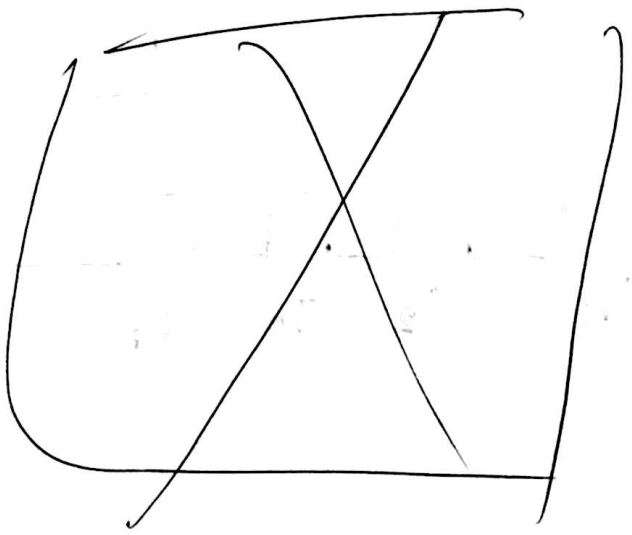
P-11


SI
Shift Register A
SO
SI
Shift Register B
SO
CLK
CLK
Clock
Shift Control

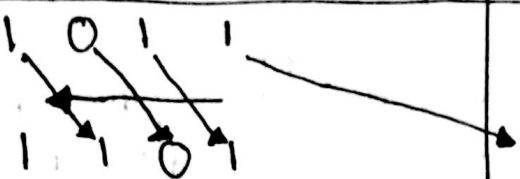

Clock
Shift Control
CLK
T1
T2
T3
T4

p-12

Soln: The serial out put (SO) of register A is connected to the serial input (SI) of register B. To prevent the loss of information stored in the source register, the information in register A is made to circulate by connecting the serial output to its serial input. The serial transfer ~~to~~ from A to B occurs in four steps.



Timing Pulse	Shift Register A	Shift Register B
Initial value	1 0 1 1	0 0 1 0
After T_1	1 1 0 1	1 0 0 1
After T_2	1 1 1 0	1 1 0 0
After T_3	0 1 1 1	0 1 1 0
After T4	1 0 1 1	1 0 1 1



Ans to the Qs No : 4 (a)

To draw the state diagram for the given state table, follow these steps:

1. ~~IDen~~ Identify States.

Each state is represented by the values of (A, B), where:

00 = State S0

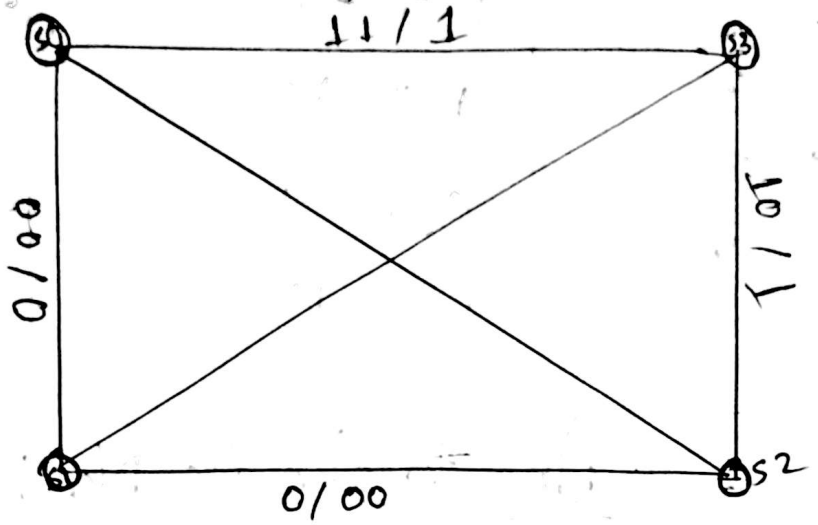
01 = State S1

10 = State S2

11 = State S3

~~2. Identify Transitions~~

~~For each row, the present state~~



2. Identify Transitions

For each row, the present state (A,B) and Inputs (X, Y) determ
the Next state (A, B) and output (Z).

Arrows in the diagram should represent transitions between
stats based on input conditions.

3. Draw the State Diagram

Represent each state (S0, S1, S2, S3) as a circle

Draw dricted arrows between states based on the next st
values.

Lable each arrow with the input condition (x, y) / z.