## **DAR ES SALAAM INSTITUTE OF TECHNOLOGY.**

## 

## 

# **DEPARTMENT OF TELECOMMUNICATION AND ELECTRONICS.**

# **DIGITAL ELECTRONIS**

# **ETU 07425**

# **ASSIGNMENT** **1**

# **BASILISA KORNEL SEHEMU**

# **REG NO: 220242423630**

**ASSIGNMENT 1**

1. The operation of a gated S-R Flip flop.

In digital electronics, a gated SR flip-flop is a basic memory component that stores a single bit of data. Additional gating signals are incorporated into the basic SR (Set-Reset) flip-flop to regulate when the flip-flop is permitted to change states. The following are a gated SR flip-flop's primary functions:

1. Initially, the flip-flop is in a known state, either set or reset, depending on the initial values of the S and R inputs.
2. When the enable input (EN) is high, the flip-flop can be set or reset by applying appropriate logic levels to the S and R inputs. If S is high and R is low, the flip-flop is set; if R is high and S is low, the flip-flop is reset.
3. When the enable input (EN) is low, the flip-flop is "latched" in its current state, and the S and R inputs are ignored. The flip-flop will remain in this state until the enable input is raised again.
4. The output of the flip-flop, labeled as Q, represents the current state of the flip-flop. When the flip-flop is set, Q is high; when the flip-flop is reset, Q is low.
5. The complement of the output, labeled as Q̅, is the opposite of Q. When Q is high, Q̅ is low; when Q is low, Q̅ is high.
6. The operations of JK flipflop.

A JK flip-flop is a sophisticated and adaptable memory component used to store a single bit of data in digital electronics. The SR flip-flop's problem with an incorrect state is fixed by an expansion of the basic SR flip-flop. The JK flip-flop has two outputs, Q and Q' (complementary Q), and two inputs, J (corresponding to Set) and K (corresponding to Reset). Additionally, a clock input is present to regulate the timing of flip-flop state changes. The following are a JK flip-flop's primary functions:

1. Initially, the flip-flop is in a known state, either set or reset, depending on the initial values of the J and K inputs.
2. When J and K are both low, the flip-flop maintains its current state. This means that if it was set, it will remain set; if it was reset, it will remain reset.
3. When J is high and K is low, the flip-flop is set. This means that the Q output is high and the Q̅ output is low.
4. When J is low and K is high, the flip-flop is reset. This means that the Q output is low and the Q̅ output is high.
5. When J and K are both high, the flip-flop toggles its state. This means that if it was set, it will be reset; if it was reset, it will be set. This behavior is sometimes referred to as "flip-flop" or "toggle" mode.
6. The output of the flip-flop, labeled as Q, represents the current state of the flip-flop. When the flip-flop is set, Q is high; when the flip-flop is reset, Q is low.
7. The complement of the output, labeled as Q̅, is the opposite of Q. When Q is high, Q̅ is low; when Q is low, Q̅ is high.

3. **The difference between the D flip flop and other flipflops.**

Another sort of memory component in digital electronics that is used to store a single bit of data is the D flip-flop, also known as the Data flip-flop. In terms of input setup, simplicity, and behavior, it differs from other flip-flops such the SR and JK flip-flops. The D flip-flop differs from other flip-flops in the following significant ways:

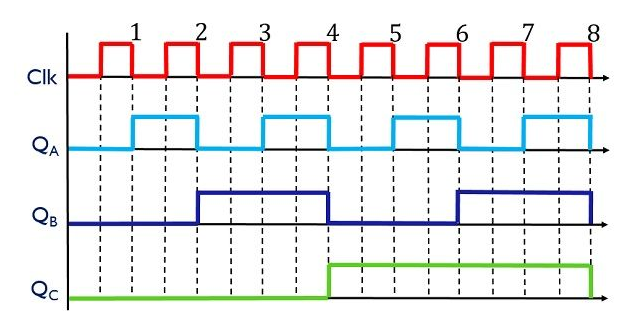
* The input and output behavior of the D flip-flop differs from that of other types of flip-flops. A D flip-flop has one input labeled D, which stands for "data." Depending on its present state, this input is utilized to set or reset the flip-flop. The flip-flop is set when the D input is high; when the D input is low, the flip-flop is reset.
* The output behavior of the D flip-flop differs from that of other types of flip-flops. The output Q of a D flip-flop follows the input D, shifting to match the input as the flip-flop is timed. Other varieties of flip-flops, such as the JK flip-flop, which toggles its output when both inputs are high, or the SR flip-flop, which can have indeterminate output when both inputs are high, may have more sophisticated output behavior.
* The D flip-flop is a popular choice for many applications since it is simple and basic. Other forms of flip-flops may be more complicated, but they provide extra utility or advantages in specific situations.

1. The timing diagram of the asynchronous and synchronous counter.
2. The timing diagram for the asynchronous counter (consider a 3-bit asynchronous counter):

Where the output of each flip-flop is connected to the next flip-clock flop's input

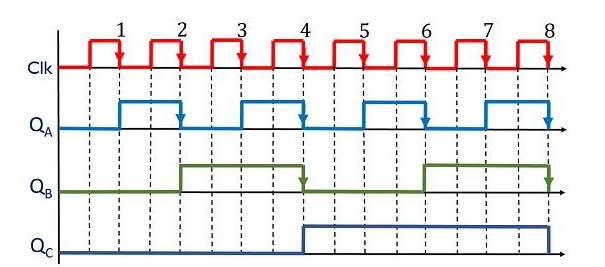
The clock signal is a square wave that changes state from low to high and back again, as shown in the timing diagram. Every clock pulse causes the outputs of the first flip-flop, QA, to change state. Because its clock input is coupled to the QA output, the output of the second flip-flop, QB, changes state every second clock pulse. Similarly, because its clock input is coupled to the QB output, the output of the third flip-flop, QC, changes state on every fourth clock pulse.

The timing diagram demonstrates that the counter's outputs follow a binary counting sequence: 000, 001, 010, 011, 100, 101, 110, However, we can also see that there is a delay between the first and last output change due to the ripple effect. This delay is known as the propagation delay, and it limits the maximum operating frequency of the counter.



1. The timing diagram for the synchronous counter (consider a 3-bit synchronous counter):

The clock signal is a square wave that changes state from low to high and back again, as shown in the timing diagram. The flip-flop outputs change state with each clock pulse, resulting in a clean and predictable output waveform.

According to the timing diagram, the counter's outputs follow a binary counting sequence: 000, 001, 010, 011, 100, 101, 110, 111, and then repeat. Because the synchronous counter does not suffer from the ripple effect and may run at greater frequencies than the asynchronous counter, it is better suited for applications requiring exact timing and precision.