Combinational Logic:

* Output is a pure function of its current inputs
* Output doesn’t change based on how many times the logic is triggered - idempotent
* AND gates

Sequential Logic:

* Output depends on current inputs, past inputs, and their history. The previous information is fed into the system
* OR gates

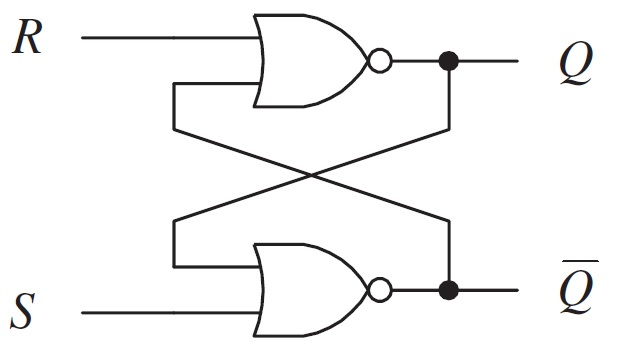
Design a circuit with 1 bit of memory

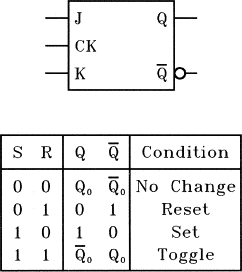
Simplest form of a circuit that can store a bit of information is a latch

2 Types of sequential circuits; synchronous and asynchronous,

* Synchronous depends on the computer’s clock; steps happen based on the outputs of this clock, kind of like a metronome
* Asynchronous doesn’t need a clock and is based off of its own time
  + In gate - type asynchronous systems, the storage elements consist of logic gates whose propagation delay provides the required storage
* Storage elements used in clocked sequential circuits are called flip flops
* A flip flop is a binary storage device capable of storing a bit of information

RS(SR) Bistable





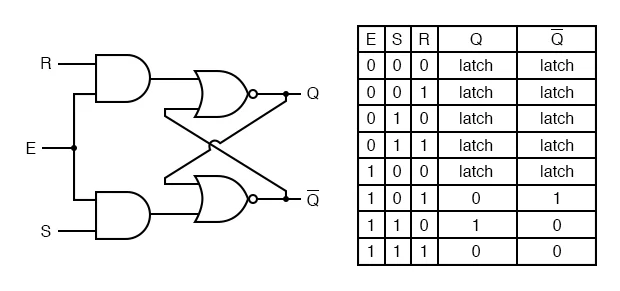
* Cross connected nor gates (outputs feed back into inputs)
* When Set = 1, Q should be forced to 1
* When Reset = 1, Q should be forced to 0
* When neither are 1, Q should remain at its present value (Hold state)
* Both cannot be 1 because it would cause Q and Q’ to be oscillate repeatedly

Criteria for a bistable

1. Able to set (preset) => force Q = 1
2. Able to reset (clear) => force Q = 0;

Latches are bistables that include a new clock input

The clock input will tell the latch when to ignore the inputs (when C = 0), and when to respond to them (when C = 1)



D Latches

* Data latches store data when the clock is low and pass data when the clock is high
* D latch is just an SR latch with the D input run into the S input and inverted into the R input

Latches are the basic circuits from which all flip-flops are constructed

Although they are useful for storing information

Bistables, latches and flip-flops

* Bistables
  + Asynchronous
  + No clock input
* Latches
  + Asynchronous
  + clock/enable input
  + Level sensitive
    - Outputs can change anytime clock = 1
* Flip-Flops
  + Synchronous
  + Clock input
  + Edge sensitive
    - Output

Master-slave D-FF

* One way to build a D-FF is to take 2 D-latches and chain them together
* Let the clock go directly to one D-FF but be inverted into the other