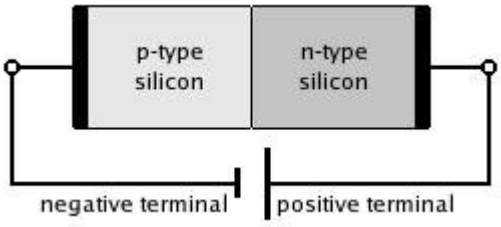
CSC258 Study Notes

Transistors

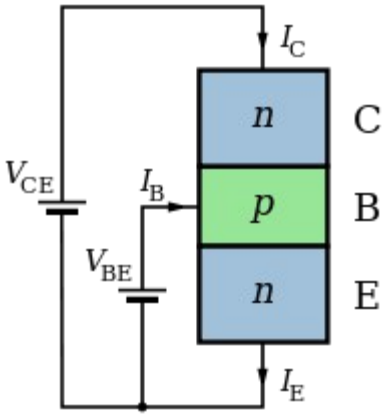
- Logic Gates made from Transistors made from pn-junctions made from semiconductors.
- Semiconductor are made from **Silicon** and **Germanium** that take the behaviour of conductor or insulator, depending on factors like temperature and impurities in the material.
- Semiconductors are solid and stable at room temperature (only allowing a weak current to flow), but energy can make electrons from the valence layer become loose.
- The valence layer is the outer-most layer where electrons can be gained or lost.
- Semiconductor do not conduct electricity naturally, so impurities are introduced to increase the number of **free charge carriers**.
- For example, Silicon (Si) has 4 electrons and Phosphorous (P) has 5. If we have a grid of Si (of 4 electrons each), then the grid will be stable. If we replace a Si with a P, then we have one free charge floating that is not allocated. (Causing it to have a positive charge).
- **n-type impurities**: adding elements from group 15, which have 5 electrons in their valence layer. (e.g. phosphorous, arsenic). The extra electrons flow freely through the material.
- **p-type impurities**: adding elements from group 13, which have 3 electrons in their valence layer. (e.g. boron). The carriers are instead called **holes**, to represent the electron gap as a particle as well. The holes can move from element to element. Holes give a positive charge.
- This is called **doping** the semiconductor.
- **p-n junctions.**



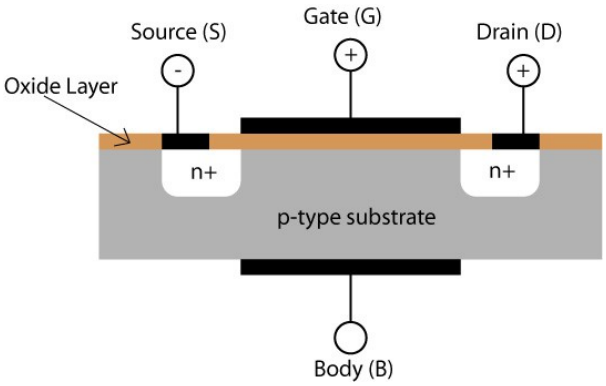
- Since the p-type silicon has extra electrons and the n-type silicon has “holes”, the two materials cancel each other out creating a particle-free section called the **depletion layer** between the materials. The specific act of electrons moving to cancel out holes is what creates the depletion.
- The depletion layer will expand, and when it gets wide enough the doping atoms that remain will create an **electric field** in that region. The reason for this is that the elements that lose

electrons to cancel out holes gain a positive charge (and a negative charge for the elements that gain the electron), and this process repeats constantly (thus the expanding field).

- The current caused by the field is called **drift current** and the current created by the electron/hole recombination is called **diffusion current**. At rest, these two currents are at equilibrium.
- **Forward Bias:** When voltage is applied to the p-type silicon, electrons are injected into the n-type section. This narrows the depletion layer and increasing the electron diffusion rate. The smaller depletion layer allows electrons to travel more easily through the junction. (AKA allow current through). This causes a **short-circuit.**
- **Reverse Bias:** When voltage is applied to the n-type silicon, the depletion region becomes wider preventing the carries from passing. A small current flows through the circuit, but is weak and does increase with an increase of applied voltage. This causes an **open-circuit** (no current flows).
- **Bipolar Junction Transistors**



- The BJT has three contact points. Collector, Base and Emitter. Increasing voltage to the base and emitter, we can make the current from **the base to collector** go from zero to higher values. This causes amplification of the current from emitter to collector.
- **Metal Oxide Semiconductor Field Effect Transistors** (MOSFETS)



- There are four main components to the MOSFET. The **Source** (n-type), the **Drain** (n-type), the **Gate** (conductive material separated by an insulator) and the **Body** (substrate layer, silicon p-type bulk).
- The Source and Drain have voltage applied, but they are separated by the p-type Body that prevents current.
- When voltage is applied to the Gate, it causes a buildup of positive charge which draws the negative charge from the substrate bulk to its surface (where the Source and Drain are), creating a channel that simulates an n-type material, allowing voltage to flow.

DESIGNING CIRCUITS

- Karnaugh Maps (for min terms)

	$\overline{C} \cdot \overline{D}$	$\overline{C} \cdot D$	$C \cdot D$	$C \cdot \overline{D}$
$\overline{A} \cdot \overline{B}$	m_0	m_1	m_3	m_2
$\overline{A} \cdot B$	m_4	m_5	m_7	m_6
$A \cdot B$	m_{12}	m_{13}	m_{15}	m_{14}
$A \cdot \overline{B}$	m_8	m_9	m_{11}	m_{10}

- The pattern for K-Maps can be figured out by looking at the truth table values.
- When drawing boxes, the number of high values inside the box must be a power of 2 (e.g. 1, 2, 4 and so on).
- Boxes can wrap around the edge of the map.
- For max terms, do the same just for the low values instead of high.

LOGICAL DEVICES

- **Multiplexers** or **muxes** take 3 inputs, X, Y and S and give one output, M. Depending on the value of S, the MUX will output the value of either X (S = 0) or Y (S = 1).

  In Verilog: **assign** M = (!S & X) | (S & Y)
- **Half Adders** take 2 inputs, X and Y; and give two outputs S and C. X and Y are binary digits, while S is the their sum and C is the carry. Does binary addition without a carry in.

  In Verilog: **assign** C = X & Y

  assign S = (X & !Y) | (!X & Y)
- **Full Adders** take 3 inputs, X, Y and C_in; and give two outputs S and C_out. (See half-adder)

  In Verilog: **assign** C_out = (X & Y) | (X & C_in) | (Y & C_in)

  assign S = (X & !Y & !C_in) | (!X & Y & !C_in) | (!X & Y !& C_in)
- **Ripple-Carry Binary Adder** are full adders linked in serial, where each full adder's C_in is the C_out of the full adder before it.
- **How to get the negative binary number** (giving a signed number):
  - Get **1's compliment** by negating each bit (1's becomes 0's and vica versa)
  - Add 1 to the end of the **1's compliment** to get the **2's compliment**.
- **How to do binary subtraction**
  - Get the 2's compliment of the term being subtracted (if A – B, get B)
  - Add the new number to minuend (if A – B, get A).
  - If there is an carry_out, the final value is positive and does not change.
  - If there is no carry_out, get the 2's compliment of the result and a negative sign to it.
- Remember, the first bit in signed numbers is whether it is positive or not. 0 means positive!
- The most negative number is a 1 followed by all 0's.
- **Comparators** compare the value of two binary numbers and return a single output if the comparison is true. The pattern for equals is easy to derive.

For example, this is the A > B comparator when A and B are 3 bits each.

assign O = (A[2] & !B[2]) | (X2 & A[1] & !B[1]) | (X2 & X1 & A[0] & !B[0])

where **assign** X# = (A[#] & B[#]) | (!A[#] & !B[#]);

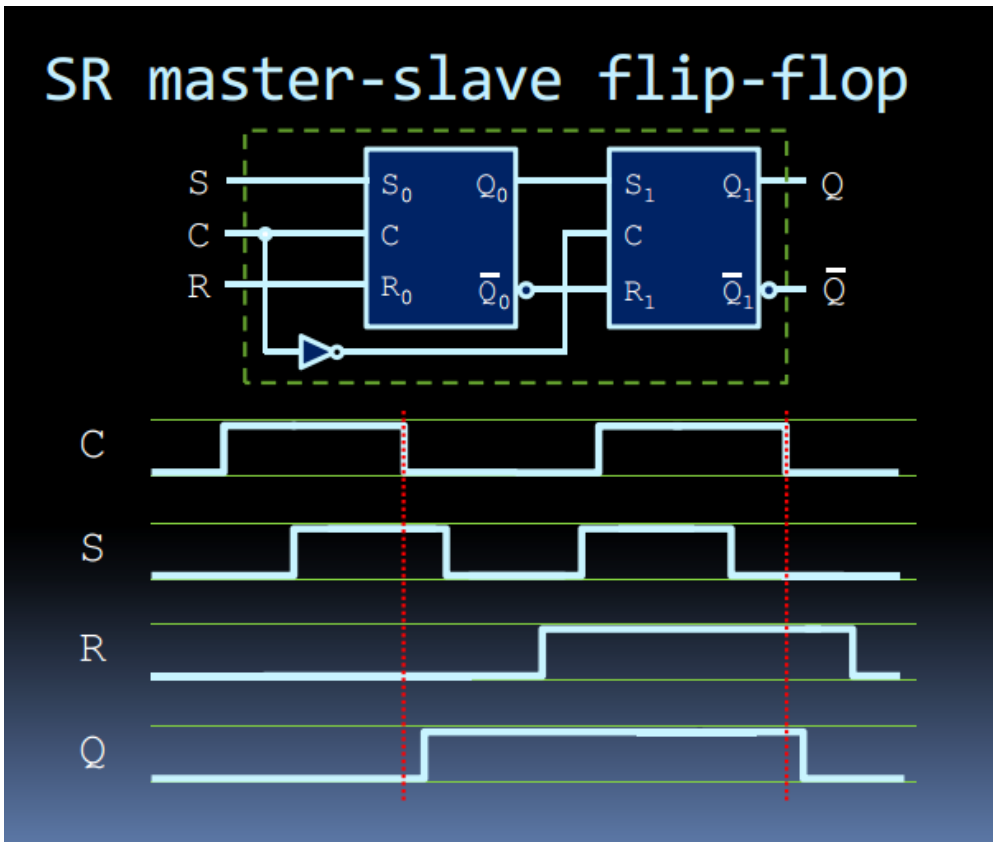
The pattern for A > B for any n bit binary numbers is:

X as above

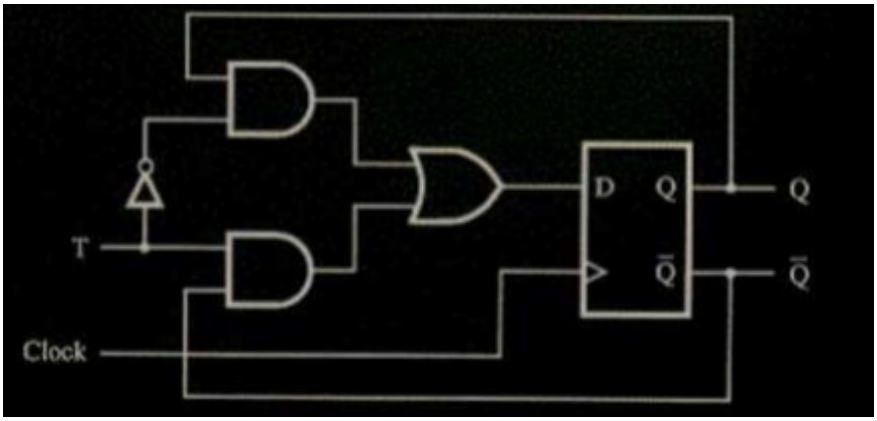
O = (A[n] & ! B[n]) | ... | (X[n...1] & A[0] & !B[0])

SEQUENTIAL CIRCUITS

- **!S!R LATCH** is made from two NAND gates.
  - Assume !S is 1 and !R is 0 at start.
  - !R sets the input of !Q to 1, which sets the output of Q to 0.
  - Setting !R to 1 keeps the value of !Q at 1, which maintains both values.
  - 00 to is a forbidden state.
  - **SR LATCH** is similar, but is made from NOR gates and the values are inverted (e.g. 11 is now the forbidden state).
- **Clocked SR Latch** uses NAND gates, but an additional set of NAND gates which reverses the negation on Set and Reset. The Latch gets only updated when C is 1. **D Latch** is derp.
- **Latch Timing Issue** – When Clock Signal is high, the output value (Q) will fluctuate between low and high,
- **Flip-Flop** is the solution to the timing issue.



- A **positive-edge triggered** flip-flop inverts the clock before the first latch. These are the most common flipflop.
- **T flip-flop** has D replaced with T, and the value of Q is toggled as long as T is high.



- **JK flip-flop** has replaces D with J and K, which respectively act like S and R.
  - If J and K are 0, maintain output.
  - If J and K are 1, toggle output.
  - S and R for J and K otherwise.