

## 1. Transistor review

As seen in lecture, two common types of transistors are NMOS and PMOS transistors. They can be used as switches that are controlled by the voltage difference across Gate (G) and Source (S) and connect Drain (D) and Source (S). They turn out to be extremely useful in digital logic design, since we can implement Boolean logic operators using switches.

Assume  $V_{tn}$  and  $V_{tp}$  are the threshold voltages for the NMOS and PMOS transistors, respectively.



Figure 1: nMOS Transistor



Figure 2: pMOS Transistor

- In an NMOS transistor, for what range of voltage differences across Gate and Source does the NMOS short Drain and Source? Essentially, what are the conditions for closing the NMOS switch?
- Assume that the voltage range is from ground to VDD. If Source of an NMOS is connected to VDD, would the switch ever close? What if it is connected to ground?
- In a PMOS transistor, for what range of voltage differences across Gate and Source does the pMOS short Drain and Source? Essentially, what are the conditions for closing the PMOS switch?
- Assume that the voltage range is from ground to VDD. If the Source of a PMOS is connected to VDD, would the switch ever turn on? What if it is connected to ground?

## 2. De Morgan's laws

De Morgan's Laws provide a method to transform Boolean expressions containing **ANDs** into Boolean expressions containing **ORs**. Essentially, they are transformation rules for Boolean expressions. They are very useful when we want to simplify logic or are constrained to use a certain logic gate.

As a quick reminder from the previous discussion, the symbols used in Boolean algebra are  $\neg$  for **NOT**,  $\wedge$  for **AND**,  $\vee$  for **OR**, and  $\oplus$  for **XOR**.

In the last discussion, you showed that binary addition can be done through Boolean operations. For one-bit binary addition  $(A)_2 + (B)_2$ , the formulas you derived were

$$C = A \wedge B$$

$$S = A \oplus B$$

where  $C$  is the carry and  $S$  is the sum.

Now, we are going to see if we really need all of those different symbols, or whether we can just write Boolean expressions using only **ORs** and **NOTs** while still being able to express everything.

- Write the truth table for  $C = A \wedge B$ .
- Flip all the Boolean values in the truth table you just made. What is the resulting Boolean expression represented by the flipped truth table? Do you need an **AND** to represent this?
- Now, let's similarly convert  $S = A \oplus B$ . First, write out  $A \oplus B$  using only **NOTs**, **ORs**, and **ANDs**.
- From the expression you just made, use the rule you learned in part (b) to remove the **ANDs**.
- Was there something special about **OR** vs **AND**? Using only **NOTs** and **ANDs**, rewrite the expression  $S = A \oplus B$ .

### 3. KVL/KCL review

To help transition into the new material that the course is about to go into, it would be very useful to review Kirchoff's Circuit Laws.

Use Kirchoff's Laws on the circuit below to find  $v_x$ ,  $i_s$ ,  $i_{in}$  and the power provided by the dependent current source. You can use  $R_1 = 2\Omega$ ,  $R_2 = 4\Omega$ , and  $R_3 = 2\Omega$ . To help with solving the problem, we have already found the voltage difference across  $R_1$  and  $R_3$ .

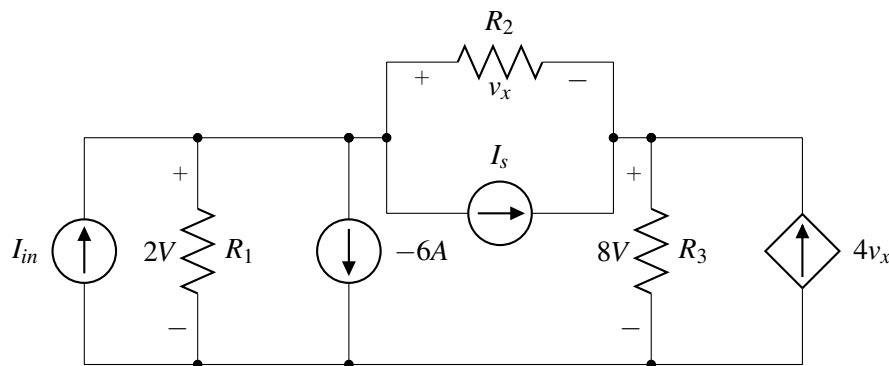


Figure 3: Example Circuit

- What is  $v_x$ ?
- What is  $I_s$ ?
- What is  $I_{in}$ ?
- What is the power output of the dependent current source on the far right?