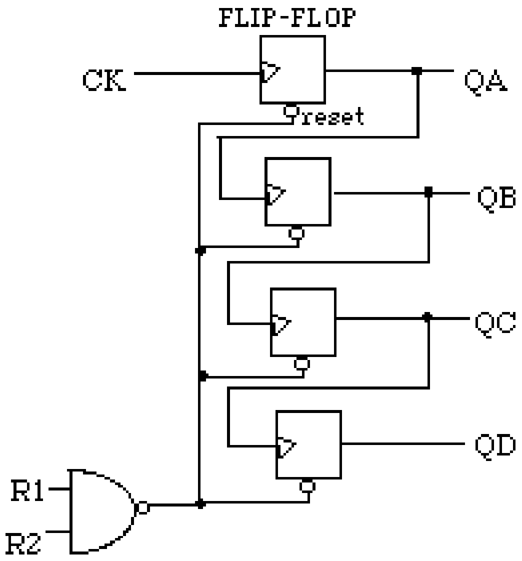
CS 240 Laboratory 2 Notes
Numeric Representation and Basic Gates

Laboratory 2 uses the binary counter introduced in Lab 1, along with other digital and discrete devices, to illustrate concepts of numeric representation that have been discussed in lecture. It also introduces the oscilloscope as a laboratory tool, and the transistor as a discrete device for implementing basic logic gates.

Binary Counter

In lab last week, you learned that the outputs of the **binary counter** device represent the sequence of binary numbers from 0 to 15. The output of the circuit was incremented each time an input clock signal changed from high voltage to low voltage, or from 1 to 0 (which occurred when you pushed the pushbutton switch and then released it).

You may have wondered what was inside the binary counter to make it act this way. Internally, the binary counter consists of 4 devices called **flip-flops**, connected in the following fashion:



Each flip-flop (a device which will be discussed in lecture in a couple weeks) is a tiny memory, capable of storing 1 bit of data, a 1 or a 0.

The value being stored (the output QA for the first flip-flop) changes or flips from the previous value when the clock input CK (provided by the pushbutton switch in our circuit last week) changes from a high to a low value (the device is activated on the 'falling edge' of the clock input pulse).

Notice that QA, the output of the first flip-flop, provides the clock for the second flip-flop, and similarly for the third and fourth flip-flops.

The flip-flop may also be cleared, or set to a 0 value, by inputting a 0 to it's reset input. This type of input is known as **active low**, meaning that it is active when a low signal is applied. Active low inputs are indicated by a clear bubble, as shown in the flip-flop diagram.

The reset from all 4 flip-flops is tied to the same wire, which is the output of a circuit which has the R1 and R2 signals as inputs. The **truth table** (list of expected outputs for all possible input combinations) for this circuit is:

R1	R2	reset
0	0	1
0	1	1
1	0	1
1	1	0

In other words, the output of the gate (which feeds into the reset input on all 4 flip-flops) is only 0 when R1 and R2 are both 1, which means that the flip-flops will only be reset when the inputs R1 and R2 are both set to 1.

In lab last week, the R1 and R2 inputs of the binary counter device (pins 2 and 3) were both tied to ground. This means that the outputs of the binary counter were not being reset. Therefore, the circuit always counted up to the maximum value of 15.

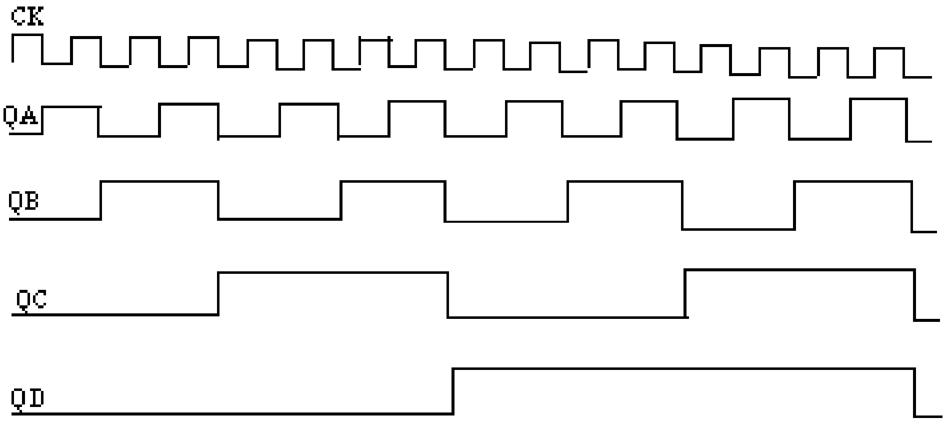
However, the count sequence of the device can be modified by setting R1 and R2 to 1 at a particular count. This resets the binary counter to 0000 at that count, and starts the count sequence over again.

For example, if the reset occurs after 9, the circuit will count from 0 to 9 over and over again, instead of from 0 to 15. Such a circuit is called a **binary coded decimal (BCD)**, or decade, counter. In lab today, you will learn how to set R1 and R2 to 1 at the correct time to modify the count sequence for the binary counter.

Below is a truth table for the binary counter; CK is the input, and QD - QA are the outputs of the device:

CK	QD	QC	QB	QA
0	0	0	0	0
1				
0	0	0	0	1
1				
0	0	0	1	0
1				
0	0	0	1	1
1				
0	0	1	0	0
1				
0	0	1	0	1
1				
0	0	1	1	0
1				
0	0	1	1	1
1				
0	1	0	0	0
1				
0	1	0	0	1
1				
0	1	0	1	0
1				
0	1	0	1	1
1				
0	1	1	0	0
1				
0	1	1	0	1
1				
0	1	1	1	0
1				
0	1	1	1	1

Graphically, this truth table may be represented by the following (assume voltage is the vertical axis and time is the horizontal axis):



Today in Lab, you will be using the TTL output of the function generator to produce a signal that constantly changes values from high to low. This changing signal will be used as an input clock signal to the binary counter. Every time the input clock signal changes, the output of the binary counter will be incremented (this will happen automatically; you won't have to push the switch).

As the outputs of the binary counter change over time, the resulting input and output values can be observed on an **oscilloscope**. The oscilloscope is an instrument that can graphically display signals which change with time. The signals you observe should verify the truth table, and should be similar to those shown above.

Binary Arithmetic

You have learned in lecture to use **two's complement** to represent binary numbers . Two's complement has a sign bit (0 for a positive number and 1 for a negative number) in the leftmost digit. Therefore, the largest value which can be represented by n digits is $2^{n-1} – 1$, since the n^{th} digit is used as the sign bit.

When you perform addition on two numbers represented in two’s complement, you must be careful that the result will still fit in the number of digits available to represent the resulting value. If it does not fit, it is called an **overflow**. When performing calculations on a computer, it is very important to check for overflow, to avoid incorrect results!

An overflow cannot occur when the two numbers being added have different signs. However, when the two numbers have the same signs, an overflow can be detected if the result has a different sign that the two values being added.

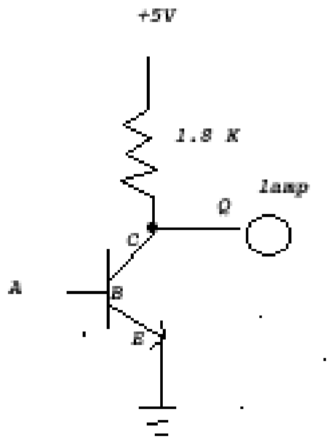
In lab today, you will construct a simple 4-bit adder, using an adder chip. You will perform some simple additions, and observe the results, with attention to carry and overflow.

Transistors

At the device level, all modern digital circuits consist of many interconnected **transistors**. A transistor is, basically, a tiny, very fast switch. It has a collector, base, and emitter (labeled B, C, and E in the diagram below). When the base is connected to a positive voltage, the transistor closes, to allow current to pass from the collector to the emitter.

When the base is connected to ground, the transistor opens (you can think of it as the emitter disconnecting from the rest of the circuit), so that current cannot flow from the collector to the emitter.

The following diagram represents a circuit which uses a transistor to produce the function Q = NOT A (Q is the complement, or opposite, of A):



When A (the input to the base) is set high, a circuit is formed which is much like the simple one we built in lab last week (consisting of positive and negative terminals of a voltage source, with a resistor between). In this circuit, measuring the voltage at Q yields a low value, since Q connects to the negative terminal of the source. So, Q is low when A is high.

When A is set low, the circuit is open (there is a break between the emitter and ground). In this case, any point in the circuit which is connected to the positive terminal of the voltage source will also be high (since no current is flowing, there is no potential difference between any connected points). So, Q is high when A is low.

In lab today, you will create some very simple circuits using transistors, observe their behavior, and record their truth tables.

You have seen that a high or low voltage can be interpreted as a 1 or a 0, so that all possible inputs and outputs of a circuit can be described using the binary number system (the truth table for the binary counter, above, is an example).

When using transistors, a high voltage (2 – 5 volts) can be interpreted as a 1, and a low voltage (0 – 1 volts) can be interpreted as a 0. This is called **positive logic** (the opposite assignment of high and low is also sometimes used, and is then referred to as **negative logic**). In lab today, you will use positive logic to interpret the behavior of the transistor circuits.