EE283 Labs 10 and 11
Elements of Digital Logic
And
Digital Project

Schedule:

This exercise will be completed in two lab sessions. The report from the gate section is not due until one week after the first lab session.

Objectives:

To understand

- The binary number system
- The basics of binary operations, and
- The functioning of digital logic gates and circuits.

The Binary Number System

Many practical variables in real life have only two states. A binary number system uses the digits 0 and 1 only and thus represents the real-world situation effectively. Some examples are: 0 - false; 1 – true (or vice versa); 0 - open switch; 1 - closed switch (or vice versa); 0 - low; 1 - high (or vice versa) and so on. In this laboratory exercise, we let 0 to represent a 'low' voltage, a voltage between 0 V and 0.8 V, and 1 represent a 'high' voltage, a voltage between 2 V and 5 V. (We assume that voltages between 0.8V and 2V occur only during transitions.)

The Binary number system is a positional number system like the decimal system we commonly use. Each digit has a different weight according to its position in the number. For example, 768 in the decimal system is equal to $7x10^2 + 6x10^1 + 8x10^0$, where we call 10 as the "base" of the decimal number system. In the binary number system, the base is 2, and so a number 1110 will have an equivalent value of $1x2^3 + 1x2^2 + 1x2^1 + 0x2^0$ or decimal value of 14. So, we need 4 bits to represent any decimal number from 0 through 15. Similarly, three bits in binary will represent any decimal number from 0 through 7, five bits will represent 0 through 31, and so on.

Truth Tables

A truth table lists the output value for ALL possible values of the input variables. Let us consistently use 1 for 'true' or 'high' and 0 for 'low' or 'false'. These truth tables are used to explain the binary operations described next.

Binary Operations and Logic Gates

A "gate" is a digital electronic circuit that performs a function. A simple digital device usually includes several gates together in one package. The device is called an Integrated Circuit (IC), or informally, a "chip," since the actual circuit is a tiny microelectronic silicon device combining transistors and other components. Devices are identified by a number on the package, for example "SN74LS04N" designates a Texas Instruments (SN) 74xxxx series digital device (74) of the Low power Shottky technology (LS) that is a hex inverter (04) in a dual inline package (N). We will omit the manufacturer and package codes normally. We will refer to this device as a 74LS04. The other devices used in this laboratory exercise perform the AND (74LS08), OR (74LS32) and ADDER (74LS283) functions. The hex inverter, AND gate and the OR gate are the basic building

blocks from which most other digital integrated circuits are fabricated.

An "Output" is an electrical signal that asserts a 1 or a 0. That is, even if you have it drive a load, say, an LED, an output signal tries to keep its voltage high if it is supposed to be putting out a 1, and low if it is supposed to be putting out a zero. It is possible to load down an output so that it can't properly function. If you were to connect an output to ground, or through a small resistance to ground, the signal would not be powerful enough to assert a 1. It would appear to be a zero even though it should be a 1. Likewise, a signal connected to 5 Volt power can never show a good zero, even if it should. For the devices we are using, the manufacturer guarantees that a 1 output will be at least 2.7 volts (as long as you don't drain out more than 0.4 mA) and a 0 output will be no more than 0.5 volts (as long as you don't dump in more than 8 mA). Notice that "1" outputs are "weaker" than "0" outputs.

An "Input" is a signal that is sensed by the gate, driven by some external source, usually another gate or a switch or pushbutton, to be a 1 or a 0. When a "1" (a high voltage) is supplied, a tiny amount of current flows from the external source into the gate (no more than 20 µA). When a "0" (a low voltage) is supplied to the gate, a small amount (no more than 0.4mA) of current flows out of the gate into the external source, so the external source must be able to sink that current. (If the source cannot sink that amount of current, the voltage rises above 0.8 Volts, and the behavior of the gate becomes unpredictable.) Because of the internal electronics of the gates we are using, leaving an input unconnected results in the gate recognizing that input as being a "1". A voltmeter on an unconnected pin will usually show a voltage of 1.7 Volts. The signal is actually ambiguous, and electrical noise can easily cause it to vary enough to cause trouble. While with TTL an open, unconnected input does (usually) register as a "1", do not leave inputs floating, unconnected. In some technologies (like CMOS) the consequences can be very, very bad (destruction of the device) as has been demonstrated by students in the past. Generally inactive (high) pins are pulled high by a 1K Ohm to 10K Ohm resistor.

Hex Inverter (74LS04)

Figure 1 shows the inverter symbol, package top view diagram, and truth table. A 1 on the input results in a 0 at the output and a 0 on the input results in a 1 at the output.

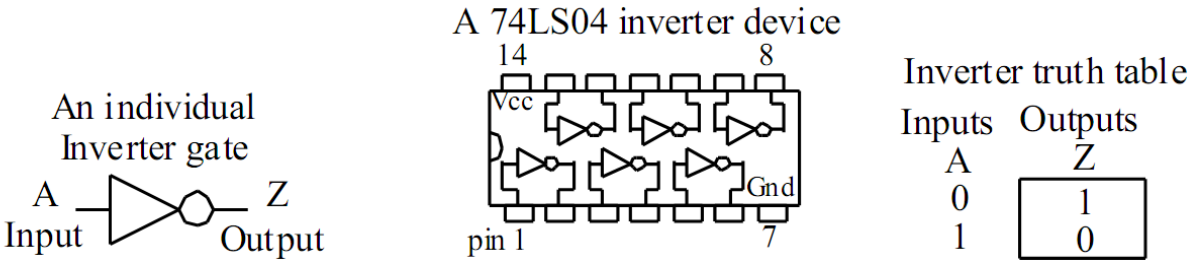


Figure 1

Quad Two Input AND Gate (74LS08)

Figure 2 shows the AND gate symbol, package top view diagram, and truth table. A 1 must be present at both the A and B inputs before the output can be a 1. A 0 on either or both inputs results in a 0 at the output.

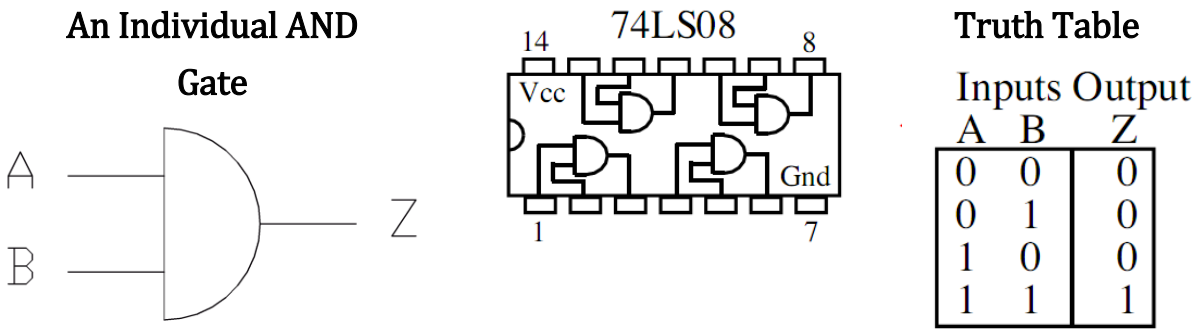


Figure 2
74LS08 Quad AND Gate

Quad Two Input OR Gate (74LS32)

Figure 3 shows the OR gate symbol, package top view diagram, and truth table. A 1 at either or both the A and B inputs results in a 1 at the output. Both the A and B inputs must be 0 for the output to be a 0.

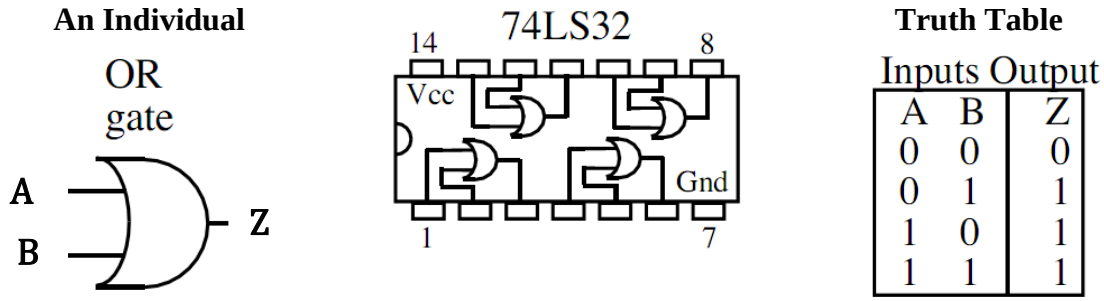


Figure 3
74LS32 Quad OR Gate

Procedure For Gates:

The +5 volt power and ground connections for the 74LS04, the 74LS08 and the 74LS32 are all the same so to verify the truth tables for each of these IC's connect the +5 volt power and ground connections as shown in Figure 4. Then insert the IC's one at a time into the breadboard to verify the truth table for each IC. Use the DMM to measure the 1 or 0 (+5 volts or ground) at the output of each gate (you only have to verify the truth table for one of the gates in each IC). Use jumper wires connected to the gate input or inputs and connect them to either +5 volts (a 1) or ground (a 0) to verify the IC truth table. The instructor or his assistant must verify the truth table for each IC.

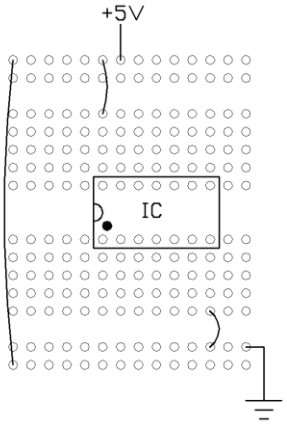
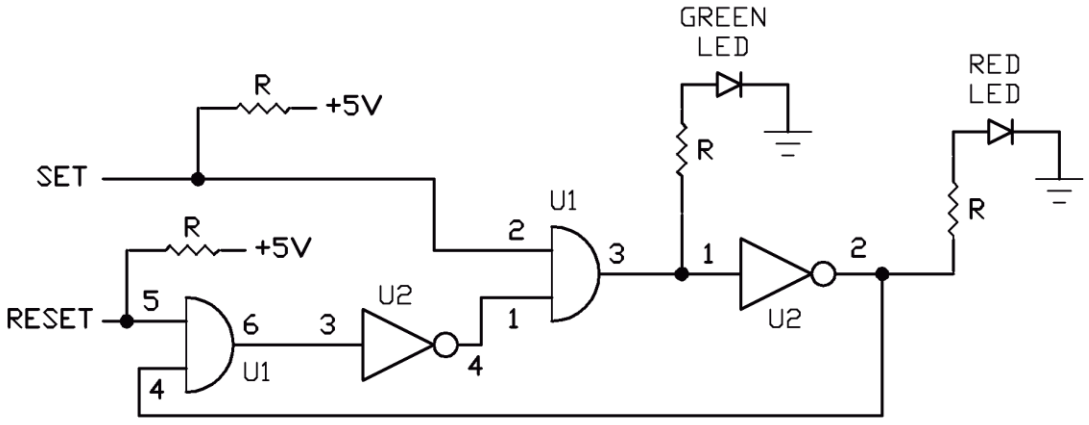


Figure 4
Truth Table Verification Setup

Construct the simple latch circuit shown in Figure 5 on your breadboard. Use the layout shown in Figure 7 for the construction. The dark lines in this figure are wires. All of the resistors are 470 ohms. Observe what happens when the Set input is touched to ground. Then touch the Reset input to ground and observe what happens. The correct circuit operation must be demonstrated to the instructor or his assistant.

Using the truth tables shown in Figures 1 and 2 and the schematic shown in Figure 5 complete the timing diagram shown in Figure 6 starting with the Set and Reset inputs set to a logic 1 and the output of U2-2 (pin 2 of U2) at a logic 0. **<u>You must submit a report (it does not have to be a formal report) showing Figure 5 and a completed Figure 6. The report is due Nov 21st.</u>**



U1 is a 74LS08 and U2 is a 74LS04
Figure 5
A Simple Latch Circuit

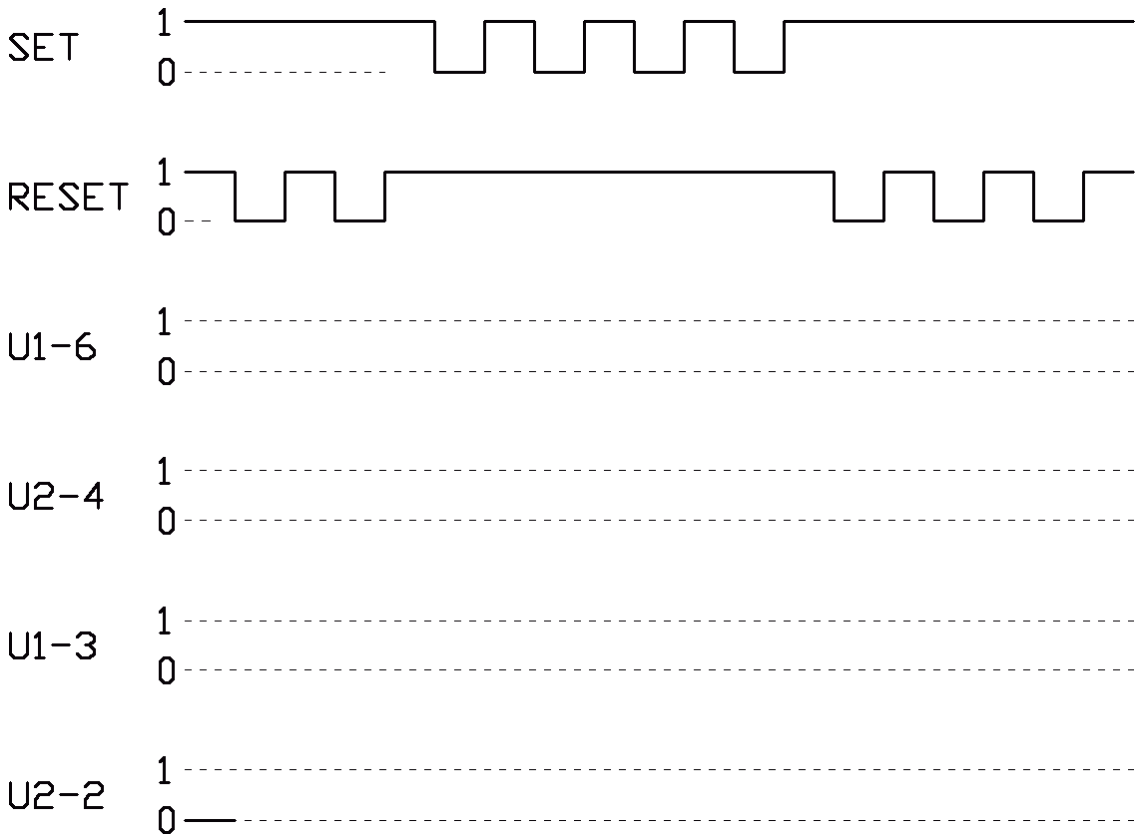


Figure 6
Timing Diagram For The Simple Latch Circuit

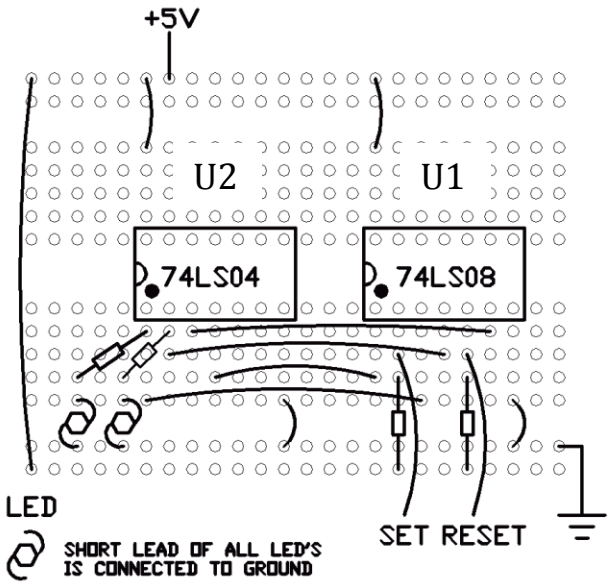


Figure 7
Layout For The Simple Latch Circuit

Binary Adder:

One of the primary functions of a computer is to add numbers. Since the computer uses only 1's and 0's in the binary number system a circuit must be designed to add binary numbers. Suppose we want to add two numbers 5 and 7. The binary number for each is shown in Figure 8.

MSB LSB

$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$	Number
0	1	0	1	5
0	1	1	1	7
1	1	0	0	12

Figure 8
Binary Number For Digits 5, 7 and 12

The position of the each binary digit is important. As shown in Figure 8 the leftmost digit (MSB Most Significant Bit) has a value of 8, the next digit has a value of 4, the next digit has a value of 2 and the rightmost digit (LSB Least Significant Bit) has a value of 1. Therefore:

The binary number for 5 is 0 1 0 1 0+4+0+1=5
The binary number for 7 is 0 1 1 1 0+4+2+1=7

To add binary numbers the following rules apply:

0+0=0
0+1=1
1+1=0 and carry 1 to the next positon

So to add the binary numbers

```
 1  1  1
 0  1  0  1    5
 0  1  1  1    7
---------------
 1  1  0  0   12
```

A four bit binary adder is a complicated circuit so we are going to use the 74LS283 which is a four bit binary adder in one integrated circuit. The internal circuitry for the 74LS283 is shown in Figure 9.

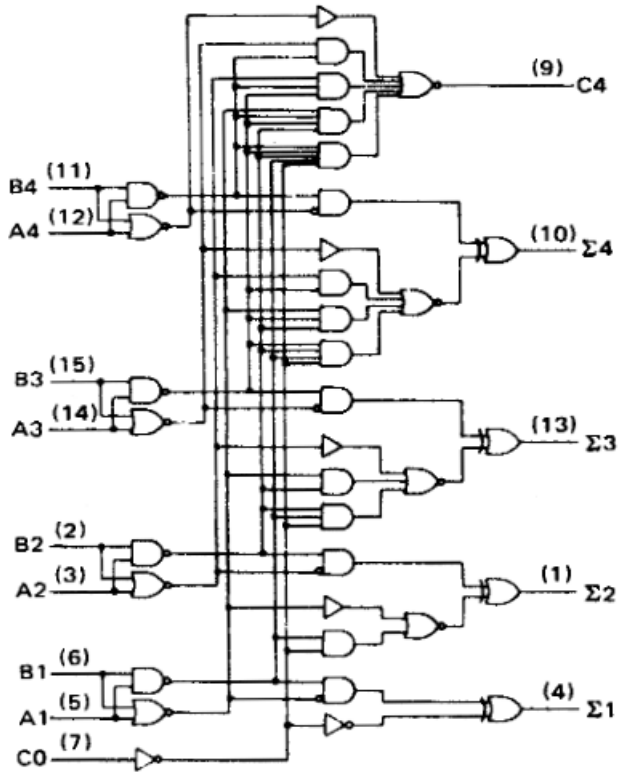


Figure 9
Internal Circuit For a 74LS283 Adder

Procedure For Four Bit Binary Adder:

Construct the circuit shown in Figure 10. All of the resistors are 470 ohms. The dark lines in this figure are wires. There are additional wires (not shown in Figure 10) that must be added. There are three points marked A1 in Figure 10. These points must be connected together. The same goes for the points marked A2, A3, A4, B1, B2, B3 and B4. The DIP switch has 10 single pole single throw (SPST) switches. The four leftmost switches represent the binary number of the first number to be added (the A number). Closing a switch will illuminate the corresponding A number LED. The four rightmost switches represent the binary number of the second number to be added (the B number). Closing a switch will illuminate the corresponding B number LED. The two middle switches are not used.

A list of the numbers to be added will be given to each student. The adder must be demonstrated to the instructor or his assistant. No report is required for the Adder section of this Lab Exercise.

Extra Credit +30%

I will allow two students to combine their breadboards to add eight bit binary numbers with a sum as high as 256. If successful each student will receive the extra credit.

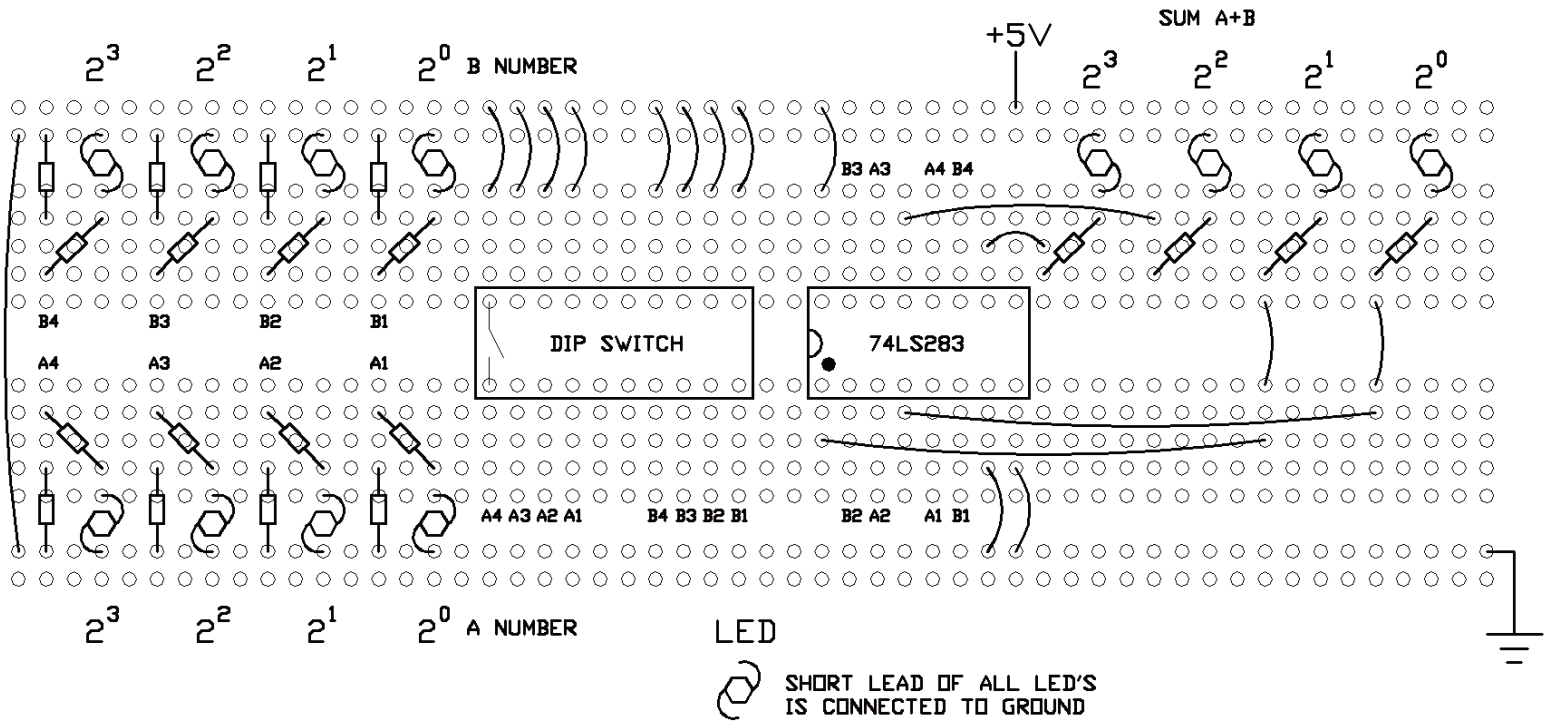


Figure 10
Circuit Layout For The Four Bit Adder

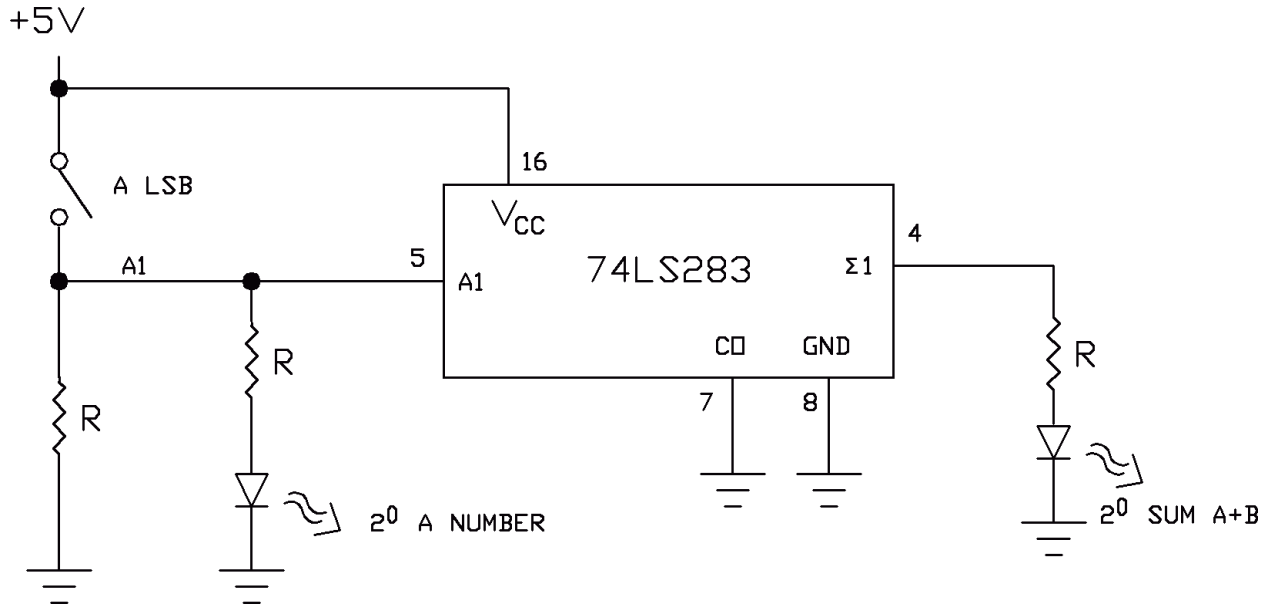


Figure 11
A Simplified Schematic For 1 Bit Of The Four Bit Adder

EE283 Laboratory Exercise 10 Form Report

Name:_____ Section:_____ Station:_____ Date:_____

Show figure 5 here

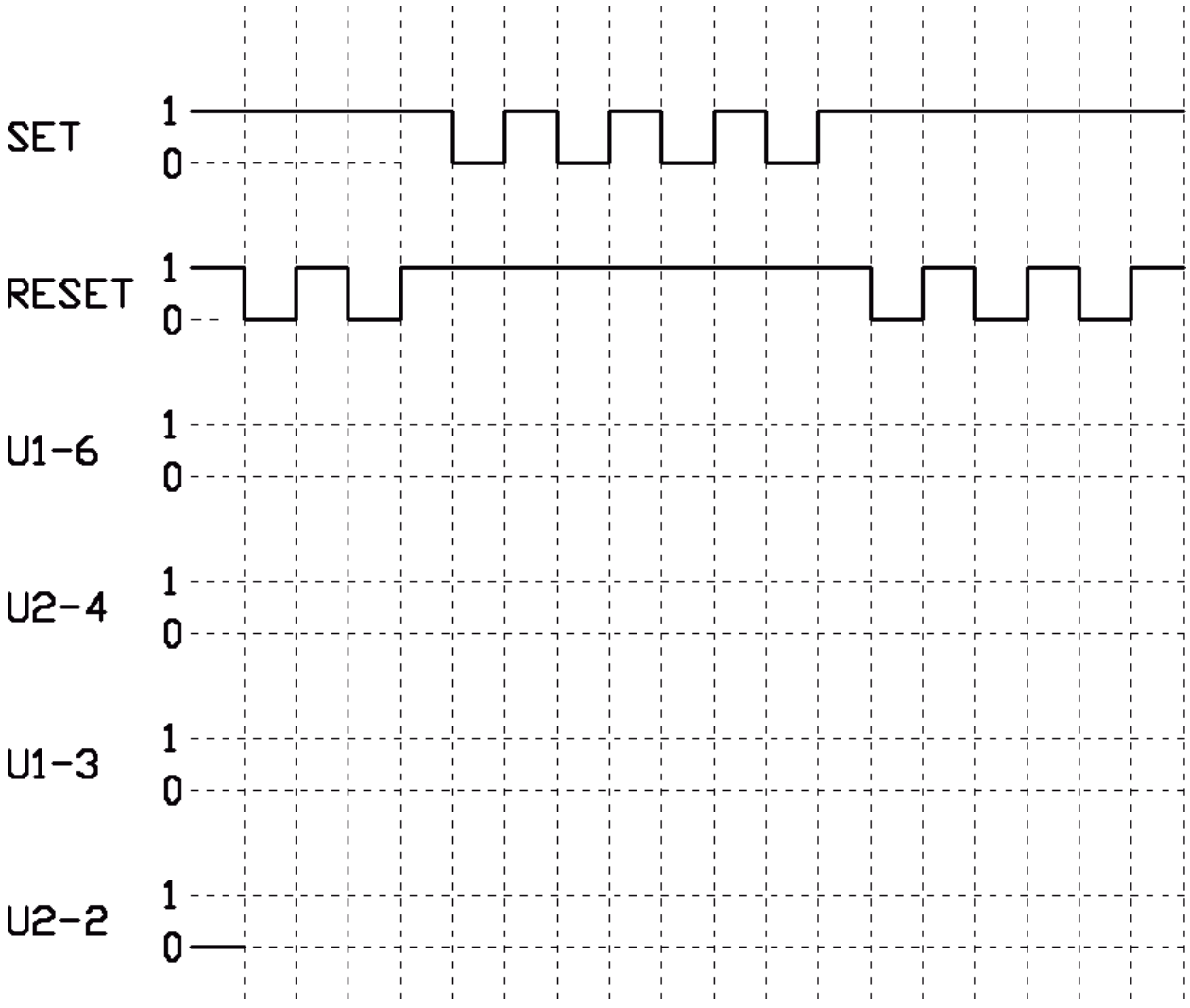


Figure 6
Timing Diagram For The Simple Latch Circuit