CSC 258 lab notes, Fall 2008

Successful completion of the three graded labs in this course involves a significant amount of preparation work. Much of your grade will be based on your preparation. Prior to the lab, you will design circuits and plan how to wire them up. During the lab, you might make changes to your plan based on comments from your lab TA; then you will do your wiring, and test and debug your circuit. You will not have time during the lab to do substantial circuit design nor even to decide which chips and pins to use, so good preparation is essential.

Each graded lab has a handout, which is like a problem set or assignment. It poses problems, the solutions to some of which involve circuit design, the solutions to others of which are written. In this way, the lab handouts are similar to assignment handouts.

However, for the lab handouts you will also be building the circuits which you have designed. That is what this document is about.

Lab teams

You are expected to form teams of two students for the purpose of working on the labs. Try to find a lab partner as soon as possible. Your team members must both sign up for a full set of lab sessions. (If there are an odd number of students in the course, one person will work alone; other than that, you *must* form teams of two.)

We strongly recommend that you exchange e-mail addresses, telephone numbers, and all other contact information with your lab partner. You have a responsibility to your lab partner to inform them as early as possible if you will miss a lab or if you drop the course.

Implementation plan

Building circuits involves many practical issues which we ignore when initially sketching our logic diagrams. Experience with these issues is the purpose of the lab.

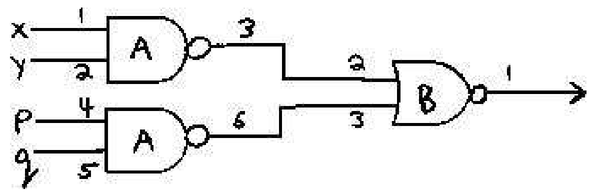
You will be using gates and other circuits on SSI and MSI chips. These chips require power and ground connections in addition to the connections implied by the logic schematics, and gates may not be available in precisely the configurations implied by your schematics—you may want a three-input OR gate, but have to use a four-input gate, or three two-input gates.

A companion document lists the chips available in the lab. In order to construct your circuit, you first need to decide which gate to use in each chip. You should make a list of chips needed, give them names of “A”, “B”, etc, and label each gate in your logic circuit with one of these letters. Then you should label each wire going into and out of each of these gates with a pin number. (See example, below.)

If you are not clear about the exact properties of one of the chips described on the data sheets, make a note and be sure to ask your lab TA for clarification at the beginning of the lab.

(continued)

Example schematic with implementation plan:

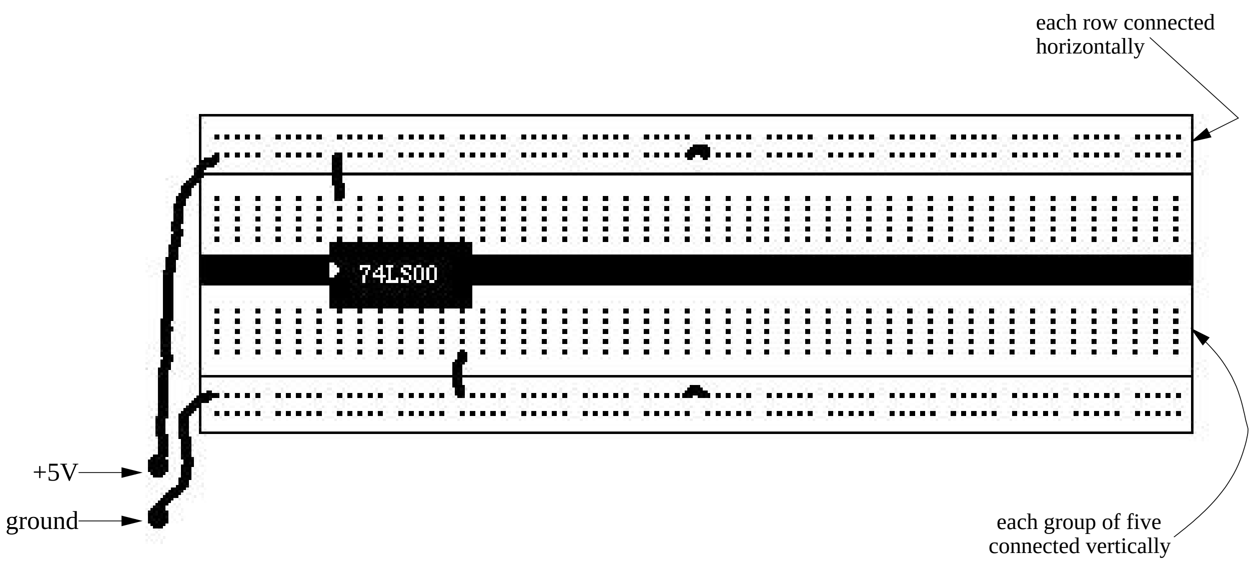


Pkg	Type
A	74LS00
B	74LS02

Wiring principles

After you have planned which chips to use for which gates, you are ready to go to the lab and construct your circuit. You will do your wiring on a ‘‘breadboard’’, to which you will be introduced in the lab familiarization session. This device allows for easy connections of wires and chip pins by inserting the wires and chips in holes in the breadboard. Certain groups of holes are connected together underneath.

The key to understanding the breadboard is to understand which groups of holes are connected to each other. The breadboard consists of three rows of the following structure:



The horizontal rows of groups of five horizontal holes are typically used for power and ground. Each horizontal row is fully connected, except that there is a break right in the middle, so you will have to connect across this, as shown in the diagram.

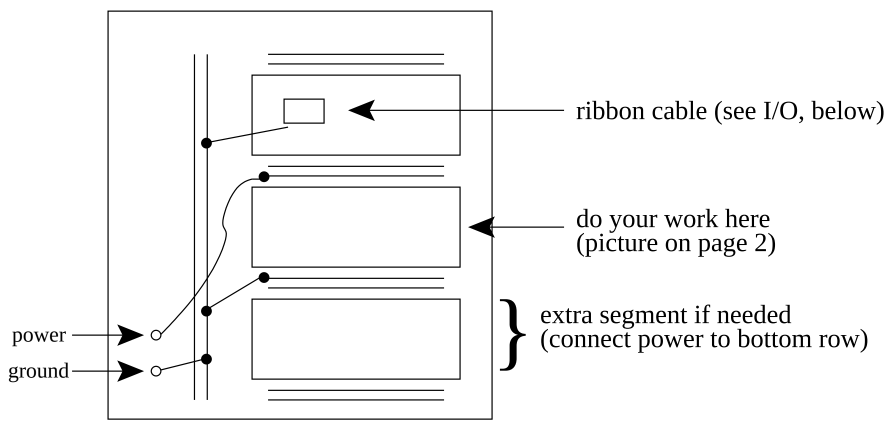
There are also two sets of groups of five vertical holes, these two sets of hole groups separated by a gap. Each group of five holes is itself connected together, but none of these groups are connected to each other. This gap is constructed so that a chip will fit neatly straddling the gap, as shown in the above diagram. Each pin of the chip connects to a different group of five holes, allowing up to four wires to be connected easily to each pin.

Your circuit is constructed by inserting the chips required into this breadboard, and using wires to connect the inputs and outputs of the chips. You will also need to supply power and ground to your chips. There is a notch on one end of the chip. With the notch on the left (as shown above), pin 1 is directly below the notch, and the pins are numbered counter-clockwise. Some chips have 14 pins and some have 16. The power pin is always the last pin, directly above the notch (i.e. pin 14 for a 14-pin chip, or pin 16 for a 16-pin chip), and the ground is always diagonally opposite the power pin (i.e. pin 7 for a 14-pin chip, or pin 8 for a 16-pin chip).

Since there are horizontal rows both above and below the vertical row areas, the best way to run power and ground for each chip is to connect the row below to ground and the row above to power, as shown above. This way, short wires from the horizontal strips to the appropriate chip pins can connect power and ground. If you use short wires, and if you connect chips' power and ground wires first, then you are much less likely to knock out the power to a chip when moving wires around in the debugging stage. Connecting power to ground is very bad (destroys power supplies); to decrease accidents, use only one of each pair of horizontal rows, as done above.

You can only ignore individual chip pins if the data sheets make it clear that they involve portions of the chips you are not using. For example, if you are using an OR chip with three OR gates but require only two of the gates, you can safely ignore the pins related to the third gate; however, if you are using a five-input OR chip to OR together only four values, you cannot ignore the fifth input. Since the identity for OR is *false*, you would ground this fifth input, representing a logic "0". (For an AND, you would supply power to the fifth input, representing a logic "1", because *true* is the identity for AND.) Similarly, a RESET line which you are not using cannot be left unconnected, or you may get spontaneous resets; if you do not want to use the reset feature of an MSI chip, you would ground the RESET line. In other words, every gate whose output you use must have *every* input connected to *something*. You can only ignore inputs if you ignore all outputs of the gate.

Suggested overall breadboard layout:



I/O

Your circuits will not require sophisticated input and output facilities, but they will require *some* I/O or you will not be able to interact with them, nor will you know what they are doing!

The I/O facilities we will be using consist of a number of input switches plus a number of output LEDs (lights). This part of your lab kit is referred to as the "digital i/o board" or "digital board". Although the input switches and output LEDs on the digital i/o board are vertically aligned, there is no connection between them unless you make one.

You will attach a "ribbon cable" from the digital i/o board to your breadboard, as shown in the figure on the previous page, using an adapter which plugs into the breadboard just like a chip, straddling the gap. The list of its "pin-outs" is on a page attached to the data sheets which describe the chips. If the notch of the adapter is facing downwards, then the odd-numbered pins are on the bottom and the even-numbered pins are along the top (this numbering is different from the chip pin numbering! — both are standard, unfortunately). Pin 1 is in the lower left, and pin 2 is in the upper left. 8 of the pins on the bottom are grounds (see the pinouts list); all of these ground pins are connected to each other, but you must connect the ground on the breadboard to the ground on the digital i/o board, via the ribbon cable, as shown in the figure on the previous page.

The ribbon cable adapter is wider than a chip; you have to be sure to leave one row of holes above it and one row of holes below it when plugging it in to the breadboard. Be very gentle when removing the adapter from the breadboard! Rock it gently; be careful not to break the pins. Be very patient.

The input switches have two positions, and generate a logic 1 or 0 as they are switched on or off, respectively. An LED *under* the input switch (i.e., towards you) indicates whether the switch is on or off. If you do not need a given input switch, you can leave it disconnected.

The output LEDs will light if they are supplied with a logic 1, and will be dark if they are supplied with a logic 0. (If they are disconnected, they will light; thus *they default to being lit*.) Again, if you do not wish to use a given LED, you can leave it disconnected.

Thus these I/O facilities are suitable for the input and output of individual bits or of small binary numbers.

The I/O board has additional facilities such as a clock, but we will not need them for the labs in this course—you will want to control all on and off transitions by hand so as to be able to observe the behaviour of your circuit more fully.

Wiring techniques

It is important to be as tidy as possible when wiring your circuit. If the circuit does not work, you will need to trace where all the wires go; keep wires short. (On the other hand, don’t cut up all of the long wires; there are probably already plenty of short wires accumulating in the bottom of the box.) When inserting chips, be careful not to bend the pins: check that all of the pins are lined up with the holes in the breadboard before pushing down. Be careful to get chips the right way around (notch or non-centred small circle at the left). Ask your lab TA for instructions for stripping wires so that the exposed (uninsulated) portion at the end is of reasonable length: if it is too short, it won’t go into the breadboard far enough and won’t make a good connection; if it is too long, it will more easily come into contact with adjacent wires, making unintended connections.

Make your connections with the power off. After you turn on the power, if some of your chips get very hot, they may be wired wrong (e.g. ground to the power pin and power to the ground pin). If something seems unusually hot, if you see sparks, or if you smell burning plastic (seriously!), turn off the power immediately. If you can’t then find the problem, ask your lab TA for assistance; don’t turn it back on when you know that there’s an electrical problem.

Some people (such as I) find it difficult to keep track of the wiring on the breadboard at the same time as you are following your schematic diagram in your lab writeup. One technique is to have one person reading (e.g. ‘‘connect pin 3 of chip A to pin 2 of chip B’’) while the other person makes the connections.

Pay close attention to your lab TA's explanation of the proper technique for removing chips. Do not pull the chips out of the board without using the chip-puller. If you pull the chips up with your fingers, you will *frequently* bend or break the pins—the chip comes up too suddenly.

If you think you have discovered a broken chip, do not put it back; let your TA know so that they can decide whether to throw it out.

Here is a recommended procedure for wiring your circuits:

- Make sure that the power is off.
- Put all of your required chips on the board, thinking about their placement so as to minimize the number of long wires.
- Connect the power and ground to all chips.
- Briefly test each chip to ensure it is not obviously broken.
- Build your circuit in stages, testing each stage as you complete it. For large circuits, in your planning stage you should partition the circuit into smaller pieces which can be tested individually. They can also be *graded* individually; get your partial circuits graded before you continue, so that if you fail to complete the lab you still get partial marks.
- As you make each connection, mark it somehow on your schematic. You may wish to bring some different colours of pens for this purpose, to assist in the possible rewiring of sections of your schematic.

Preparation goes a long way. Your chance of successfully completing the lab is greatly increased if you are well-prepared.

Debugging

You should try to debug your circuit by yourself, but if you are having lengthy problems, ask your lab TA for assistance. Of course, they may respond with hints rather than direct answers.

The key to debugging is to isolate the problem. Use your ingenuity; try things out; reject hypotheses. Here are some specific ideas.

The logic probe is your friend. It will indicate whether a given wire shows logic 1, logic 0, or is not connected. Do not push the logic probe into the holes on the breadboard; this will enlarge the holes, and then when a wire is inserted it will no longer make a good connection. Instead, touch the logic probe to exposed parts of chip pins and wires. If necessary, insert an additional wire to allow a probe.

Begin by testing the logic probe against power and ground sources. This just takes a moment, and you have to be able to trust your debugging equipment!

Then, first of all, you may not have wired the power and ground correctly on all of your chips. This is easily checked with the logic probe (touching the power and ground chip pins themselves on all chips).

Other wiring mistakes are harder to uncover. Proceed through the circuit, looking for logic signals which are wrong (unexpected). Check inputs and outputs of each gate. As well as checking for wiring mistakes, be checking for errors in your design. Use a scientific-method process of forming hypotheses as to the possible causes of the aberrant behaviour you are observing, then attempting to reject these hypotheses by tests and observation.

Remove wires to isolate parts of your circuit. You can then hook up input switches to test smaller pieces of the circuit. (However, it is easier to start by building the pieces individually and testing them before wiring them into the rest of the circuit.)

A kind of wiring mistake which merits special note is the unintentional connecting together of two *output* pins. This can be confusing because the output from one can make the other look as though it is giving the wrong output. If an output pin seems to be giving a wrong answer but its inputs seem correct, try disconnecting all wires leading out of its output, and see if this makes the problem go away.

If you suspect that a gate or chip is broken, disconnect it from your circuit and test it individually. Similarly, if you suspect that other facilities on the breadboard or i/o board are not working as they are supposed to, test them, in isolation as much as possible. Ask your lab TA for assistance if necessary; do ask your lab TA for assistance if you have concluded that anything is indeed broken. (The lab technicians rely on reports of broken equipment; if we don't report it, it won't get fixed.)

Note that if the output LEDs are disconnected, they will be lit. Thus it is easy to confuse a disconnected output and a (perhaps unexpected) logic 1 output. Use the logic probe to tell these situations apart.

Grading

Unlike in experimental labs, you will not be contributing substantially to your lab report based upon what happens during the lab. Even if you are uncertain about how a particular circuit will behave, you will know what you are going to test, and the precise results will be a small portion of your writeup compared to your circuit design, pin assignments, and answers to other questions on the lab handout. Indeed, you will find it essential to prepare your design, pin assignments, and other plans in writing in advance.

Thus you are expected to have your lab writeup nearly complete prior to the scheduled lab time. You will be too busy doing your wiring, testing, and perhaps debugging to be able to add substantially to your writeup during the lab.

A portion of the grading of the lab, thus, will happen near the beginning of the lab session. As you are getting started building your circuits, your lab TA will come around and ask to see your team's writeup. They may have some suggestions for changes which you can consider incorporating. However, all essential parts of your lab report should be ready for grading at this early stage.

Your team must produce a joint writeup.

After you have completed, tested, and debugged your circuit, ask the lab TA to examine it. This examination will primarily take the form of your demonstrating your circuit, although the lab TA may wish to try it out too.

You will leave your completed writeup with the lab TA at the end of the lab session, unless they have already examined it sufficiently to assign a grade at that point. Label your lab report with your tutorial room or TA name in case it has to be retained for grading at the end of the lab time.