WEEK -1-

1. Objective

Design a controller for a stepper motor that will be capable of:

- Making the motor rotate with variable speed (the user should be able to adjust the rotational speed easily and without powering down the controller or the motor);
- Making the motor change direction of rotation;

2. Parts & Equipment

Part description	Part number	Quantity
Square Wave Oscillator/ Monostable Multivibrator	CD4047	1
UP/DOWN Binary Counter	CD4516	1
Quad 2-input NAND gates	CD4011	2
Hex Buffer (Inverting)	CD4009UBE	1
Quad Darlington Power Driver	UDN2540B	1
Resistor Network - 220Ω/8 140mW; Resistor in DIP	221G	1
Red LED	P300-ND, 25mA max	5
NPN General Transistor, Vceo 40V; hfe 100 min.	2N2222A	1
Capacitors (**for decoupling**)	0.1μF, 50Vdc	10
Current Limiting Resistor (for LED)	220Ω ; ¼ W	1
Trimmer Pot; Top Adjust	100KΩ, ½ W	1
Current Limiting Resistors	4.7KΩ, ¼ W	3
Capacitor	10μF, 25Vdc, 20% tolerance	1
Airpax Mechatronics Stepper Motor	26M048B2U	1
Single Pole Double Throw Switch; 1A @ 24Vdc.	202972CD	1

- **Equipment:**
  - Agillent 54261D Oscilloscope (for troubleshooting)
  - NI5411 Arbitrary Waveform Generator (optional)
  - NI4060 Multimeter (for troubleshooting)
  - 8102 Lodestar Power Supply

3. Motivation and Introduction

By now you have been taking a lot of math, physics, and chemistry courses, and you have survived the two general engineering courses. So you might be wondering, when am I going to do hands-on electrical engineering? Well, wonder no more!

In this laboratory you will build a controller for a stepper motor. You are probably not familiar with a stepper motor, what it does, and where it is used. If this is the case take some time to read appendix A of this lab handout, the title of the appendix is “Stepper Motor 101”. This lab is not intended to teach you everything there is to know about stepper motors nor is it intended to teach you everything there is to know about designing controllers for these motors. What this lab will do is introduce you to the basics of building a fairly complex system (circuit) out of smaller, simpler building blocks (in this case discrete-logic CMOS devices). You will also learn how to read and understand data sheets for electronic components, a very useful skill that you will use throughout your career.

This is not an easy lab! Start early, read through this handout before you come to class, and MAKE SURE you have a good idea of what is going on and what you need to do once you start working. If you get stuck contact your T.A. prior to your lab session, he is there to help you understand the concepts used in this lab.

4. Understanding the Big(er) Picture

If we are to build a controller for the stepper motor (and we will) we need to know how this stepper motor is controlled. At the end of this document there is a references section in which we have listed all the data sheets for all of the components used in this lab. Use the data sheet for the stepper motor (model number 26M048B2U) to find out if this is a unipolar or a bipolar motor (notice the last letter in the model number) and fill in Table 4.1 with the normal 4-Step sequence that will make the motor spin clockwise.

Step	Q0	Q1	Q2	Q3
1				
2				
3				
4				

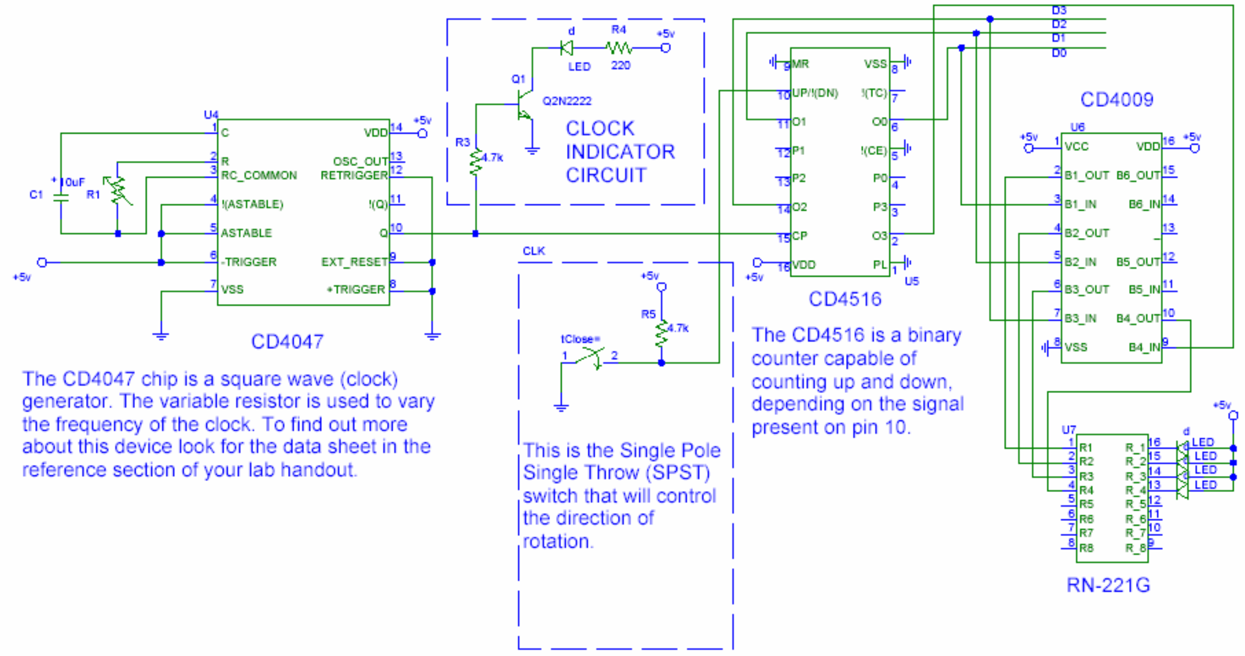
Table 4.1

Note: If you fill in this table with wrong values and use these wrong values in your design, then your circuit will not work, i.e. your motor will not rotate.

Now you understand that this motor is controlled by 4 bits of data per step. You can also logically reason and conclude that by stepping through the patterns listed in Table 4.1 you will make the motor rotate. The faster you step through the 4 sequences, the faster the motor will rotate. This brings us to the first major conclusion about the controller design! If we want the motor to have variable rotational speed (one of the requirements listed in Section 1) we must have a clock that is capable of "ticking" at variable rates. When the clock "ticks" faster, we go through the 4 steps faster, thus the motor rotates faster; when the clock "ticks" slower we achieve the opposite effect.

For a clock we have chosen to use the CD4047 Square Wave Oscillator, a.k.a. Monostable/Astable Multivibrator. This device will be configured as Astable; if you want to know more about the differences between monostable and astable configurations try www.*google.com*. Google is your friend, learn how to use it!

But a clock by itself doesn't serve any purpose. We need a counter that will count the ticks of the clock. We have chosen to use a 4-bit binary counter, CD4516**. Note that we will use the lower 2 bits of the counter, because in order to address our 4-bit data for controlling the motor we need only 2-bits ($2^2 = 4$).** Figure 4.1 shows the circuit for clock generation and binary counting. Notice that the 4 outputs from the CD4516 (the counter chip) are wired to a hex inverting buffer who's outputs are wired to LEDs through a resistor. Figure 4.2 show a more detailed picture of the CD4009, the resistor network and the LEDs. Note that this part of the circuit, the CD4009, the resistor network chip, and the LEDs, are not required for the operation of the motor, but we want you to see the actual binary counting that takes place.



The CD4047 chip is a square wave (clock) generator. The variable resistor is used to vary the frequency of the clock. To find out more about this device look for the data sheet in the reference section of your lab handout.

This is the Single Pole Single Throw (SPST) switch that will control the direction of rotation.

The CD4516 is a binary counter capable of counting up and down, depending on the signal present on pin 10.

Figure 4.1

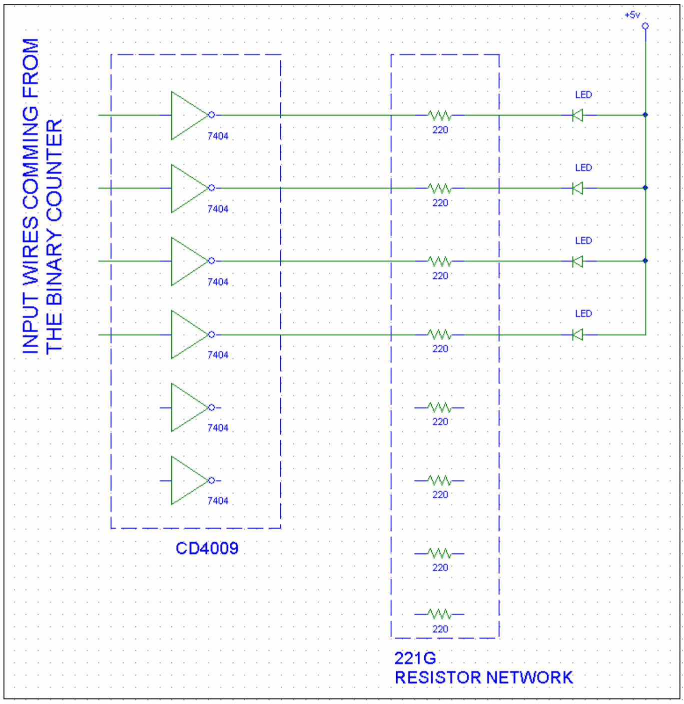


Figure 4.2

5. Wiring the Circuit

In the previous section we explained the components used in the circuit, and why they are used. If you have further questions about the circuit do not hesitate to ask your T.A, he is there to answer your questions.
Wire up the circuit shown in Figure 4.1. Here are the guidelines that you **must follow** when wiring up the circuit. **These guidelines are set in place so you will learn how to wire up neat and organized circuits on a breadboard!** We recommend that you follow these guidelines in all successive lab courses you take at MTU. **If you choose not to follow these guidelines, your T.A. will not help you with debugging your circuit.**

- **The blue line (top and bottom of the breadboard) is a common ground; Do not use it for anything else BUT ground;**
- **The red line (top and bottom of the breadboard) is a common rail (+5V); Do not use it for anything else BUT +5V;**
- **All wires going to ground are one consistent color (green, or black);**
- **All wires going to rail are one consistent color (red or orange);**

- **Note on the use of decoupling capacitors in electrical circuit designs:** A common problem in all electronic systems is the inductance associated with power supply lines. When transistors in integrated circuits switch, the change in current can result in a significant voltage drop across power lines of a chip (dI/dt noise). DC supply voltage must be filtered from fluctuations and this is done by adding decoupling capacitors. The rule of thumb is to place a decoupling capacitor as close as possible to the pin on the chip that is connected to a power rail. Figure 5.1 illustrates an example of where a decoupling capacitor should be placed when wiring up a CD4516 chip. In this **and all other labs** you will do at MTU we expect you to use decoupling capacitors. If you notice, in the diagrams shown in figures 4.1&4.2 we do not show the decupling capacitors. **It is your job to place these capacitors in every circuit you build, regardless of whether they are shown on the diagram or not.**

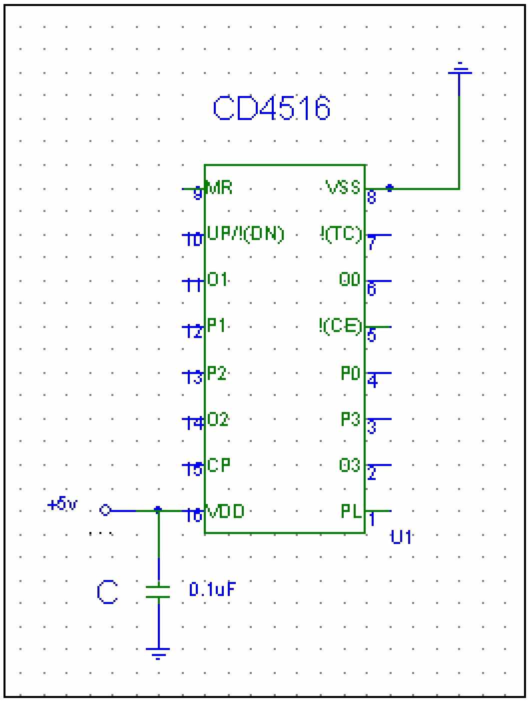


Figure 5.1

- **Note on using CMOS devices in your design:** If you notice in figure 4.1 & 4.2 we do not show the unused pins, on each chip, being connected to anything. **You should not leave them unconnected!** Integrated Circuits, a.k.a chips, built in CMOS technology **must** have all of their inputs connected even if not all of the inputs are used. This is because inputs on a CMOS chip can easily "be driven" by a static electric charge thus making the circuit behave at best unpredictable.

Figure 5.2 shows an example of a semi neatly wired circuit. While you are wiring up your circuit keep in mind that you have to preserve space on the breadboard. You can easily run out of room on the breadboard if you are sloppy with your layout. **You will not be allowed to use more than one breadboard for implementing the controller so pay attention to details and think ahead!**

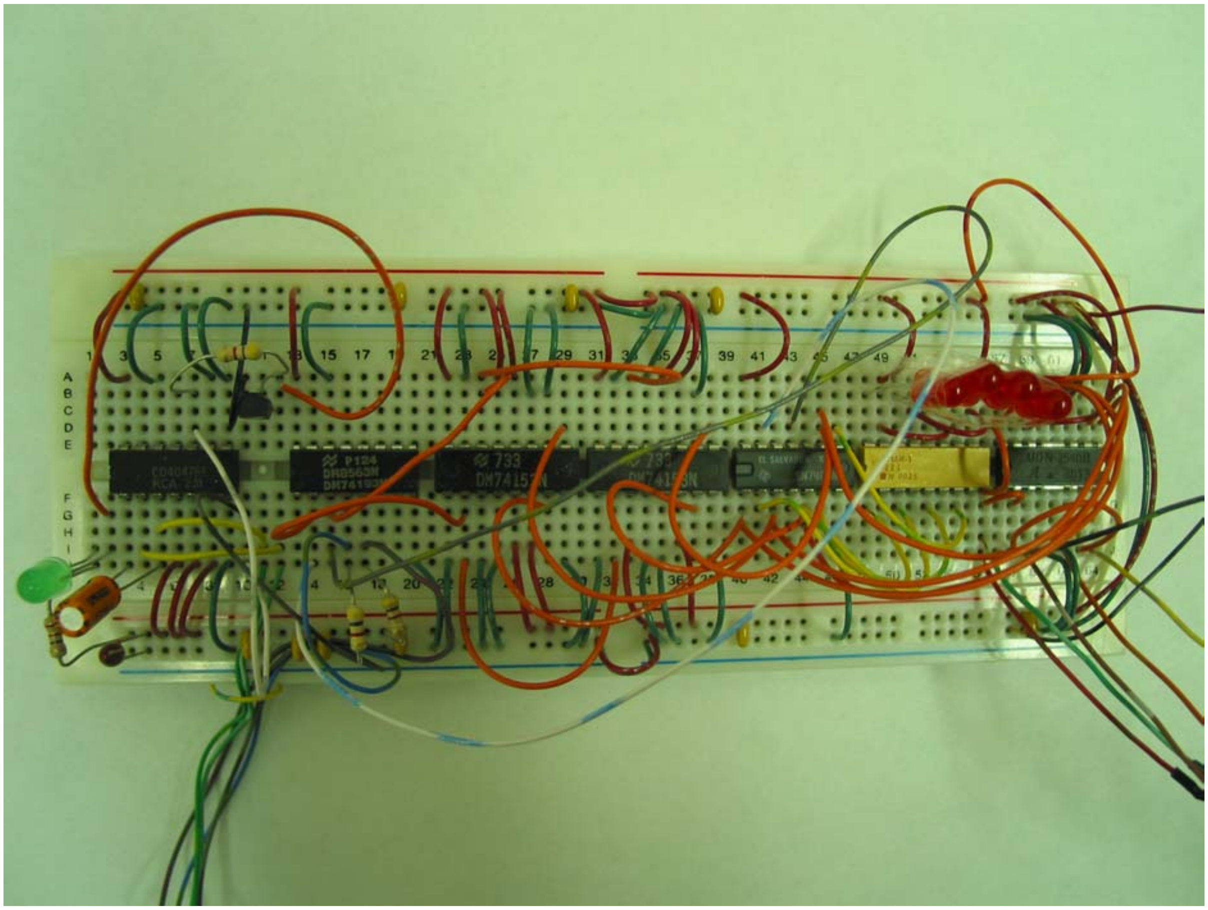


Figure 5.2

6. Testing Your Circuit

This is a two week lab. Do NOT disassemble the circuit that you just built! You will need it for week 2.

Before you power on your circuit have your T.A. inspect the circuit and sign the signoff sheet. When your T.A OKs your circuit, power it on! You should see the LED that is part of the “clock indicator circuit” (see figure 4.1) blink on every tick of the clock. The 4 LEDs should be turning on and off, representing the bit sequence as the counter chip counts.

If your circuit is not functioning start debugging. Here are some helpful tips for debugging your circuit.

- Make sure the power is turned on;
- Using a voltmeter confirm that all chips on the breadboard are getting +5V on the proper pins. Refer to figure 4.1 to see which pins, on each chip, should be wired to +5V;
- Using the oscilloscope confirm that the CD4047 device is generating a clock on pin 10;

  - If you are not seeing a 50% duty cycle square wave (clock) make sure that your Pot is not set to 0Ω, if it is increase the resistance and check for a clock signal;
- Make sure that the interconnections between the chips is correct. Specifically, make sure that pin 10 from CD4047 is connected to pin 15 on the CD4516.

7. References

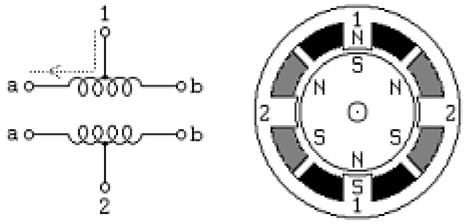
UDN2540 – Darlington Power Driver, data sheet
Airpax Stepper Motor Data Sheet
CD4009 – Hex inverting buffer, data sheet
CD4047 - Square Wave Oscillator (Variable Frequency), data sheet
CD4516 - UP/DOWN Binary Counter, data sheet
2N2222A - NPN General Transistor, data sheet

Appendix A

Stepper Motor 101

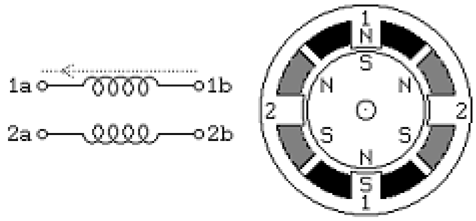
Stepper motors are widely used in motion control applications because they are easy to use compared to DC, DC brushless or servo motors. Stepper motors don't have brushes and contacts, and the shaft position can be accurately controlled by applying a series of pulses to the motor windings. Stepper motors are highly reliable, rugged, inexpensive and produce high torque at low speeds. Stepper motor essentially is a synchronous motor with a rotating magnetic field. They have a set of stator coils and soft iron or a permanent magnet rotor. A rotating magnetic field is created by switching currents in the stator windings. By energizing one set of coil after another, the motor "steps" form one position to another. The step size is given in degrees. As long as there is current in the stator winding(s), stepper motor produces a torque, and will hold its current position. This torque is known as the holding torque.

There are two types of stepper motors depending on how the stator coils are constructed. The coil(s) of a "Unipolar" motor has a center tap, while those of "bipolar" motor do not have a center tap. Figure 1 shows the construction of unipolar and bipolar motors.



Unipolar & Bi-polar Stepper Motors

ABOVE: Unipolar Stepper Motor Stator Coils
BELOW: Bipolar Stepper Motor Stator Coils



For a more detailed introduction/tutorial on stepper motors visit the following site: http://www.cs.uiowa.edu/~jones/step/

Or just try www.google.com with “stepper motor”.

T.A. Sign-off Sheet; Week - 1 -

Group Members: ______________________

____

____

1. The circuit shown in Figure 4.1 is wired properly (or at least it will not blow up when powered up);

____

2. Demonstrate the functionality of the circuit shown in Figure 4.1

____

Pre-lab Questions – Week 1

Student Name: ______________________________

1. What is the difference between a bipolar and unipolar stepper motor?

2. For a given stepper motor model describe the characteristic of stepper motors which determines the number of steps per rotation (i.e. degrees per step).

3. Draw the pinning diagram for the CD4516 chip. Label each pin and write a short description of what each pin is.

Note: You might find it useful to look at the data sheet for this device.