LAB 7
Servo and Stepper Motors

Suggested Reading
What's a Microcontroller Ch. 4, "Controlling Motion"
Thompson/Airpax *Stepper Motor Handbook*
H21A1 photo-interrupter module datasheet
SAA1027 stepper motor driver datasheet

1. Servo Motors

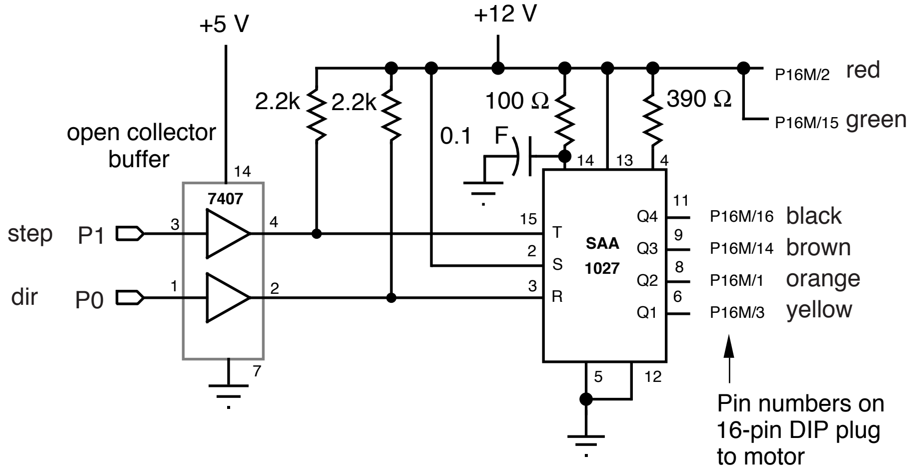
Read and do Chapter 4, "Controlling Motion" from *What's a Microcontroller, ver. 2.1.*

I can't improve on this one.

2. Stepper Motors

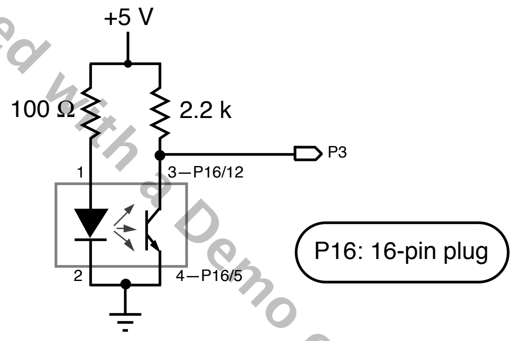
The SAA1027 stepper motor controller allows one to control stepper motors with only two signals. The "T" signal, pin 15 of the SAA1027, causes the motor to execute one step for each pulse. The "R" signal, pin sets the direction of the step. (These labels make sense in German.) The "S" signal enables the functioning of the controller.

Connect the circuit of the next figure on your breadboard, Be sure to keep 12 V away from any TTL chip. It's best to put the 12 V chips on a separate breadboard. Before connecting the circuit to pins P1 and P0, put the step control on one of the Stimulator Board's push button and connect the direction pin to a toggle switch. Verify that the stepper motor makes a step on each press of the push button and that is reverses direction when the toggle switch is changed. After you verify operation with the push button, try getting the step signal from the 1 Hz clock signal.



The opto-coupler allows you to sense when a in the rotating disk is lined up with the led-phototransistor pair and thus get feedback on the stepper motor position.

You should connect the opto-coupler pair is shown in the next figure. Instead of connecting the output to P3, connect it to an led on the Stimulator board and verify that it signals properly when the disk is aligned with the sensor.



After you have explored operation with the pushbutton, toggle switch and led, connect the inputs and outputs to the pins of the microcontroller. Write a program which searches for the slot and then counts the number of steps for a complete revolution of the disk. Have the program do one complete revolution at a slow step rate which is sure to be reliable. Then gradually increase the step rate for subsequent revolutions. At some step rate the motor should start skipping steps because the rate has become too fast for reliable operation. Write a program to search for the slowest step rate at which the motor becomes unreliable.

Change the stepping algorithm to start the disk's rotation at a slow rate and then gradually ramp up the stepping rate. It should be possible to achieve a faster rotation with reliable stepping by ramping up from a slow speed. (See the *Stepper Motor Handbook.*)

Your own ideas:

____

____

____