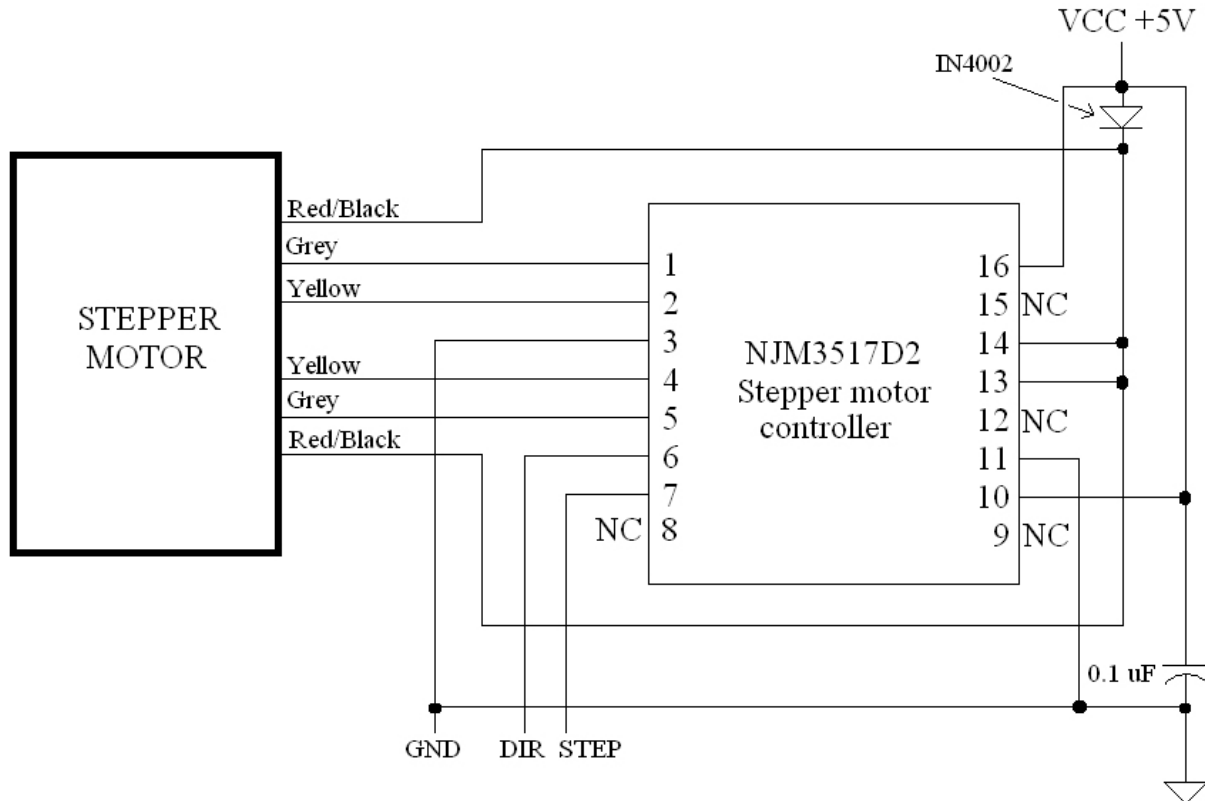


## Stepper Motor Lab

Today, you will build a circuit that drives a stepper motor and write some code that controls it. The circuit diagram for the stepper controller is below.



This is the simplified version of the diagram in the NJM3517D2 data sheet (see handout), so don't be confused by the differences. Take care in keeping track of the color of wires coming out of the motor – there are 2 groups of 3 wires (grey, yellow, and black/red tied together). Also, take care in connecting them in the right way to the NJM3517D2 pins. Also, connect pin 3 directly to the ground wire on the parallel port connector (signal ground), then, connect the system ground (power supply ground) to the signal ground right next to pin 3. Make sure that the diode is oriented correctly – it needs to isolate Vcc from the center-tap (red/black wires) and pins 13 and 14 on the NJM3517D2. **Test your circuit with a function generator before connecting it to the parallel port.**

The parallel port can be used for inputs and outputs and can be configured to operate in different ways (see handouts). **Here, we will just be outputting data to bits 0 and 1 of the PC's port 0378h (LPT1), so no extra configuration is necessary.** Bit 0 controls the direction (DIR) of the motor's rotation and bit 1 is the step (STEP) signal. One step is generated on the negative edge of a STEP pulse, so you will have to raise STEP to a "1" and then drop it again to a "0" to accomplish 1 step. Note that we have tied the NJM3517D2 so that it operates in full step mode, meaning that one pulse yields one full step. Make sure to insert delays between state changes in the STEP pin to allow the signal to be properly received.

Your code should do the following. It should present the user with a menu screen that allows the user to control direction and rotations.

---

Dr. T's stepper motor control system

- 0 – Test the motor
- 1 – Turn the motor
- 2 – Stop the motor
- 3 – Change direction
- 4 – Quit

Status: At rest

---

If the user selects “0”, the motor should make 1 complete revolution in either direction (and only 1 complete revolution) and stop. The system need not announce that it is testing.

If the user selects “1” ...the screen then shows...

---

Dr. T's stepper motor control system

- 0 – Test the motor
- 1 – Turn the motor
- 2 – Stop the motor
- 3 – Change direction
- 4 – Quit

Status: TURNING *ORIENTATION*

---

where *ORIENTATION* is CLOCKWISE or COUNTERCLOCKWISE. At anytime, the user should be able to hit “2” to stop the motor and return to the start-up screen or “3” to change the direction of rotation (make sure to update *ORIENTATION* on the fly). If the user hits “4” at anytime, the system should stop and exit cleanly.

NOTE: The speed of rotation should be fixed for this lab, although you need to make sure you don't spin it too fast or the motor will lock up. Play around with the rotation speed and pick one that you like.