# Stepstick Tester

Need to test:

* Step, Dir, En pins work:
  + Motor does nothing when not enabled
  + Motor moves depending on step and dir
* Microstepping pins work
  + Need to change MS pins and observe actual motor moved distance. Requires motor feedback.

Test Procedure

1. Board is provided with power. Microcontroller with test firmware boots and displays “Ready” on display.
2. Stepper driver is inserted into test socket.
   1. If inserted incorrectly, it should not damage the testing jig.
3. User presses button to begin test.

For each of the following test steps, the test continues until either a step fails or until the test is completed. Hence, a failed step will cause the test to stop prematurely with a message explaining the test failure.

1. Microcontroller enables 5V power line to stepper driver. Waits short period, and checks that there is no short on the 5V power line.
2. Microcontroller enables 12V power line to stepper driver. Waits short period, and checks that there is no short on the 12V power line.
   1. The microstepping pins are set to a specific microstepping setting (microstepping = 1x on first iteration).
   2. The motor shaft’s current position is measured using the attached encoder, and the position recorded.
   3. The DIR pin is set to a known state.
   4. The driver is enabled via the EN pin.
   5. The STEP pin is driven to move the motor ¼ rotation.
   6. The motor shaft’s new position is measured, and the difference calculated. If the motor has moved ¼ rotation (+/- some allowed tolerance) the test is considered passed.
   7. The DIR pin is inverted.
   8. The STEP pin is driven to move the motor ¼ rotation.
   9. The driver is disabled via the EN pin.
   10. The motor shaft’s new position is measured. If it is the same as the start position (+/- some allowed tolerance) then the motor direction has changed, and the test is considered passed.
   11. The STEP pin is driven to move the motor ¼ rotation.
   12. The motor shaft’s new position is measured. If it is the same as the previous position (+/- some allowed tolerance) then the motor did not move when disabled, and the test is considered passed.
   13. This iteration of the test is now completed. We return to Step A, incrementing the microstepping setting, or, if all MS settings have been tested, we complete this step.

# V2 Improvements

* Fix DRIVER\_SUPPLY enable MOSFET circuit so that it can successfully switch 12V on/off
* Change LCD pinout to match cheaper Chinese I2C modules, rather than Adafruit I2C/SPI pinout
* Integrate microcontroller and required supporting circuitry into board
  + Current test firmware requires more RAM, so preferably something with more headroom would be good (see microcontroller selection section)
* Increase current rating of fuse in DRIVER\_SUPPLY circuit so that it does not overload on normal driver+motor current draw
  + Likely needs bigger footprint too
  + Currently rated at hold 200mA / trip 400mA. Should increase hold current to at least 500mA.
* OR replace MOSFET + fuse + ADC setup with load switch with overcurrent protection and status monitoring
* Prevent USB power from back-feeding to point where input power is measured
* Allow driving external motor
  + Add some way to switch between integrated stepper motor and secondary / external connector
    - E.g. pin jumpers on stepper coil phases, or some other switching solution
  + Add firmware option to drive external motor, using integrated motor as encoder + software rate multiplier – i.e. allow jogging an external motor back and forth with controllable speed
* Allow measurement of stepper driver/motor phase current, so that driver module’s current setting can be seen and then adjusted by operator
* Change PCB dimensions and mounting holes to match a COTS plastic enclosure
  + PCB should mount as enclosure’s ‘lid’, to match the current printed setup
  + Enclosure selected must be deep enough to house motor + encoder board
* Make driver insertion + removal easier
  + ZIF socket? Or regular headers with lower insertion force?
* LEDs on stepper config pins to watch signals change?