DAQC2plate Motor Controller Manual

Revision 1.0

Introduction

It is beyond the scope of this document to describe stepper motors in detail. There are a number of resources available for this including our own web site: Pi-Plates.com. Instead, this document will focus on how to use this application to test out the features of the of the Motor Controller mode of the DAQC2plate. This application is also handy to test and verify your motor connections before moving on to your custom code.

Overview

The DAQC2plate can independently drive two unipolar stepper motors with coil currents as high as 3 amps each. All of the timing functions are controlled by the DAQC2 leaving your Raspberry Pi free to perform other tasks. The DAQC2 Motor Controller application is primarily a demonstration of this feature. If you wish to stack additional DAQC2plates and drive more than two motors, then you will need to write your own code using our intuitive Python commands. See the command reference for the DAQC2plate at Pi-Plates.com for more details.

About Unipolar Stepper Motors

Unipolar stepper motors are the easiest to drive from the electronics standpoint for two reasons:

1. The drive electronics is a simple set of four open collector or open drain transistors instead of a pair of H bridges.
2. The software required to do the stepping sequence can be as simple as turning the individual transistors on in sequence: 1-2-3-4-1-2-3-4… or 12-23-34-41-12-23-34-41...

Of course, with that simplicity comes the tradeoffs of:

1. Slightly less torque
2. More difficult to microstep

In most cases, a unipolar motor can also be a bipolar motor but the reverse is not true. To ensure you have a unipolar stepper, count the count the wires attached to the motor. If there are five or more you should be good to go. If you have four wires, check out the Pi-Plates MOTORplate board here. Below are two examples of unipolar motors. Note that they each have 6 wires.

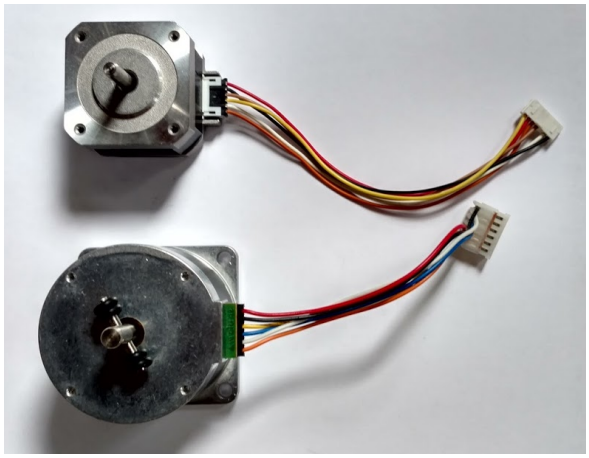

Figure 1

Before You Get Started

The following applies to all of the DAQC2plate applications:

1. You might be tempted to run the Oscilloscope App and the Function Generator App at the same time. However, due to how the Raspberry Pi handles certain hardware resources, this could cause contention issues which would result in both apps misbehaving. Even if you try this with two different DAQC2plates on a single RPi, you could still have problems. So, do not run more than one application at a time and follow this rule: One stack = One app.
2. When launched, an application will run on the first addressed DAQC2plate on the stack. For example, if you have two DAQC2s stacked together at addresses 0 and 5, the application will run on the board at address 0.
3. Do not expect these applications to have the same performance of $1000 instruments from Fluke and Tektronix. They are for the most part, capable of operating in the audio band of 10 to 20Khz.
4. All knobs can be rotated with a mouse pointer for coarse adjustments. For fine adjustments, the left and right arrows on your keyboard can be used.

Specifications

The motor controller application for the DAQC2plate has the following specifications:

Timing Accuracy	+/- 1%
Maximum Coil Voltage	30 Volts
Maximum Coil Current	3 Amps
Motor Count	2
Maximum Step Rate	500 steps/second
Step Sizes Supported	Whole and Half

Motor Controller Operation

The DAQC2plate provides two specific modes for motion:

1. MOVE - in this mode, the motor rotor rotates a specific number of steps of a predetermined size (whole or half) and at a predetermined step rate. This is the most common use of stepper motors
2. JOG - in this mode, the motor rotor will rotate at a predetermined step rate and step size. Motion will continue until a STOP or OFF command is received.

In addition, when not rotating, the stepper motor can be in two different modes:

1. STOP - this mode usually occurs after a stepper motor has just gone through a MOVE sequence. While the rotor is stationary, the coils are still energized and applying “holding torque”
2. OFF - this is the default mode of the motor when the application is first launched. In this state, all coils are off and no torque is being applied to the rotor.

Connections

Refer to the figure below for the connection points of your motors. Phases A, B, C, and D of Motor 1 attach to Digital Output Block terminals 2, 3, 4, and 5. While Phases A, B, C, and D of Motor 2 attach to terminals 6, 7, 8, and 9. Connect the power source for your motors to terminal 1 to avoid destroying the drive transistors with flyback voltages. Use the 5VDC on pin 10 only if you have a good power supply connected to your Raspberry Pi (> 2 amps) and if your motor coils are greater than 5 ohms.

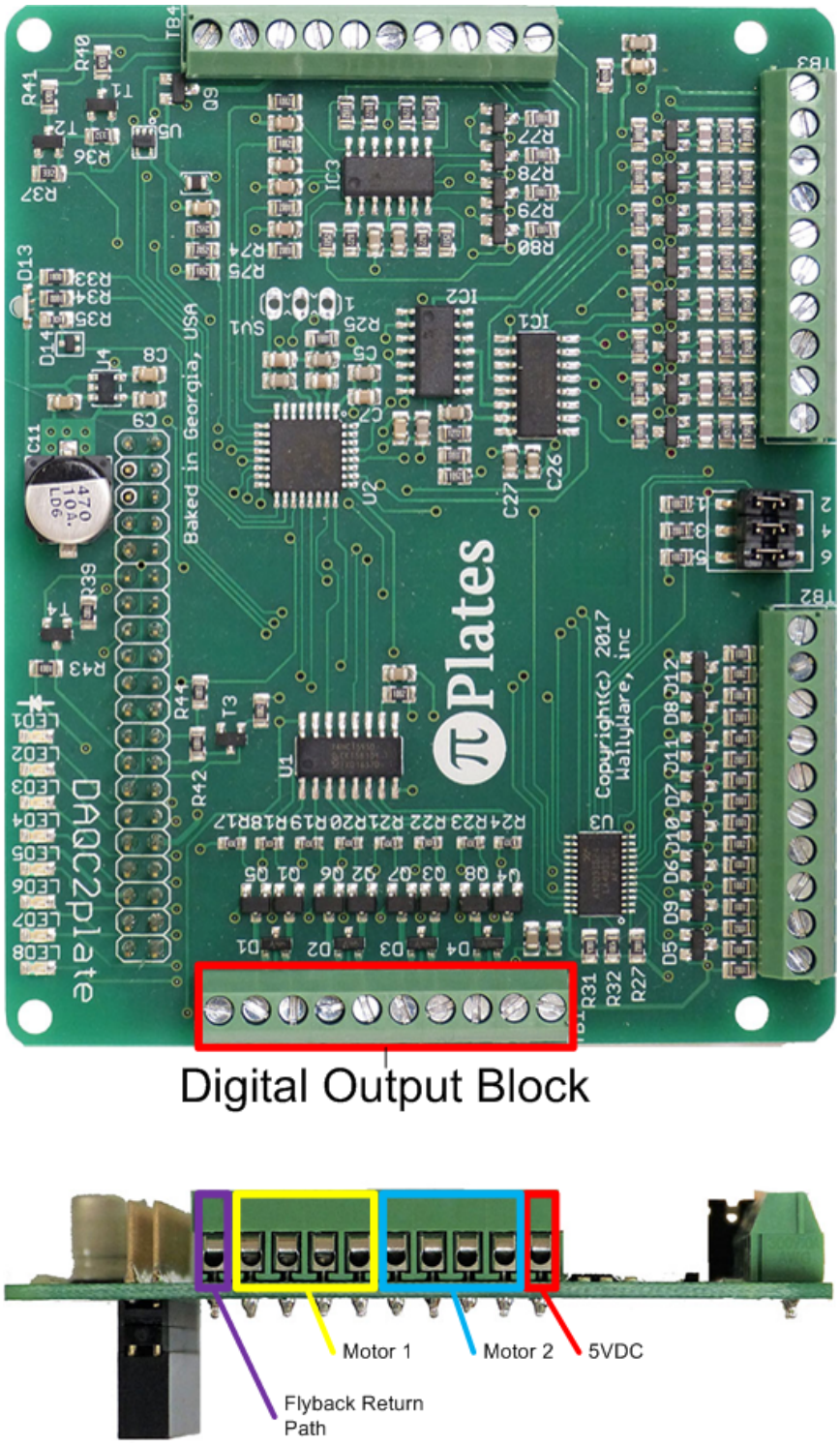

Figure 1

The figure below shows the correct method of connecting two 12V motors to the DAQC2plate:

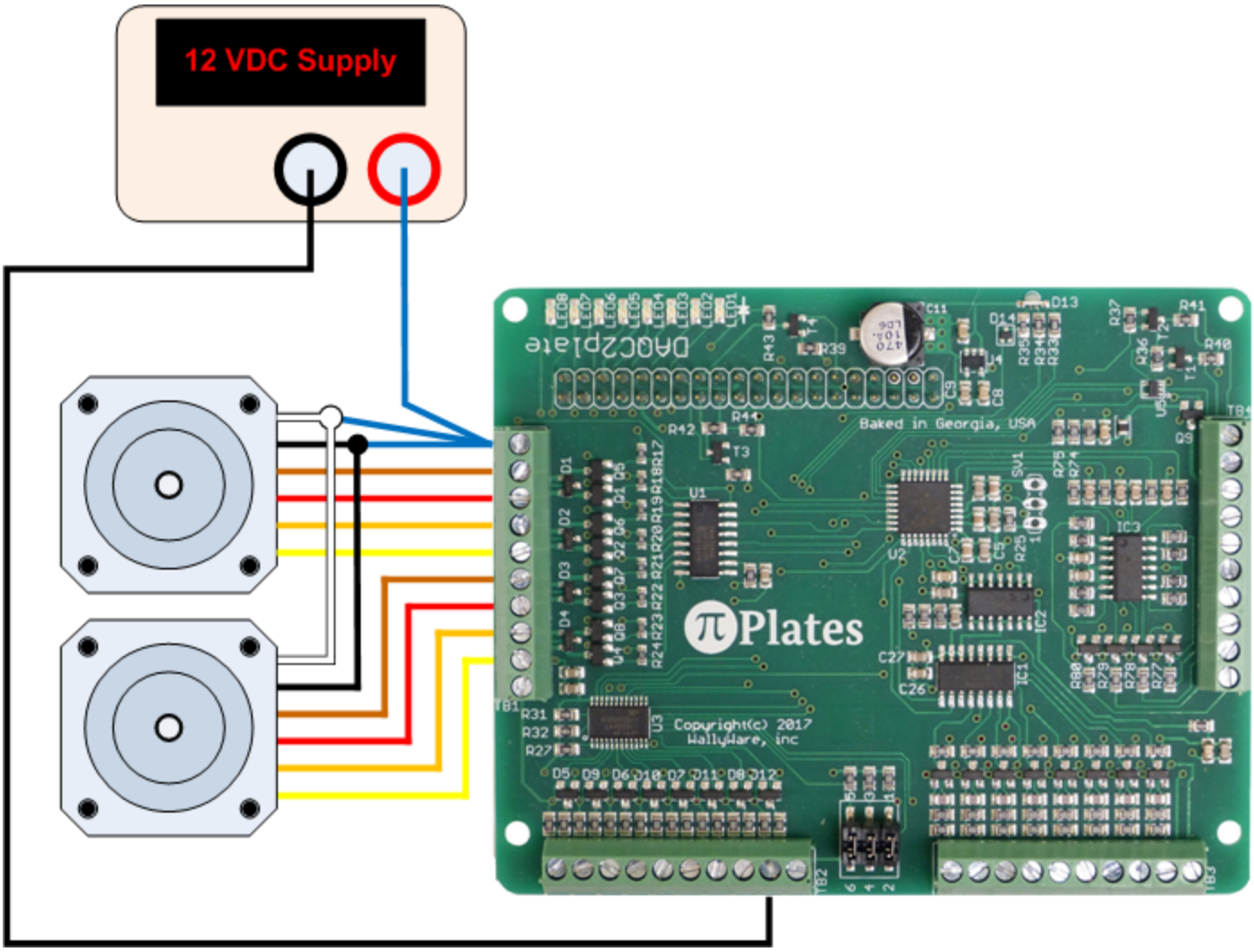

Figure 2

Note how the blue wire from the power supply connects to the common taps on the stepper motor as well as to terminal 1 on the connector. This is a critical connection - if it is not made there's a good possibility that you will damage the drive electronics on the DAQC2plate. Also note that the ground on the power supply connects to terminal 9 on the lower left connector - this provides a return path from the DAQC2plate back to the power supply.

The wire colors shown are what we have on the motors that we use. Some experimentation and/or making measurements with an ohmmeter might be required to determine the pinout of your motors.

If you have motor that operate at 5 volts and a clean 5 volt power supply, the following connections can be applied:

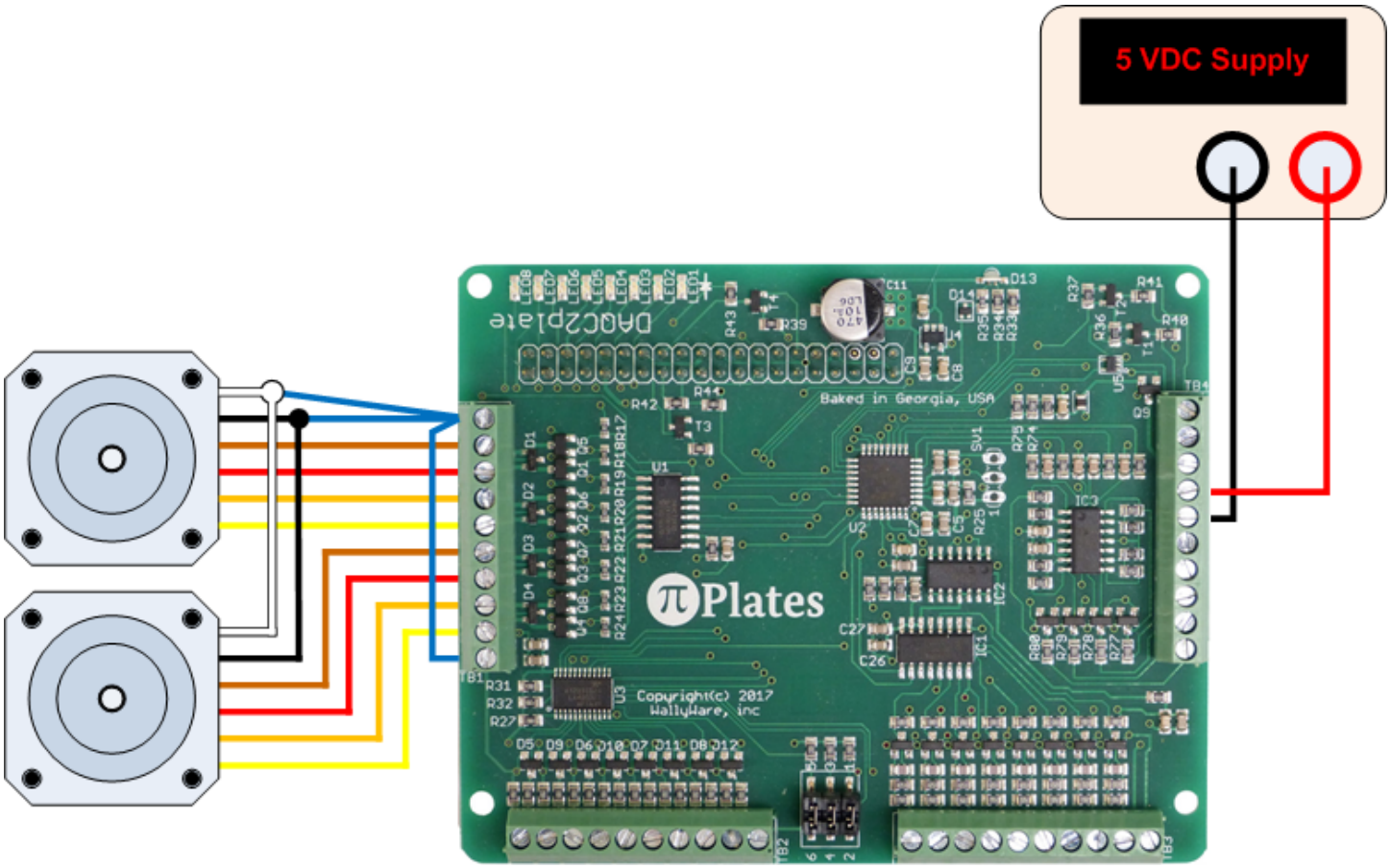

Figure 3

This approach takes advantage of the auxiliary power inputs on the DAQC2plate. It requires that you attach the ground and 5V power leads to terminals 6 and 7 respectively on the Extended Functions Block. It also requires that you run a wire between terminal 1 and 10 on the Digital Output block. Using this setup will also power you Raspberry Pi and any other Pi-Plates on your stack.

Application Layout

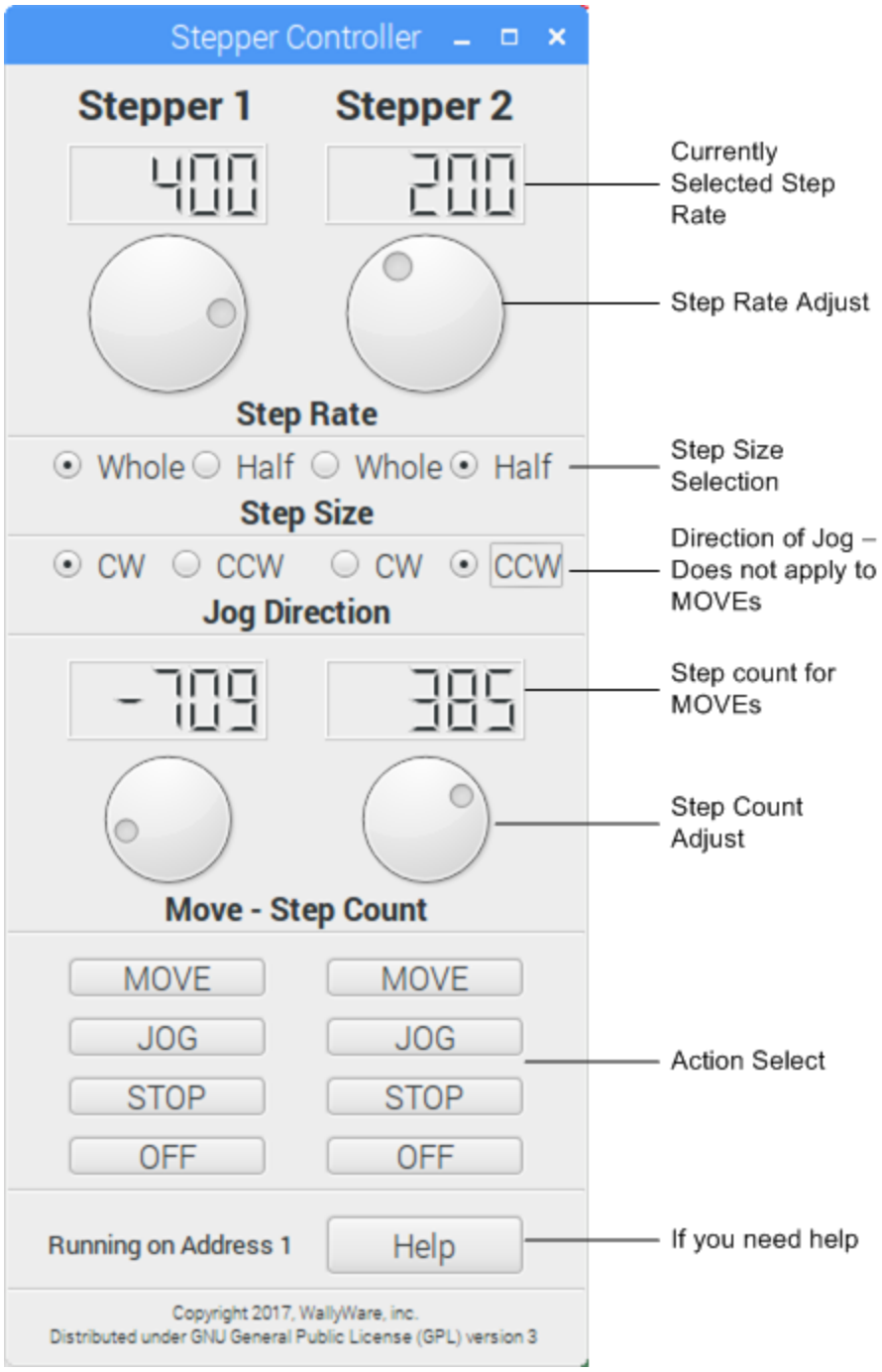

Figure 4

Step Rate Adjust

This adjustment sets the rate at which the motor will be told to step. In this application, the range is from 1 to 500. As an example, if you are using whole steps and a stepper motor that travels 1.8 degrees per step, a step rate of 200 steps/second will result in a full rotor rotation every second.

Be aware that there is a tradeoff between torque and step rate. As the step rate increases, the torque will decrease. In addition, if you set your step too high, you will likely either miss steps or stall out. For more information, refer to the data sheet for your motor.

Step Size

The motor controller on the DAQC2plate supports full and half steps. In the case of a stepper motor that has a resolution of 1.8 degrees, a half step would be 0.9 degrees and a full rotation of the rotor will require 400 steps.

Jog Direction

These buttons only apply to the Jog mode. To save space we used the common acronyms which translate to:

- CW: Clockwise
- CCW: Counter Clockwise or Anti Clockwise if you are British

This setting is only applicable if you have correctly attached your stepper motor to the DAQC2plate.

Move - Step Count

This control sets the number of steps (or have steps) the rotor will rotate when the MOVE button is pressed. This control has no effect when the JOG button is pressed. Note that this adjustment has a range from -1000 to 1000 with positive values resulting in clockwise rotation and negative values cusing counter clockwise motion.

Action Buttons

MOVE

Pressing this button will turn the rotor on the selected motor:

- By the number of steps selected by the Step Count adjustment
- At the rate selected by the Step Rate adjustment
- With a step size selected by the Step Size buttons

JOG

Pressing this button will start the rotor spinning on the selected motor:

- At the rate selected by the Step Rate adjustment
- With a step size selected by the Step Size buttons
- In the direction selected by the Direction buttons

STOP

If rotating in either a JOG or MOVE state, pressing this button causes the motor to instantly stop. Note that the coils are still energized after a STOP is received.

OFF

When stopped, this button de-energizes the coils.

Revision History

Revision	Description
1.0	Initial Release