

# Dual Axis Stepper Motor Controller

by David Rowe

This page describes a dual axis stepper motor controller design and printed circuit board for an equatorial mount. Much thanks goes to Bill Prewitt for his help in the design of this web page.

You may use the information herein without charge. Commercial use is prohibited. Copyright David Rowe 1998.

[E-mail me about this project.](#)

■ Download the AutoCad drawing file of the PC board. [drivebd.zip \[16K\]](#)

■ Download encapsulated postscript file of the artwork. This file contains the artwork for both sides of the printed circuit board [artwork.zip \[31K\]](#). Upon decompression you will find both mirrored and unmirrored versions of the top and bottom artwork.

The following GIF files can be viewed with your browser. To print them use landscape mode.

■ [Schematic Diagram, sheet 1](#)

■ [Schematic Diagram, sheet 2](#)

■ [Schematic Diagram, sheet 3](#)

■ [Hand Controller Schematic](#)

■ [Assembly Drawing, I/O Definitions and Board Size](#)

■ [Parts List](#)

## Introduction

The circuit described on this page was designed to drive the stepper motors on an equatorial mount for the purpose of astrophotography. I have designed in several features which are needed for photography but which are unnecessary for the person wishing a motorized drive for a visual-only telescope where a single-axis, constant-rate drive would prove adequate. The main complications in the circuit come from having two driven motors and from the necessity of having slow motion control on both axes. However, even for visual work, I have found that this system, especially the fast slewing function, makes the telescope more enjoyable to use.

This project is not easy if you don't have some electronics experience. For that reason, I have included detailed schematics of the circuit. This may help you understand what's supposed to happen in the circuit in case you get into trouble. Access to an oscilloscope may be necessary for troubleshooting.

There are several ways to fabricate this circuit. I have included artwork for a homemade double-sided PC board and some instructions that you may find useful. On the other hand, most of the circuitry could be interconnected using wire wrapping techniques. I have not tried this method. If you do, be careful about the thermal dissipation of the SAA1042 stepper motor IC's.

## Description of Operation

The circuit described herein uses the Motorola SAA1042 stepper motor controller IC to provide the high current

signals to the stepper motors. There is one SAA1042 for the polar axis motor and one for the declination axis motor. The clocks for the SAA1042 are derived from a 4 MHz crystal oscillator which is divided down by a programmable divider comprised of four 74HC160 counter IC's and a 74HC02 Nor gate. The programmable divider allows the user to set the stepping rate for sidereal motion over a wide range. A further set of dividers is used to create a number of clock frequencies that are switched into the SAA1042's to allow motion control for both axes. On the schematics, these clock frequencies are labeled 0.5x, 2x, etc. and denote the frequency with respect to sidereal rate (1x). Two multiplexers (74HC151) are used to select the clock frequency for motion control. Finally, four NPN transistors and twelve resistors are used to level shift the clock and direction inputs to the SAA1042 from 5V CMOS logic to the logic levels needed by the SAA1042. A 5V regulator (78L05) is used to make the power supply voltage for the CMOS devices.

There are four input/output ports for this circuit: 1) Power and Ground; 2) Handpad Interface; 3) Programmable Divider Control; and 4) Stepper Motor Interface. Each of these is discussed in the following.

**Power and ground.** The input power to the circuit directly drives the stepper motors, so, if you plan on using batteries to power the circuit then the rated voltage of the motors should be a convenient voltage which can be derived from the batteries. In most instances, 12 volts should be used. A linear voltage regulator (78L05) is used to provide the CMOS circuits with 5V. This regulator will drop-out at an input voltage of 7V or lower. For this reason, *the input voltage should always be maintained above 7V.*

The stepper motor control IC comes in two flavors: The SAA1042 is intended for 6V to 12V steppers and the SAA1042A is intended for use with 15V to 24V steppers. So, if you want to use a higher-voltage stepper, order the SAA1042A.

When you wire up the power and ground supply leads, be careful about the polarity. *This circuit does not have reverse polarity protection.*

In some applications, a lot of current may be used to drive the steppers. If so, be careful to use a wire gauge that will support the current without a substantial voltage drop. This is important for both the input voltage lines and the wires to the stepper motors.

**Handpad Interface.** The handpad interface allows motion control of the telescope via a hand held controller. I have not yet interfaced this circuit to an autoguider so you are on your own here. However, I include a description of the interface operation so that the adventurous can try it.

Seven lines interface the circuit to the handpad. They are labeled N, S, E, W, SEL1, SEL2 and COM. COM is simply 5V from the regulator. N, S, E, and W are active-high 5V-CMOS inputs that control the motion of the telescope. If all four of these lines are open or low, the telescope executes normal sidereal motion. If one of these lines is pulled high or connected to COM then the telescope slews in that direction at a speed that depends on the state of SEL1 and SEL2. Only one of the motion control inputs should be high at any given time.

During motion control, the slew rate of the telescope depends on the state of SEL1 and SEL2. The following table defines the slewing rate for the four possible states of SEL1 and SEL2:

SEL1 SEL2 Operation

Low	Low	2x sidereal rate
High	Low	0.5x sidereal rate
Low	High	12x sidereal rate
High	High	Undefined

**Programmable Divider Control.** A seventeen-line interface is provided for programming the divider that sets the clock rate for sidereal operation. The COM line is connected to the 5V power supply and the other sixteen lines are BCD inputs which control the divide ratio. See the sections on Setting the Divide Ratio and Assembling and Using the Thumbwheel Switch for more information on the operation of this port.

## Making Your Own Double Sided PC Board

If you want to try making your own PC board, be forewarned that this is not necessarily an easy process. It took me three attempts to get it right the first time that I tried it. Several interesting problems can be encountered including the alignment of the top and bottom artwork, getting the exposure correct and drilling the holes in the PC board accurately. In addition, I use the device leads to make connections from one side of the board to the other, so if you make a PC board ***you will have to solder the device leads on both sides of the board*** to assure connectivity from the top circuit to the bottom circuit.

**Materials Needed.** You will need a double-sided pre-sensitized positive-resist printed circuit board (I recommend a 6"x6" piece), a bottle of positive developer, a bottle of PC board etching solution (Ferric Chloride), a 6"x6" piece of window glass and perhaps an Etch Resist Pen. You will also need a sunny day to expose the resist (these can be hard to come by, so an ultraviolet lamp may be necessary.)

**Making the Artwork.** This is a photolithographic process, so you will need a positive transparency (artwork) for each side of the board. I make my own artwork directly from the artwork file on an HP laserjet printer using viewgraph transparency material. You will have to find a way of turning the artwork into transparencies so that you can expose the PC board photo-sensitive resist through it. In doing so, make sure that the emulsion side is down for both the top and bottom circuit. Take a minute to note the orientation and alignment of the top and bottom circuits so that the artwork you produce is correct (and not mirrored).

There are two possible problems at this point. First, the process that you use to produce the artwork may have non-unity scale factors for the x and y directions on the film. This will cause the two films, when placed back-to-back (emulsion sides facing each other) to no longer line up. In addition, if the process shrinks or expands the artwork then the DIP lead spacing will not be quite right. Check this carefully making sure that the distance between the alignment targets is within 2% of 3 5/8". If you can't align the two films then the holes that will be drilled in the board will not align with the device pads, causing much grief. Second, check the density of the artwork. If there are regions that are not solid black, fill in these regions with a black marker. This will make for a much better print on the PC board resist. Otherwise you may be plagued with pinholes (or worse) in the etched copper.

**Aligning the Artwork.** There are four alignment targets in the four corners of the board. These targets are used to align the top and bottom circuit artwork. I start by taping the top circuit artwork to the PC board. I then carefully drill four holes at the center of the alignment targets using a very small drill bit (0.025"). I then remove the artwork, brush off the PC board, and re-tape the artwork to the board, aligning the targets to the holes using an eyepiece as a magnifying glass. I then turn the board over and do the same thing to the other side using the artwork for the bottom circuit. At this time it is important to ***carefully check the relative orientation of the top and bottom artwork***. The above steps must be done in subdued light so as not to expose the resist (which is primarily sensitive to UV).

**Exposing, Developing and Etching the board.** I recommend that you test a small section of the PC board before you try to make the final article. Take the process all the way through on the small test piece, including the etching step. I have found that the exposure times given by the manufacturer are not optimum in all cases.

To expose the board, lay the board on a suitable piece of wood covered with cloth. Lay the glass over the artwork making sure that the artwork is in intimate contact with the board. This may require some extra weight on the glass. Expose one side in bright sunlight, turn the board over and expose the other side. Remove the artwork from the board and develop the resist in the positive developer. Make sure that the resist is completely developed on both sides before removing from the solution.

**Etch the board in the Ferric Chloride solution.** I start by etching the board briefly (about a minute) and then I remove the board and inspect the quality of the resist. At this point you should be able to tell if there is going to be problems with the resist. If you find a problem, use the Etch Resist Pen to carefully mask off the areas that are giving you trouble. Finish the etching.

**Drill out the Board.** Before removing the resist, drill all of the holes in the board with a 0.025" diameter drill. The resist will help to align the drill with the pads. Remove the resist either by exposing it to sunlight and developing, or by dissolving it with acetone. Switch to a 0.043" diameter drill and drill out the large pads for the I/O wires and for the two Zener diodes. Hopefully, your alignment was good enough so that the drilled holes intersect the pads on both sides of the board.

**Tin the Board.** This step is not absolutely necessary but it does make soldering a lot easier. You can tin the board in one of two ways. Either purchase tinning solution and dip the board or apply liquid flux to the board, heat the traces with a soldering iron and apply solder. If you use the second method, remove excess solder with solder wick.

**Add the Jumper Wires and Vias.** There are four jumper wires shown on the assembly diagram. Add these now and remember to solder the wires on both sides of the board (at least where a trace comes off the pad). In addition, there are two vias which connect traces from the top side to the bottom side. These two vias are marked on the assembly diagram and are located under U5 and U6. Stick a wire through the hole and solder both sides. Note that a DIP goes over each via, so be careful to minimize the height of this connection.

## Notes on Populating the Board

If you have not already done so, add the jumper wires to the board as marked on the assembly drawing. If you are using a homemade PC board remember that you must solder each lead to both sides of the board. When inserting the parts, be careful about the orientation. The tab on the transistors (Q1-Q4) denotes the emitter connection and the base of the transistor is the middle lead. The resistor network in the single in-line package (SIP) has an orientation bar next to the common connection. Insert the SIP so that this bar is in the same orientation as the circle on the assembly diagram. Note that the orientation of the 78L05 is determined by the flat side of package.

## Making and Connecting the Handpad

The handpad is for motion control of the telescope. It has four motion control buttons which, when depressed, will cause the telescope to move in either a North, South, East or West direction. In addition, a three-position rocker switch selects the slewing rate for this function. The three slewing rates are 0.5x, 2x and 12x sidereal rate. The middle switch position selects the 2x sidereal rate. More detail on the handpad is provided in the accompanying diagram. The DB9 connector is the method that I prefer in connecting the handpad to the drive electronics because I can then disconnect it for transportation. Use what seems best for you. Whatever connection method you use, make sure that the signal lines on the handpad match those given on the assembly drawing.

The common line (COM) is +5V and goes to each switch. When a switch is open, a resistor on the circuit board pulls down this signal line to ground and when it is closed the COM line pulls the signal line up to 5V.

I made my handpad from solid oak and shaped it to fit into my hand. I drilled a large hole in the oak to accommodate the switches and wiring, used Aluminum sheet stock to mount the switches on the front of the handpad and similar material to close the back. I put a shallow route into the oak so that the Aluminum sheet was inset into the wood for the best look and feel.

## Assembling and Wiring the Thumbwheel Switch

If you decide to use the thumbwheel switch, it will come as four separate gangs and two end pieces. You will have to assemble the gangs and the end pieces together using two screws. Before you do this, note the orientation of the switches and locate the fine lettering on each gang that tells you the function of each of the five connections on the switch. The connections on the switch should be labeled C, 1, 2, 4, and 8.

To wire the switch, I use color-coded ribbon wire. I first wire all four of the C connections (on the four gangs) together at the switch. This is the COM line. I then solder the 16 other connections to the ribbon cable, recording

the gang number, the connection number (1, 2, 4 or 8) and the wire color used for that connection. Then, referring to the assembly drawing, I connect the appropriate wire color to the board. The way it works is this: The leftmost gang on the thumbwheel switch is the "thousands" digit and the rightmost gang is the "ones" digit. Connect the corresponding gang and connection number to the appropriately labeled pad on the PC board assembly drawing. For example, the connection "8" on the leftmost gang would be wired to the board I/O labeled "thousands" - 8. Finally, connect the COM line to the pad labeled COM on the assembly drawing.

When you select a certain digit on a gang, the switch connects "C" to all of the lines that form the binary coded decimal (BCD) representation of that digit. For example, if you select the number 5, "C" is shorted to the gang connections labeled 1 and 4. The programmable divider then "reads" this information once every divider cycle.

When using the thumbwheel switch be sure that all of the dipswitches are in the "off" position.

## Selecting and Using a Stepper Motor

The circuit has been designed to use the Motorola SAA1042 or SAA1042A stepper motor control IC. The SAA1042 is made to drive motors using up to 12 volts and the SAA1042A is made for motors using 12 to 24 volts. In either case, the drive current is limited to 0.5 amps per phase, maximum.

Since the on-board power supply requires a minimum input voltage of 7 volts for correct operation, it would be wise to select motors that can withstand at least this much voltage. For high torque applications, I like the Astrosyn "Miniangle Stepper" which is rated at 10 volts and 0.5 amps per phase. This is a rather large motor (2" x 2" x 2.5") but when it's used with 0.2 amps per phase it has high torque at normal stepping rates. In the past I have also used smaller, lower current steppers with success. In general, steppers do not do well at higher rotation rates. I have chosen 12x as a reasonable high-speed slew rate for many applications. If, when inserted into your system, the motor will not reliably slew at this rate, then you can either increase the bias current or reduce the clock rate to the SAA1042 (for 6x operation.) Follow the directions in the section entitled Reducing the High Speed Clock Rate if you choose this option.

**Adjusting the Motor Current.** The SAA1042 and SAA1042A have adjustable motor drive currents. A resistor is used to select the maximum current which will be presented to each motor winding (phase) when that winding is energized. To adjust this current, select a resistance value from the graph entitled Figure 2. Bias Resistor Rb versus Motor Current given in the Motorola data sheet for the SAA1042. Insert this resistor into locations R13 (for the Dec motor) and R14 (for the RA motor). As mentioned above, I use about 0.2 amps per phase for the Astrosyn motor having a rated current of 0.5 amps per phase. You can use less than the rated current and still get reasonable results. On the other hand, using the full rated current may be required if your torque requirements are severe. This will, of course, be a big drain on your batteries.

**Connecting the Motor.** The SAA1042 will drive a unipolar (4-wire) or bipolar (5- or 6-wire) motor. The first thing to do is to figure out which wires correspond to the two windings in the motor. If you don't have an ohmmeter, you can do this by trial and error by just randomly connecting wires to the SAA1042 output (as far as I know, you can't destroy the IC or the motor by this method.) However, the best plan is to measure the resistance between pairs of wires to find out which pairs are connected across the two windings. If you have a 5- or 6-wire motor, then one or two of these wires respectively will be center-tap connections to the coils (and will not be needed.) By thinking things through, you will figure out the connections. Once the motor connectivity has been determined, simply connect one phase (winding) of the motor to the outputs labeled RA1 and RA2, and the other winding to RA3 and RA4. Apply power to the PC board (at least 7 volts) and the motor should turn at a rate determined by the programmable divider input control word. If the motor turns in the wrong direction, reverse the connections to RA1 and RA2. Similarly, for the declination motor, connect one winding to DEC1 and DEC2 and the other winding to DEC3 and DEC4. When a North or South motion control button is pushed, the motor should turn. Again, if the motor turns in the wrong direction, reverse the connections to DEC1 and DEC2.

## Reducing the High Speed Clock Rate

Should you find that the motor has problems slewing at the high speed rate, you have two possible recourses: 1) Increase the drive current to the motor (if possible); 2) Reduce the high speed slew rate by a factor of two (to 6x

sidereal).

To reduce the high speed slew rate by a factor of two, make the following modification to the PC board: Cut the trace (on the component side) which leads from Pin 9 - U14 to Pin 14 - U2. Add a jumper wire from Pin 14 - U2 to Pin 11 - U12. This will make the high slew rate clock 6x sidereal rate.

## Selecting A Gear Reduction Ratio for Your Telescope

If you have not yet designed the mechanics for your scope here are some suggestions. Because of the discrete nature of a stepping motor, it is important to have sufficient gear reduction between the motor and the driven axis so that the discrete steps executed by the motor do not show up in the telescope as noticeable jumps. On the other hand, if there is too much reduction, the stepping motor will not be able to rotate fast enough to make 12x slewing possible. A good compromise when using a motor with 200 steps per revolution is to have a total gear reduction ratio of around 10,000. This would yield an angular step size of 0.32 arcseconds per half-step. I believe that this is good enough for photography in most situations. In practice I think that a gear reduction ratio of between 5,000 and 15,000 should work OK.

To get the same slewing rate on both axes during motion control, the same gear reduction ratio should be used on both axes. On my split-ring mount I used a declination tangent arm where I carefully selected the length of the arm and the thread spacing to achieve a reasonable match between both axes.

## Setting the Divide Ratio

The programmable divider in this design works in the following manner: A binary coded decimal (BCD) input word (T) is loaded into the counters formed by the four HC160's. The counters then count up at a 4 MHz rate to the value 10,000 at which time the counters are re-loaded with the input word. Thus, the output frequency of the programmable divider is  $4 \text{ MHz} / (10,000 - T)$ . This output is further divided by 24 in the HC4040 and the two HC74's to form the clock for the SAA1042 Stepper Motor Controller. So, the frequency into the SAA1042 (which is defined to be clock rate for normal tracking) is  $4 \text{ MHz} / (24 * (10,000 - T))$ . Now assume that the stepper motor being used has M steps per revolution or 2M half-steps per revolution. Since the SAA1042 is wired for half-stepping operation, the rate of rotation of the motor will be  $R = 4e6 / [48 * M * (10000 - T)]$ . Further assume that the motor is coupled to the polar axis through a total gear reduction ratio G. Then the polar axis will turn at a rate of  $S = 4e6 / [48 * M * G * (10000 - T)]$  in revolutions per second. Since we desire to turn the polar axis at some given rate (sidereal, solar, etc.) we can now determine T:

$$(10000 - T) = 4e6 / [48 * M * G * S].$$

As an example, I built a split-ring equatorial where the ring diameter is 24.5" and the shaft that drives the ring is 5/16" in diameter. This yields a reduction ratio of 78.4. The drive shaft is coupled to the motor by an additional reduction of 168 using a worm gear and timing pulleys, thus,  $G = 78.4 * 168 = 13171.2$ . The telescope uses a stepper motor which has 200 steps per revolution, thus,  $M = 200$ . I set up the drive to track at sidereal rate, thus,  $S = 1$  revolution per 86164 seconds or  $S = 1.16058e-5$  RPS. So:

$$M = 200, G = 13171.2, S = 1.16058e-5,$$

Using the above formula,  $(10000 - T) = 2726$  (rounding to the nearest integer), and  $T = 7274$ .

If you are using the thumbwheel switch, this is the number that you would set the thumbwheel switch to. If you are not using the thumbwheel switch then you would set the dipswitches on the PC board to the BCD value of 7274. The following discusses BCD and how to set the dipswitches.

A binary coded decimal (BCD) digit (having a value of 0 to 9) is encoded by four digital signals. These four signals are weighted in a binary fashion (1, 2, 4 and 8) and are added together to form the decimal digit. The following table shows how the numbers from 0 to 9 are represented in binary:

<u>Decimal</u>	<u>B3</u>	<u>B2</u>	<u>B1</u>	<u>B0</u>
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

B0 is called the least significant digit (LSB) and B3 is called the most significant digit (MSB). The LSB has the value of 1 and the MSB has the value of 8.

The divider control word (T) is made up of four of these BCD digits and therefore has 16 binary inputs. The first BCD digit would correspond to the "ones" value, the second to the "tens" value, the third to the "hundreds" value and the fourth to the "thousands" value. Because of layout constraints, the dials on the PC board are not in the most obvious configuration. Here's how they work: There are two 8-switch packages labeled SW1 and SW2 on the schematics and on the parts placement diagram. Each switch in a package is labeled with a number from 1 to 8 (on the package.) The following table shows the correspondence between the switch and the control word, T. The order given below corresponds to reading the switches from left to right on the PC board.

<u>Device</u>	<u>Switch Number</u>	<u>Partial Value</u>	<u>Comments</u>
SW1	1	8	MSB of "ones" digit
SW1	2	4	
SW1	3	2	
SW1	4	1	LSB of "ones" digit
SW1	5	80	MSB of "tens" digit
SW1	6	40	
SW1	7	20	
SW1	8	10	LSB of "tens" digit
SW2	1	800	MSB of "hundreds" digit
SW2	2	400	
SW2	3	200	
SW2	4	100	LSB of "hundreds" digit
SW2	5	8000	MSB of "thousands" digit
SW2	6	4000	
SW2	7	2000	
SW2	8	1000	LSB of "thousands" digit

As an example, to form the control word T=7274, the following switch states would be used, from left to right:

0 1 0 0 0 1 1 1 0 0 1 0 0 1 1 1

"Off" is defined to be a "0" and "On" is defined to be "1".

## Modifying the Design for Other Uses

If you do not require a dual-axis motion-control circuit, the design can be simplified for a single-axis non-adjustable drive. The easiest way to do this is to take the clock output from the HC4040 labeled 1.5X on the schematic and feed it into R1. This can be accomplished by: Grounding pin 10 of U3, eliminating U1, U2, U4,

U5, U6, U12 and U13 as well as R4-R12, Q2-Q4, D2, and RN3. This modification will change the divide ratio calculation because the clock frequency going into the SAA1042 will now be higher by a factor of 1.5. The expression for the new input control word (T) is left as an exercise for the reader.

-