Electronic Control for DC Motors Using Discrete Bridge Circuits

Gallery of motor-control circuits
and motor-control basics
by Scott Edwards

MOTORS FASCINATE Stamp users. They seem to be looking for a mythical universal motor-controller that interfaces to Stamps, controls any motor from 0.1 to 100 amps at 1 to 50 volts, is 100% efficient, and costs less than $5.

We keep those on a shelf in the back, between the perpetual-motion machines and the replacement teleporter pattern buffers.

This month, we'll look at several types of practical motor-control circuits suitable for use with the Stamps.

DC Motor ABCs. Before we look at our gallery of circuits, let's agree on a couple of terms:

• In this article, *motor* means a permanent-magnet, direct-current (DC) motor of the sort used in toys, models, cordless tools, and robots. These motors are particularly versatile because both their speed and direction can be readily controlled; speed by the voltage or duty cycle of their power supply, and direction by its polarity. DC motors also work as generators. Since generators slow down when they are heavily loaded, DC motors can be electronically braked.

• *Control* means at least on/off and direction control. Variable speed and braking are also desirable.

Most people already know a bit about DC motors from science-fair demonstrations like figure 1. When the motor is disconnected from the battery, it is off; when it is connected with the red wire to the + terminal and black to – it

turns forward; and when the wires are reversed, the motor turns backwards.

You can also demonstrate the motor/generator braking effect. Disconnect the battery and spin the motor shaft with your fingers, noticing how freely it spins. Now connect (short) the motor power wires together and try it again. The motor is harder to spin. The short-circuit load on its generator output makes it harder to turn.
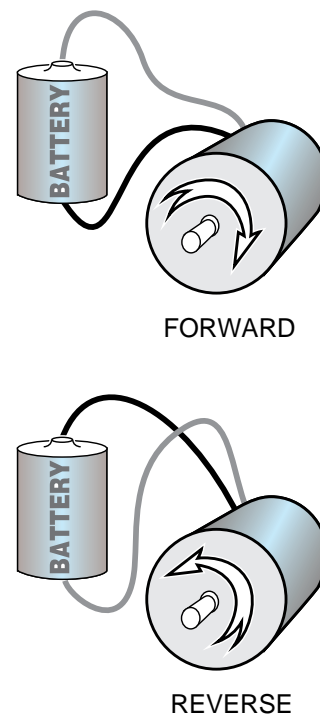

FORWARD

REVERSE

Figure 1. DC motor basics.

Manual Motor Controllers. Figure 2 shows how our understanding of DC motor control translates into five manual motor-control circuits using switches.

Figure 2a is the most closely related to the science-fair demo. A single-pole, single-throw (SPST) switch turns power to the motor on or off, while a double-pole, double-throw (DPDT) switch controls the polarity of the motor connections.

Don't be scared off by this talk about *poles* and *throws*. The component we call a switch can actually contain several switches, all activated by the same handle. These joined switches are indicated on a schematic by a dotted line joining their symbols. Each joined switch is referred to as a pole. So a switch component containing two switches is a *double-pole* unit.

Throws refer to the number of circuits a switch can make. An ordinary on/off switch makes or breaks just one connection, so it's a single-throw switch. The direction switches at the top of figure 2 select one of two connections, so they are *double-throw* switches.

Figure 2b uses a pair of SPDT switches to control direction and on/off. If the two switches are set so that they both connect to the same power-supply rail, the circuit brakes the motor using the motor/generator principle. Figure 2c is very similar, but uses four SPST switches. These switches must be turned on and off in specific combinations to run and stop the motor. Note that a couple of switch settings are not allowed, because they would short out the power supply.

You might consider figures 2d and e to be cheater's solutions to the motor-controller problem. They use a second battery to reverse the motor, thereby simplifying the arrangement of switches. However, extra batteries mean extra weight and expense. And the batteries may wear out at different rates, since in most applications motors spend more time going in one direction or the other. Still, the half-bridge design is worth knowing, because it can be very useful in cheap, efficient, dual-motor designs.

Electronic Switches. The manual motor controllers described above can all be converted to electronic (Stamp) control using one or more of the following types of electronic switches:

Relays. A relay is a mechnical switch operated by an electromagnet. The relatively small current that energizes the electromagnet (the *coil*) can control a larger current through the relay switch(es), known as the *contacts*. However, most relays are not suitable for direct connection to Stamps because even the relatively small coil current is more than the Stamp's pins can supply. This can be overcome through the use of a transistor switch to beef up current handling. See the Transistors section below or *Stamp Applications* no. 6 (available from the *N&V* Web site) for more info.

Relays have two useful properties for small motor controllers: (1) Their contacts have very low on-resistance, meaning that very little power is wasted. (2) They are available in just about any combination of poles and throws you can imagine. As we'll see, solid-state multipole/multithrow switches are usually built up from many SPST units.

On the downside again, relays are slow, make noise, and wear out. They are almost useless in schemes that switch power on and off rapidly to control motor speed (duty-cycle control).

Transistors. By allowing a small base current to control a larger collector current, transistors make good switches; see figure 3. They're fast, quiet, and can last forever. On the other hand, the base current can be somewhat high by Stamp standards (20 to 25 mA). Transistors are usable only as SPST switches, and you have to consider polarity in selecting the transistor type and connecting the load. Finally, even a fully-on transistor has a voltage drop between the collector and emitter. It's typically 0.5 volts, but can be 1 volt or more in Darlington configurations (discussed below). This wastes power and generates heat, which can damage or destroy the transistor.
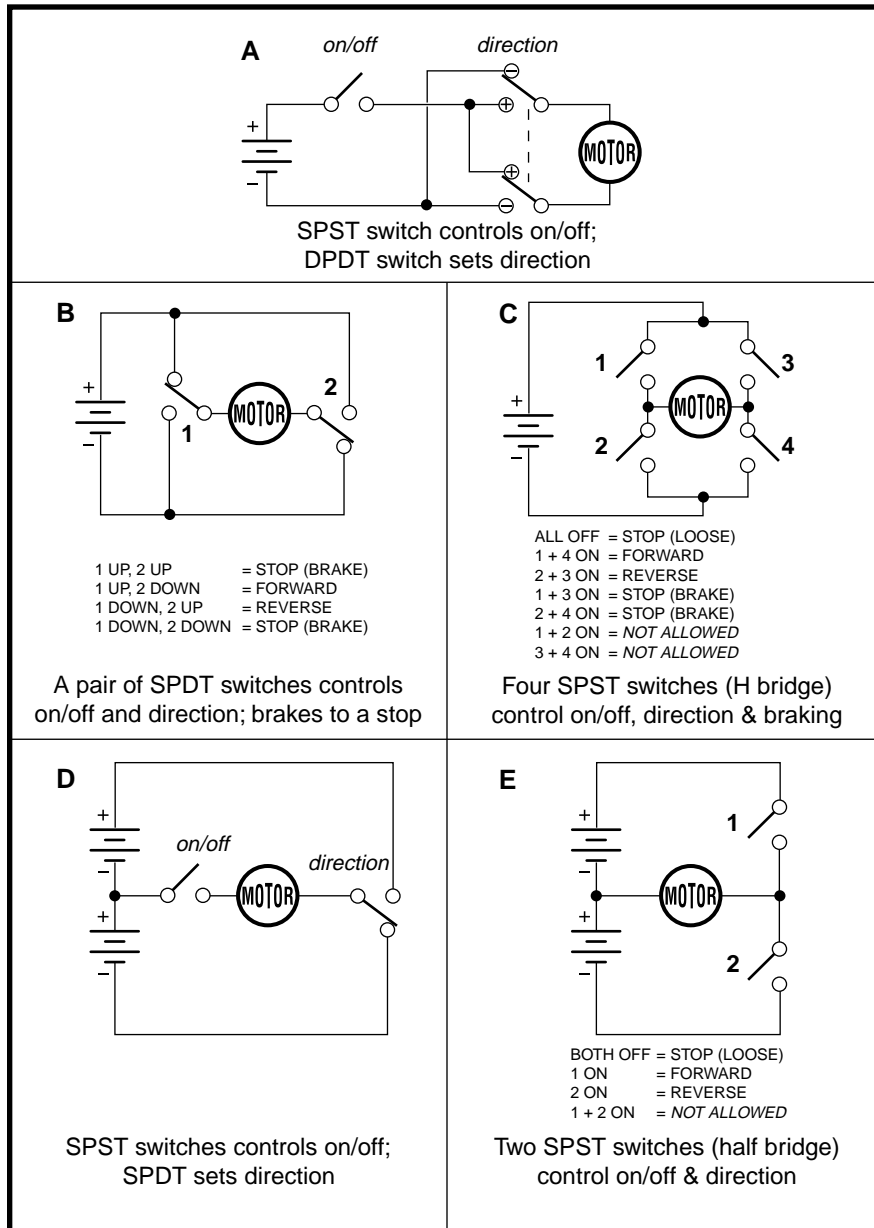
A *on/off* *direction*

SPST switch controls on/off;
DPDT switch sets direction

B

1
2

1 UP, 2 UP   = STOP (BRAKE)
1 UP, 2 DOWN   = FORWARD
1 DOWN, 2 UP   = REVERSE
1 DOWN, 2 DOWN   = STOP (BRAKE)

A pair of SPDT switches controls
on/off and direction; brakes to a stop

C

1   **3**
2   **4**

ALL OFF = STOP (LOOSE)
1 + 4 ON = FORWARD
2 + 3 ON = REVERSE
1 + 3 ON = STOP (BRAKE)
2 + 4 ON = STOP (BRAKE)
1 + 2 ON = *NOT ALLOWED*
3 + 4 ON = *NOT ALLOWED*

Four SPST switches (H bridge)
control on/off, direction & braking

D

on/off

direction

SPST switches controls on/off;
SPDT sets direction

E

1

2

BOTH OFF = STOP (LOOSE)
1 ON   = FORWARD
2 ON   = REVERSE
1 + 2 ON   = *NOT ALLOWED*

Two SPST switches (half bridge)
control on/off & direction

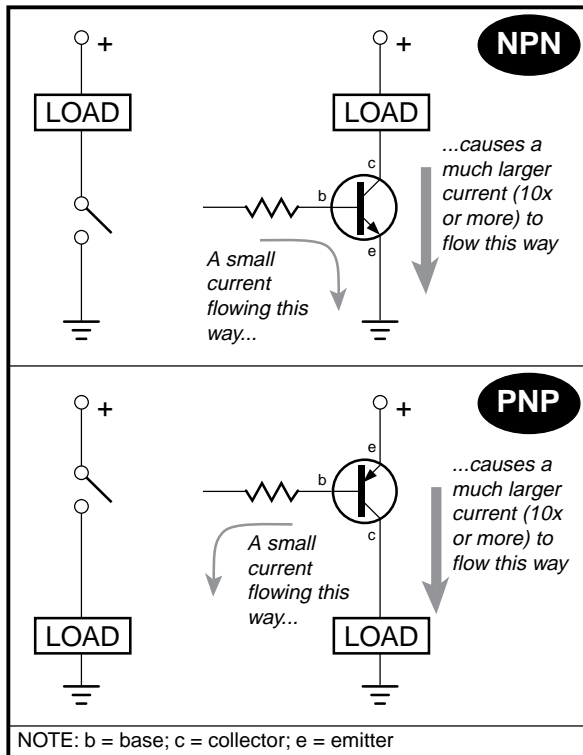*Figure 2. Motor controllers using manual switches.*

Figure 3. Transistors make good switches.

MOSFETs. At first glance, MOSFETs (metal-oxide semiconductor field-effect transistors) would seem to eliminate all the problems of relays and conventional transistors. Their control input, the *gate*, draws almost no current. It switches in response to the presence of a voltage. A turned-on MOSFET can offer an on resistance that many relays would envy. And reasonably priced MOSFETs are available in current ratings that look like a misprint—50 amps or more out of a thumbnail-sized package.

Really, the only trouble with MOSFETs is that they are at their best with supply voltages above 10 volts, and with control voltages higher than the supply. And if they are to be switched on and off rapidly, you can forget about that 'almost no gate current' stuff because the gate acts like a pretty high capacitance. MOSFETs don't hold up well in tinkerers' workshops where antistatic precautions (grounded work surfaces, wrist straps, etc.) are unheard of.

I'll catch some flak for saying this, but MOSFETs' special handling and design requirements make them unsuited to beginners' skills. If you need this kind of high efficiency,

but aren't up to building it yourself, see the next section.

Integrated, Kit, and Packaged Motor Controllers. There are lots of ICs and black boxes that make motor-controller design more of a shopping challenge than a design exercise. Two ICs that are popular with robotics enthusiasts are the L293D and LMD18200. Wirz Electronics sells these parts and provides design information via the Internet; see Sources. Basic specs are:

L293D: motor supply voltages up to 36V, current to 1 A (2A peak). Voltage drop across each leg of H-bridge, 1 to 2V.

LMD18200: motor supply voltages of 12 to 55V, current to 3A. On resistance of each leg of H-bridge 0.33Ω typical, 0.6Ω max.

The robotics-supply store Mondo-Tronics offers motor-control kits that are inexpensive, educational, and easy to build. The program listing is an example of motor speed- and direction-control using their their Mini Dual H-bridge Motor Driver kit, shown in the accompanying photos. Mondo also carries suitable DC motor assemblies (like the Tamiya motor/gearbox in the picture) and robotic components.

A nice feature of Mondo's controller kits is that all have the same connector layout and interface requirements. If you build a small robot (or motorized widget) and later want to build a larger one, just swap motor controllers.

Packaged motor controllers, called electronic speed controls (ESCs), are available from hobby shops for use in remote-control cars, boats, etc. Some ESCs are brutes capable of switching 10s or 100s of amperes. Best of all, accept just a servo-control pulse train input to operate. That's a 1- to 2-ms pulse repeated 60 times a second. As the Stamp and Counterfeit application notes show, it's relatively easy to generate these pulses from PBASIC. If you need to control more than one motor, or your PBASIC program is too tied up to meet the 60-pulse-per-second requirement, you can use my company's Mini SSC (serial servo controller) to generate eight continuous pulse streams. See Sources for catalog information.

Example Circuits. Figures 4, 5 and 6 are representative motor-control designs derived from the ideas presented figure 2a, c and e. These are tested, proven circuits suitable for use with small DC motors. Even if your application requires a much larger motor, it makes sense to get your feet wet with smaller motors so that you can begin to understand some of the issues involved. And a small geared motor assembly, like the Tamiya unit shown in the photo, has enough oomph to propel robots up to a couple of pounds at reasonable speeds. This particular motor was used to test the control circuits presented here, and draws approximately 250mA at 3.0 Vdc. Note that current is dependent on voltage, so the figures in the descriptions below are based on actual measurements.

Figure 4, Dual Relay Controller. This design uses DPDT relays to control motor direction, and an IC-packaged transistor array to turn the relays and motors on and off. Experienced eyes will spot an apparent error—this design gangs two of the ULN2003 Darlington transistor outputs together for motor on/off control. This is normally a bad idea, because slight differences in transistor characteristics will cause one of the transistors to bear all of the load. However, because these transistors are on the same silicon die, their characteristics and temperature are close enough to identical to make this trick safe.

Because the transistors inside the ULN2003 are Darlingtons, they require only a small (<1mA) control current to turn on. The downside is that they have a higher-than normal voltage drop of about 1.5 V. If the circuit is driving our sample motors, that means that the motors get 4.5V x 0.40A = 1.8 watts (W) of power, while 1.5V x 0.40A = 0.6 W is wasted. (This assumes no loss through the relay contacts.) So 75% of the total 2.4 W gets to the motor. To be fair, we need to also factor in the relay coil current, 4.5V x 0.1A = 0.45 W to the waste (non-motor) side of the balance sheet. Now our total is 2.85 W, of which the motor gets 63%.

Figure 5, Darlington H-bridge. This is the circuit used in the Mondo Mini H-bridge kits. It also uses Darlington pairs of transistors to keep control current to approximately 1mA. As with other Darlington arrangements, the penalty is increased voltage drop—I measured it at about 1.2 volt per conducting leg of the bridge for a total of 2.4 V (since two legs of the H have to be conducting for the motor to move).

With the sample motors, the motors get 3.6V x 0.30A = 1.08W of power, while 2.4 x 0.30A = 0.72 W is wasted. So 60% of the total 1.79W gets to the motor.

Figure 6, Half bridge. This circuit offers the pleasant combination of high efficiency and low parts count, at the expense of splitting the battery supply into separate sections

The circuit takes advantage of the Stamps' relatively high current outputs to eliminate the need for an input transistor or Darlington. Where the previous circuits needed just 1 mA of control current, this one requires almost 10 mA. It's a worthwhile trade for the reduced parts count.

Performance of a half bridge varies depending on which transistor is turned on; the PNP transistor dropped 0.5V and the NPN just 0.3V. Taking the worst case, the motor received 2.5V and drew 0.27A for 0.675W of power. The transistor wasted 0.27A x 0.5V = 0.135W. So 83% of the power goes to the motor.

I subsituted high-performance switching transistors in the circuit (Zetex ZTX689B for NPN and ZTX788B for PNP; see Sources). I also changed the upper 4.7k resistor to 1k to prevent the high-gain transistors PNP from switching partly on due to the difference between the motor supply (6V) and the logic-level input (5V).

The improved transistors reduced the worst-case voltage drop to 0.2V. This improved performance as follows: motor, 0.81W; waste, 0.058W; efficiency, 93%.

So splitting the battery supply may be worthwhile when you have to get the most out of a set of batteries. If you tear apart motorized toys, you may find a circuit like this.
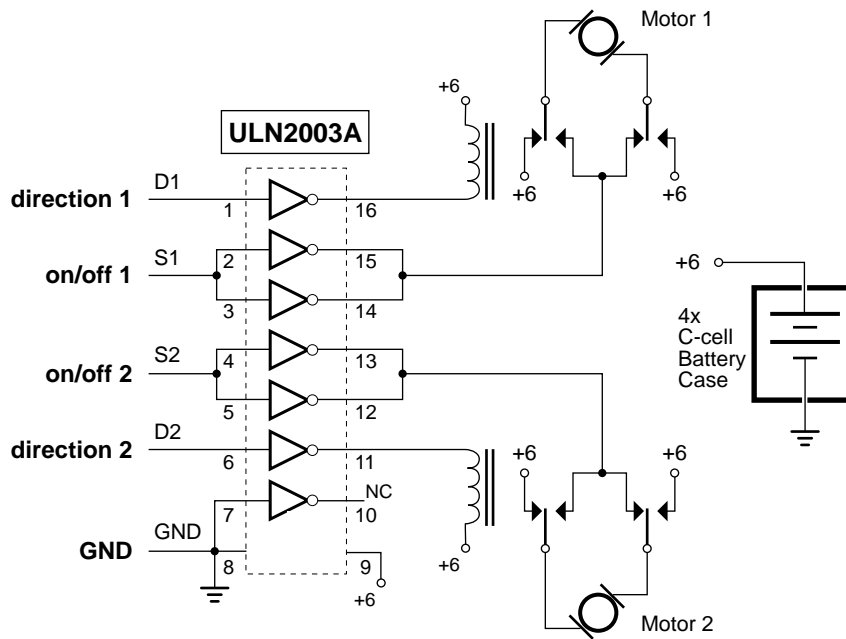
PARTS:
Relays—DPDT, 1A (or more) contacts, 5Vdc coil (JC: 99346)
ULN2003—7-channel Darlington transistor array (JC: 34278)
Motors—DC permanent magnet motors, ≤ 500 mA at 4Vdc
[See Sources for Jameco (JC) contact info]

Figure 4. Controller combines IC driver with DPDT relays.
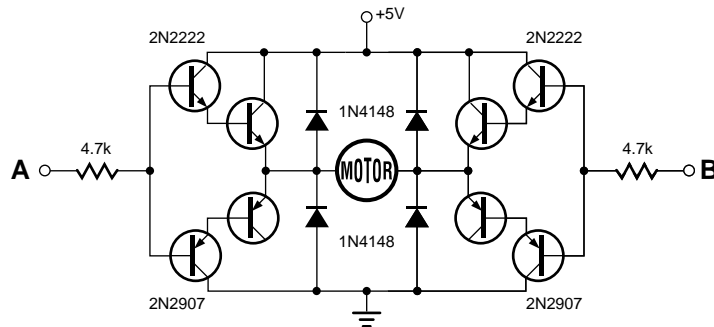
A	B	
0	0	stop
0	1	forward
1	0	reverse
1	1	stop

PARTS:
2N2222—NPN switching transistor (JC: 28628)
2N2907—PNP switching transistor (JC: 28644)
1N4148—Silicon switching diode (JC: 36038)
Motors—DC permanent magnet motors, ≤ 300 mA at 6Vdc
[Available as complete kit from Mondo-Tronics, see Sources]
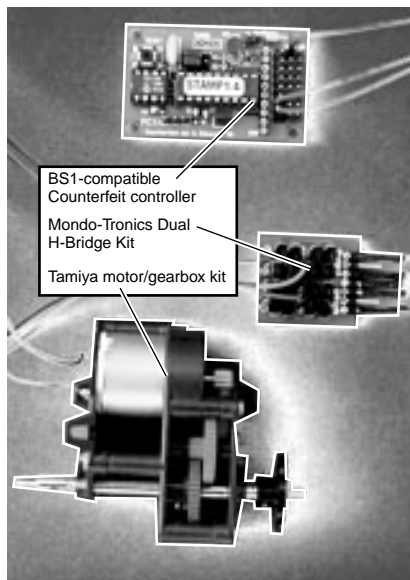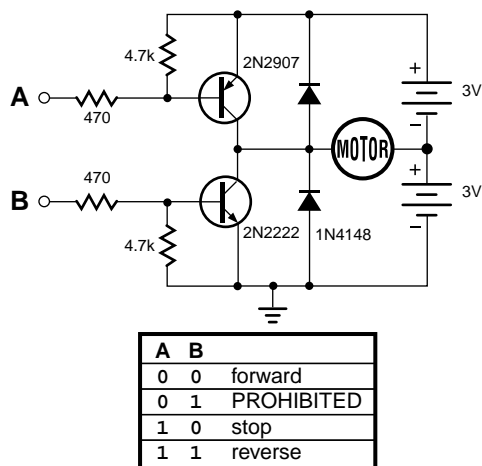
Figure 5. Mondo-Tronics H-bridge.

Photo of motor, controllers.

The half bridge can make of sense for small robots with dual drive motors for propulsion and steering. Wire the motors so that they use opposite battery supplies when the robot is rolling forward, and you'll even out the wear on the batteries.

A	B	
0	0	forward
0	1	PROHIBITED
1	0	stop
1	1	reverse

PARTS:
2N2222—NPN switching transistor (JC: 28628)
2N2907—PNP switching transistor (JC: 28644)
1N4148—Silicon switching diode (JC: 36038)
Motor—DC PM motor, ≤ 300 mA at 3Vdc
[See Sources for Jameco (JC) contact info]

Figure 6. Half bridge is cheap, efficient.

Figure 7, Ingenious H-bridge. I did not design and have not had a chance to build this last circuit, but I was struck by its cleverness. It was designed by Rick Moll, editor of the late lamented *Robotics Practitioner* magazine.

The circuit is an H-bridge with a twist—each input transistor transfers current out of the base of a PNP and into the the base of the opposite NPN. The current switches on both transistors to make the motor run. Control current is approximately 3.6 mA; not as low as the Darlingtons, but better than the half bridge. And there's no voltage-drop penalty; figure a 0.5V drop across each leg of the circuit, for a total 1V drop. Almost the best of both worlds. This H-bridge cannot brake the motor, since that requires turning both NPNs or both PNPs on simultaneously.

Wrapup. I'm out of space, but I've just scratched the surface of this fascinating subject. I'll revisit motor control in a future column. The continuation of last month's BASIC for Beginners will appear next month.

Sources. For more information on the BASIC Stamp, contact Parallax Inc., 3805 Atherton Road no. 102, Rocklin, CA 95765; phone 916-624-8333; http://www.parallaxinc.com.

Mondo-Tronics offers goodies for robotics, including the motor controller and motor-gearbox mentioned in the article (controller is part no. 3-301). Mondo-Tronics Inc., 524 San Anselmo Ave. 107-11, San Anselmo, CA 94960; phone 415-455-9330; http://www.robotstore.com.

Wirz Electronics sells H-bridge driver ICs. See them at http://cec.wustl.edu/~blw2/index.html.

Jameco supplies many of the electronic componentslisted in the circuits. Jameco Electronic Components, 1355 Shoreway Road, Belmont, CA 94002-4100; phone 415-592-8097 or 800-831-4242; fax 415-592-2503 or 800-237-6948.

Zetex transistors are available from Digi-Key, Digi-Key, 701 Brooks Avenue South, PO Box 677, Thief River Falls, MN 56701-0677; phone 1-800-344-4539, fax 218-681-3380, Internet http://www.digikey.com.

For a catalog of Stamp-related products, contact Scott Edwards Electronics, PO Box 160, Sierra Vista, AZ 85636-0160; phone 520-459-4802; fax 520-459-0623; Internet at ftp.nutsvolts.com in /pub/nutsvolts/scott; e-mail 72037.2612@ compuserve.com.
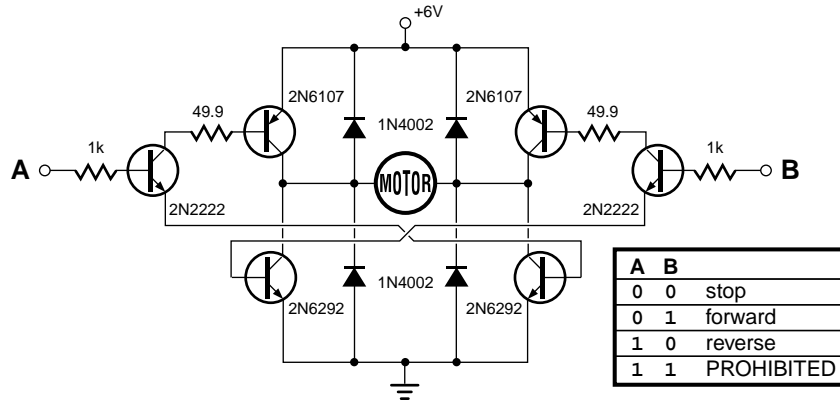
Figure 7. Ingenious H-bridge design by Rick Moll.

Listing. BS1 Program to Demonstrate Mondo-Tronics H-bridge

```
' Program: MONDOMOT.BAS (Demonstrate Mondo-Tronics H-bridge)
' This program demonstrates the Mondo motor controller to
' control the direction and speed of a DC motor. Connect
' input A of the controller to Stamp pin 0; B to pin 1; and
' GND to GND. Run the program. The motor will slowly accelerate
' to top speed, then stop and repeat the acceleration in
' reverse. This program uses a carry-the-1 method of generating
' duty cycle control of motor speed. When you add a number to an
' "accumulator" (a memory location of fixed size), the accumulator
' will overflow if the result is bigger than it can hold. The
' larger the number added, the more likely an overflow or "carry"
' is. This program adds the desired motor duty cycle to an
' accumulator, and turns the motor on only when there's a carry.
' Higher duty cycles make the motor run faster. This method works
' well with the Stamp because it is more or less independent of
' speed.

SYMBOL motAcc = b11     ' Motor-speed "accumulator."
SYMBOL motDir = bit0    ' Motor direction: 0=fwd; 1=reverse.
SYMBOL spd = b10        ' Motor speed, 0 (off) to 15 (full on).
SYMBOL cycles = b9      ' Number of loops at a given speed.
SYMBOL A_ = pin0        ' Controller A input.
SYMBOL B_ = pin1        ' Controller B input.

dirs = %11              ' Set pins 0 and 1 to output.
again:
  for cycles = 0 to 255         ' Turn 255 cycles at each speed.
    gosub motor                 ' Output to motor.
  next
  spd = spd +1                  ' Increase speed.
  if spd <= 15 then again       ' If speed is > 15, then..
  spd = 0                       ' ..turn motor off..
  motDir = motDir ^ 1           ' ..and reverse direction.
goto again                      ' Loop forever.
```

```
motor:
  motAcc = motAcc & %1111       ' Limit motAcc to 4 bits.
  motAcc = motAcc + spd             ' Add speed.
  if motAcc >= 16 then motOn    ' If carry, then turn on motor.
  A_ = B_                       ' Otherwise, motor off.
return


' If you look at the table accompanying the H-bridge, you'll
' see that the motor is on only when inputs A and B are opposite.
' Programming shorthand for this is to set A to the motor
' direction, and make B = NOT A. PBASIC1 does not have a NOT
' function per se (see LET in the manual), so we make do by
' XORing A with 1, which has the same effect: B = A ^ 1.
motOn:
  A_ = motDir: B_ = A_ ^ 1
return
```