The Nuts and Volts of BASIC Stamps
Parallax, Inc. www.parallaxinc.com
Nuts and Volts www.nutsvolts.com

**Column #52, August 1999 by Lon Glazner:**

# Stamp-Controlled High Power H-Bridge

It seems like quite a few people are using BASIC Stamps as the brains in low-level robotics. Because of simplicity, ease of use, and versatility, it's no wonder that the BASIC Stamp is finding wide spread use with hobbyists and amateur robotics groups.

On the Stamp List — an informative E-Mail list provided by Parallax — I routinely see questions about how to bridge the gap between power devices and the BASIC Stamp. Often times, the applications require relatively high power motors for robotics, or other more industrial needs. I thought it might be interesting to share my experiences with an IC that I've used several times in the past.

This circuit can be used for controlling high current motors, driving relays, controlling lamps, and powering heating elements, just to name a few of the possible applications.

**Defining the Design**

The first thing that needs to be defined is what I mean by “high power.” For the purposes of this article, I think we should limit it to about 30V and no more than 30A (so anything in the area of 900W). In the real world of hydroelectric dams and power utility companies, 900W is minuscule. But here, in the land of BASIC Stamps, I feel pretty comfortable using the "high power" moniker for this circuit.

One of the first concerns a designer has to face when confronted with high power requirements is the question of power dissipation. Any voltage drop across a device carrying current is accompanied with power being dissipated in the device as heat. In high current applications, any voltage drop can create significant needs for heatsinks and other means of removing the heat from the current carrying device.

For instance, take a power diode with a forward voltage drop of 0.5V. This diode carrying a load current of 1A would be required to dissipate 0.5W of power (Power = Voltage*Current; P=VI). That is not too big of a deal. But increase the diodes load current to 30A and you've got a problem, a 15W problem.

For this reason, designers will often attempt to reduce the voltage drops that might be found in a high current application. A device that is ideal for this is the MOSFET (are you ready for this — metal-oxide-semiconductor-field-effect-transistor), which can be considered a voltage-controlled switch. Like the BJT (bipolar-junction-transistor, such as a 2N3904), the MOSFET can be biased into its active region (not all the way on, or all the way off). But in power switching applications, that is rarely done. In this design, the MOSFETs will be used either in the on or off state. Many applications require a load to be "turned on/off." This often means closing (on) or opening (off) a current path to your load, and MOSFETs are ideal for this purpose.

Previously, I defined high power as meaning about 900W. In a perfect system, we would lose no power in our MOSFETs as they switch our load into or out of the circuit. Therefore, all of the 900W would be dissipated by our load. But, of course, this isn't the case.

A little bit of information about MOSFETs would probably be helpful at this point. As I mentioned earlier, a MOSFET can be considered a voltage-controlled switch. This differs from a BJT which is a current-controlled switch. The MOSFET, like the BJT, comes in two basic flavors. They are the P channel (analogous to the PNP-BJT) and the N-channel (analogous to the NPN-BJT). A MOSFET typically has three points of connection. They are: the gate, drain, and source (similar to the base, collector, and emitter of a BJT).

All MOSFETs have an "on resistance." This can be considered a resistor in series with your load. In a particular MOSFET data sheet, this value is typically listed as Rdson (resistance from drain to source while the device is on). N channel MOSFETs typically have a lower Rdson resistance as compared to the P channel types. Most power dissipated by a MOSFET is due to the Rdson value. The Rdson value is also dependent on the "gate drive voltage" (voltage on the gate when referenced to the source).

**Figure 52.1: Block diagram of H-bridge and HIP4081A**

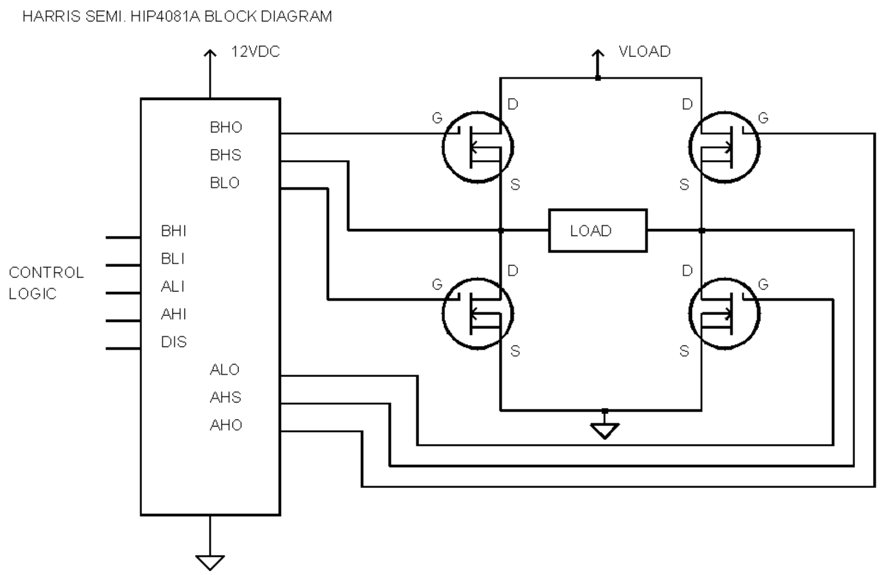
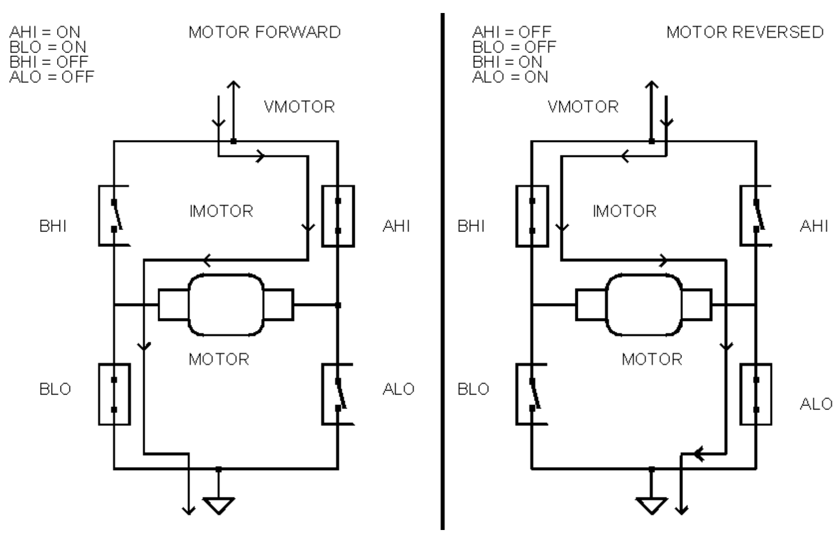



**Figure 52.2: Current Flow through an H-Bridge**



Therefore, for high power applications, you would want to do several things to minimize power dissipation in your circuit. You would use N-channel MOSFETs with low Rdson values. It is desirable to maximize the gate drive voltage in order to minimize the Rdson of your MOSFET. You would also ensure that when you turn your MOSFET on, the gate drive voltage would have a fast rise time to minimize the amount of time that the MOSFET is in its active region (the fall time of the gate drive voltage is also important when turning a MOSFET off).

This brings us to the concept of an H-bridge. The H-bridge is a configuration of four switching devices that allows you to change the direction of current flow through a load. This is done by selecting which pair of switches are on and which pair are off. This is particularly useful for controlling bi-directional motors.

We'll assume that our load is a motor for this design, although many types of loads could be controlled with the circuit that will be defined here. An H-bridge requires that there be two switches located between the system voltage supply and the load, and two switches located between the load and the system ground return. To reduce power dissipation, it is desirable to use N channel MOSFETs for each of the four switches. This creates a significant problem. To fully turn on an N channel MOSFET, the gate drive voltage should be about 12V above the voltage at that MOSFET's source. In an H-bridge, the high side switches will have their source pins at the system voltage level when they are on. For example, in the system being designed here, I'll be using 18-20VDC for the system voltage. That means I'll need roughly 30-32V to drive my high side N channel MOSFETs. Things are definitely getting complicated!

## Connecting the Parts

Have you ever looked at an engineering problem before and said "Hey, this should definitely be easier than it is turning out to be?" Well, I think the people at Harris Semiconductor (www.semi.harris.com) did just that when they designed the HIP4081A H-bridge driver chip.

This device provides a logic level interface to an all N-channel MOSFET H-bridge. It also prevents things like shoot-through (inadvertently turning on the wrong switches and shorting your supply to ground through your H-bridge, usually resulting in transistors with cracks, missing legs, and that funny melted electronics smell).

The HIP4081A has on-board charge-pumps and upper bias supplies that are used to raise the gate drive voltage of the upper MOSFETs above the system bus voltage when they are turned on. This is routinely called a "high-side driver;" two of which are required for an N-channel H-bridge. Figure 52.2 shows the logic states required for changing the current flow in an H-bridge. In many loads, the amount of current passing through the load is required for control, as well as the current direction. With a motor as your load, this translates to controlling both motor speed and direction. To control speed you can turn on the appropriate high side switch, and then pulse width modulate (PWM) the low side switch.

For the HIP4081A, the control pins are considered high for assertion, meaning that a logic high enables that function. For instance, a logic high (+5V) at the ALI and BHI pins would turn on the B high side driver, as well as the A low side driver.

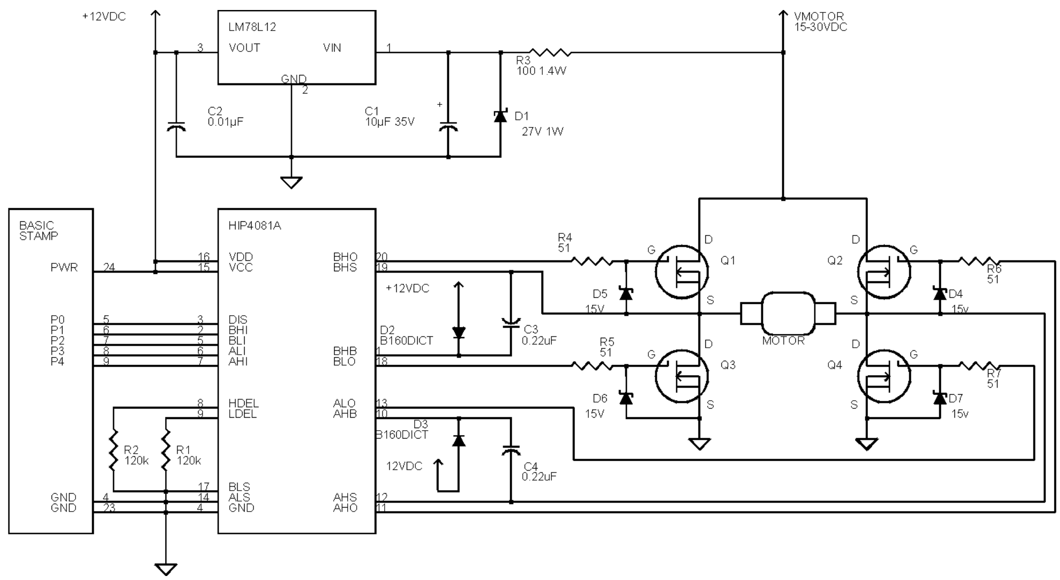
Component selection for the HIP4081A is relatively simple. I selected the Harris HRF3205 MOSFET which has an Rdson of about eight milli-ohms (0.008 ohms) for the four H-bridge switches. The charge-pump diodes selected were from Digi-Key, but virtually any 60V Schottky diode would work for this application. The charge-pump capacitor (C3,C4) should be at least 10 times the input capacitance of your MOSFET. The HRF3205 has an input capacitance of 4000pF. I used 0.22uF capacitors, which are oversized but worked fine. The HRF3205 is rated for 55V and 75A. At 30V and 30A, I think we're well within its safe operating range.

I included a power supply circuit that is required for powering the HIP4081A with the motor voltage. This was also used to provide power to the BASIC Stamp. Some additional protection should be added to any noisy circuit to help protect the logic components. In this case, a bi-direction transient voltage suppressor (TVS) could be placed across the motor. Furthermore, RC snubbers, additional supressors, zener diodes, and poly-fuses could be used to reduce or contain transient voltage spikes.

There are protective gate resistors and zener diodes on each MOSFET which will slow the rise and fall times of the gate drive voltage but they do not have much effect on the performance of this particular circuit. Figure 52.3 shows the finished schematic.
Power dissipation for this circuit at 30A is around 15W. Additionally, this power dissipation is split between two semiconductors (7.2W = 30A2*0.008ohms; P=I2R). This makes heatsinking easier. At 10A, virtually no heatsinking would be required. MOSFETs are also ideal for paralleling. They work well in current sharing applications.

The current that the device passes is based, to a degree, on the Rdson value of the MOSFET (and, of course, on the load resistance and bus voltage). In addition to being dependent on the "gate drive voltage," the Rdson value will increase as the MOSFET's junction temperature increases. So, if a MOSFET gets hot, it will pass less current because of a increasing Rdson value.

**Figure 52.3: 15-30V, 30A Motor Driver**



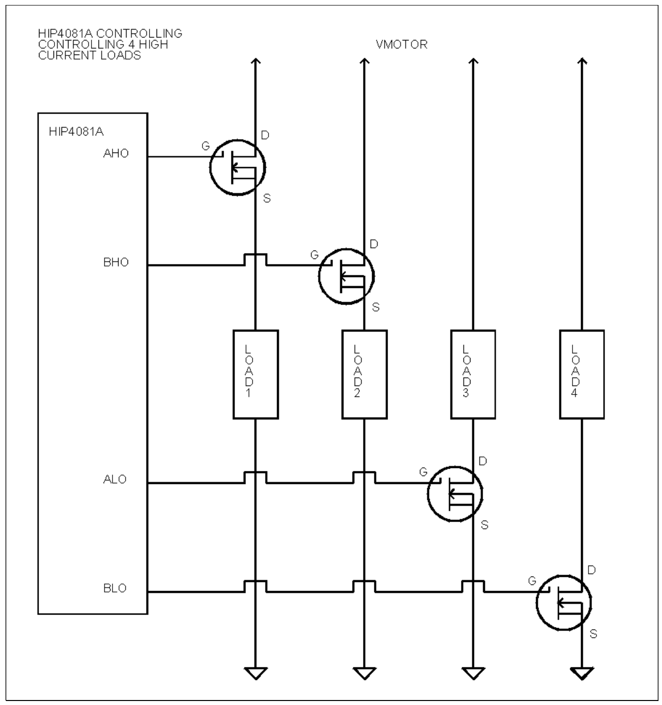

Therefore, if two (or more) MOSFETs are connected in parallel, and one begins to heat up, it will eventually begin drawing less current than the cooler MOSFET of the pair, and subsequently cool down. This is assuming that all of the MOSFETs paralleled are of the same type; 100A plus H-bridge circuits can be realized with parallel MOSFETs and an H-bridge driver like the HIP4081A.

## Writing the Code

The BASIC Stamp has limited PWM capability. The PWM command works just fine, but you can't have it run in the background (perform serial communication, etc., while doing PWM). Many applications will require a separate PWM generating device external to the BASIC Stamp. This application simply controls the HIP4081A by running the motor in one direction, stopping the motor, and then running it in the other direction. This should give most people insight into the simplicity of H-bridge control with a logic level device.

In the code example, the DISpin (disable pin) was used to shut down the H-bridge driver. You can see in Figure 52.4 that the gate drive voltage on the A high side switch stayed at about 30V. The system was running with an 18VDC bus. Since the HIP4081A was owered by a 12VDC regulator, you would expect the gate drive voltage to be 12VDC +

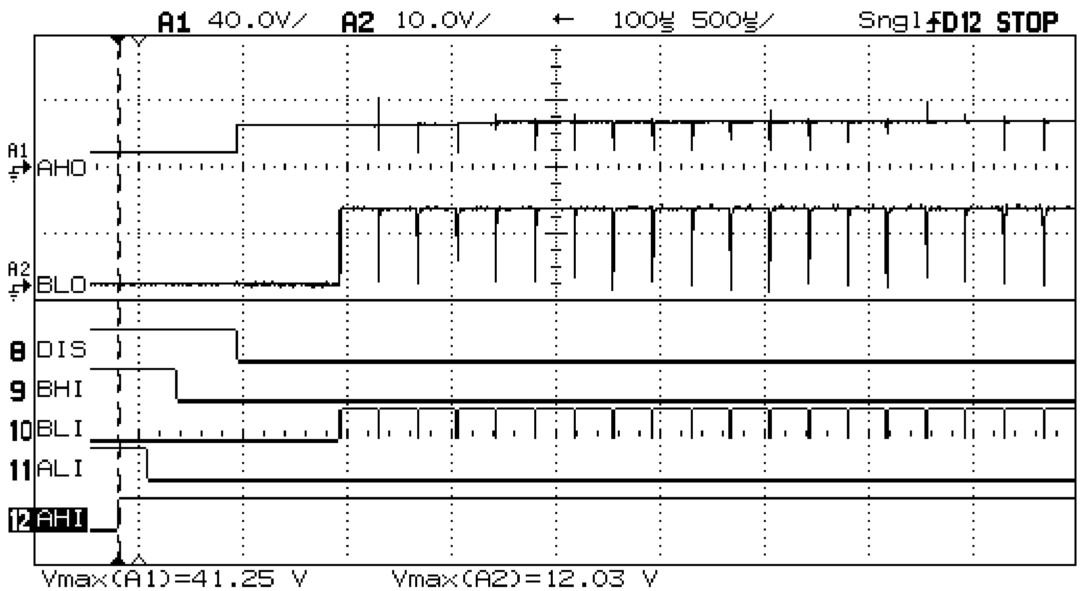
**Figure 52.4: Non- H-bridge applications**




18VDC = 30VDC, which it was. There is quite a bit of noise in this system as it stands. This can be seen as voltage spikes and noise coupling from scope probe to scope probe. Most of this noise would go away if the design were on a PCB and not on a solderless breadboard. The PWM signal can be seen on both the logic level BLI pin, and the low side driver output, BLO.

You can also see that, although the AHO output is enabled in the BASIC Stamp code via the AHIpin, the A high side driver is not enabled until the DISpin is taken low (0V), thus enabling individual control of the various output drivers.

**Figure 52.5: HP54645D Screen Capture of Reverse Command**

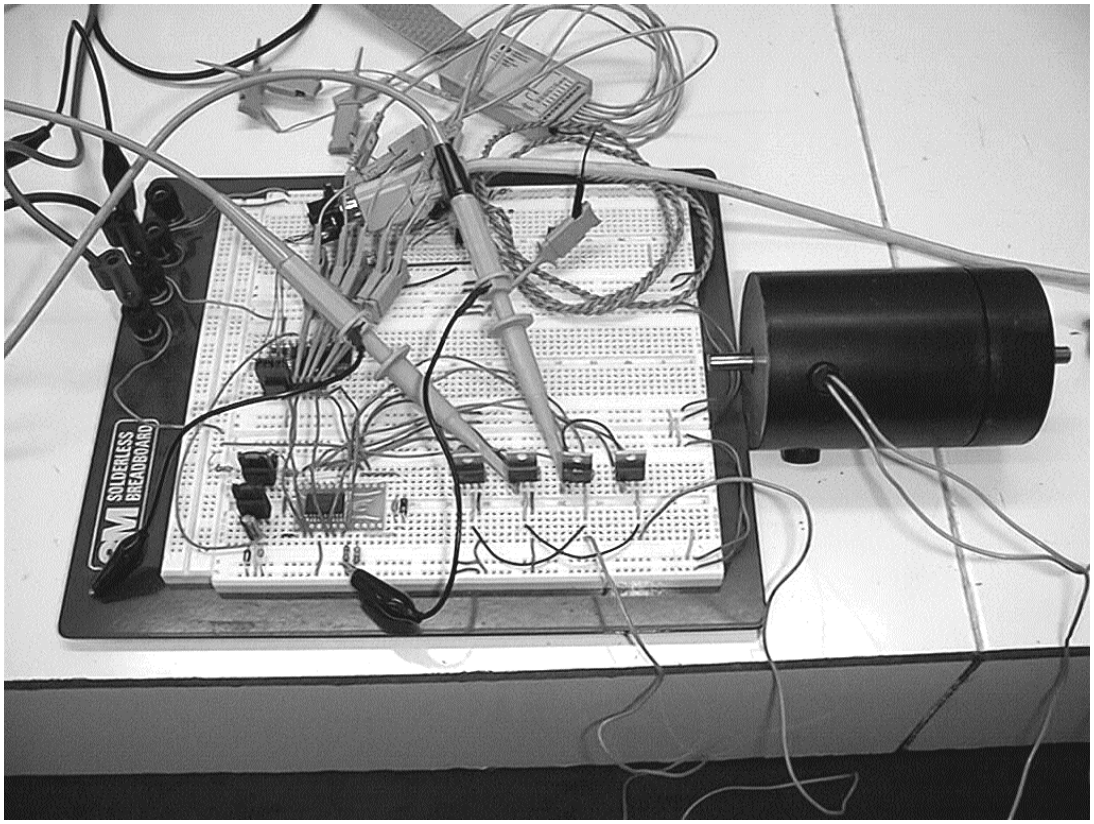



Harris Semiconductor makes a part similar to the HIP4081A, the HIP4080A. The HIP4080A is designed specifically for H-bridge control. For that reason, it would require fewer I/O lines to interface to. But I felt that the HIP4081A provided for a multitude of high current applications other than H-bridge control. So, I felt it might have wider appeal to the readers of this column. The parts are so similar, though, that virtually all information on the HIP4081A that was provided in this article holds true for the HIP4080A. Figure 52.4 gives an idea of how some non-H-bridge applications might appear.

**In Closing**

It's pretty difficult to come up with circuits that will have mass appeal, and fit them into a monthly article format. But I think this is a good one. There are quite a few parts out there in the electronics market that take a lot of anguish out of what would otherwise be a bear of a design. The Harris Semiconductor parts that I detailed here are great examples of the work many semiconductor manufacturers are doing these days.

**Figure 52.5: H-bridge test circuit**




Stop by and take a look at their web site. And, while you're at it, swing by the Maxim, Linear Technologies, Allegro, and Unitrode web sites. And, if you see anything that catches your eye, drop me an E-Mail. If it seems to work well with a BASIC Stamp, I might just write an article on it (and, of course, give you credit for pointing it out to me).

```
'Program Listing 52.1
'NV_AUG99.BS2 - This BASIC Stamp 2 program runs a high current motor one
'direction, stops the motor, and runs it in 'the other direction. This is
'repeated until power is removed.

DISpin CON    0
'Disable input, logic high disables all outputs
BHIpin CON    1
'B high side control pin, a logic high turns on B high side driver
BLIpin CON    2
'B low side control pin, a logic high turns on B low side driver
ALIpin CON    3
'A low side control pin, a logic high turns on A low side driver
AHIpin CON    4
'A high side control pin, a logic high turns on A high side driver

PWMreg VAR    BYTE   'Pulse width modulation duty cycle register
DURreg VAR    BYTE   'Pulse width modulation duration register
Loop   VAR    BYTE   'FOR…NEXT loop variable

HIGH   DISpin        'Disable H-bridge on power up
PAUSE  500

DURreg =100          'Duration = 100
PWMreg =250          'Duty cycle is 98%

FORWARD
HIGH   DISpin        'Disable all driver outputs(stops motor)
PAUSE  1000          'Wait 1s then set input values
HIGH   BHIpin        'Enable B high side driver
LOW    BLIpin        'Disable B low side driver
LOW    AHIpin        'Disable A high side driver
LOW    ALIpin        'Disable A low side driver(will PWM it below)
LOW    DISpin        'Enable all driver outputs(only B high side in this
case)

FOR    Loop = 1 to 25
PWM    ALIpin,PWMreg,DURreg  'PWM A low side driver
NEXT

REVERSE
HIGH   DISpin        'Disable all driver outputs(stops motor)
PAUSE  1000          'Pause 1s then set input values
HIGH   AHIpin        'Enable A high side driver
LOW    ALIpin        'Disable A low side driver
LOW    BHIpin        'Disable B high side driver
LOW    BLIpin        'Disable B low side driver(will PWM it below)
LOW    DISpin  'Enable all driver outputs(only A high side in this case)

FOR    Loop = 1 to 25
```

```
PWM     BLIpin,PWMreg,DURreg  'PWM B low side driver
NEXT
GOTO    FORWARD        'Change direction again

END
```