

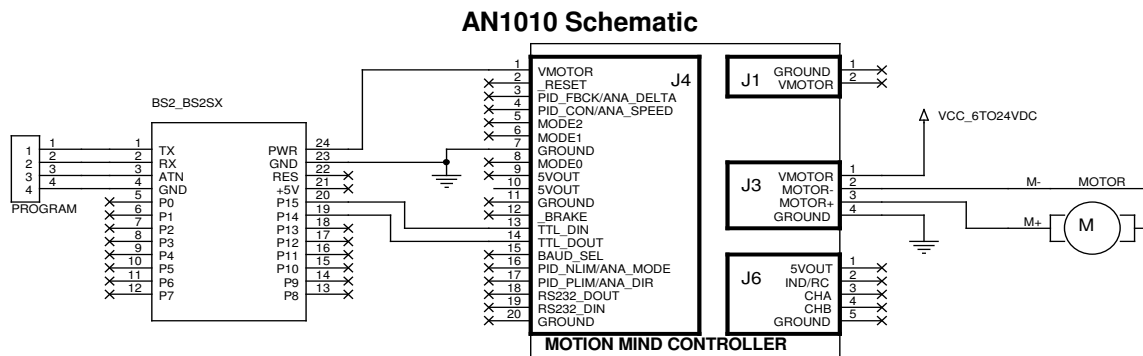
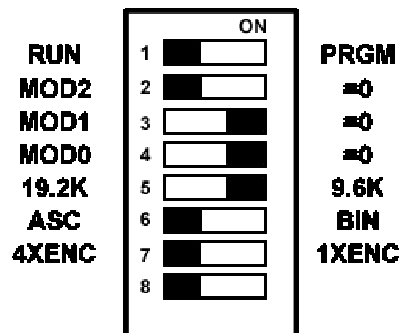
**Overview:**

The BASIC Stamp 2 is a highly versatile controller manufactured by Parallax, Inc. ([www.parallax.com](http://www.parallax.com)). When coupled with the Motion Mind controller by Solutions Cubed, LLC complex control systems may be easily implemented. Application notes here describe methods of interfacing to the Motion Mind. But they only begin to describe the control systems that the BASIC Stamp and Motion Mind can create when used in tandem. The application notes consist of a brief description, jumper settings, and a hardware schematic. The source code for the BASIC Stamp may be downloaded as a separate file (AN101x\_BS2.ZIP). The source code is commented, and is a more useful reference to how the control and functionality are achieved. Questions specific to the BASIC Stamp line of products should be directed to Parallax, Inc.

For these application notes you should set the jumpers at J2 and J5 for the mode of operation and serial communication parameters prior to powering the unit. The schematics have been left as simple as possible to reduce the number of connections required to test the specific applications.

**AN1010 Closed Loop Serial Position Control with a BASIC Stamp 2:**

This application requires a DC brush motor with an encoder (2 channels with or without an "index" output). The Motion Mind should be configured for Mode 4, 9.6KBPS, and ASCII mode of communication. The registers associated with the PID filter may be written to, and three moves are implemented to positions +30,000, -30,000, and 0. Then the moves are repeated. The code shows examples of WRITE, READ, STORE, and MOVETO\_ABSOLUTE commands. Since the BASIC Stamp2 is limited to 16 bit variables, the extent of the move is limited to +32,767 to -32,768.



**AN1011 Open Loop Serial Speed Control with a BASIC Stamp 2:**

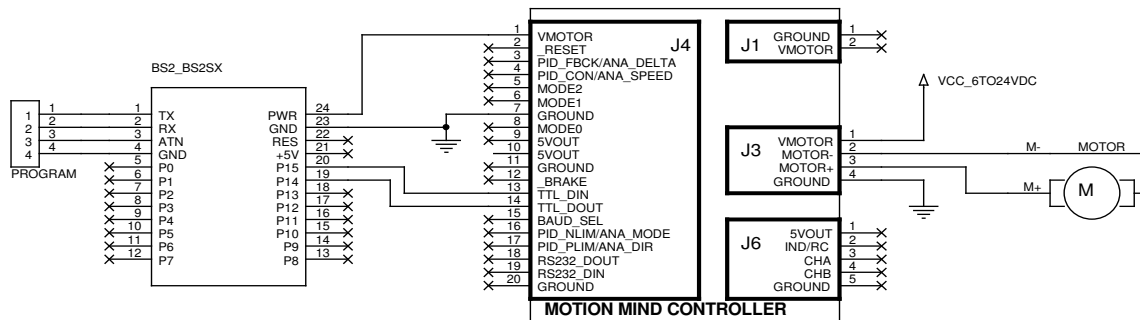
This application requires a DC brush motor. The Motion Mind should be configured for Mode 3, 9.6KBPS, and binary mode of communication. The VELOCITYLIMIT and TIMER registers are written to after power up. By changing the VELOCITYLIMIT and or TIMER registers the rate of change in motor speed may be controlled. As an example lets assume that the current motor speed is 0 and a CHANGE\_SPEED command is sent with a desired speed of 1000. With the default settings of the Motion Mind (VELOCITYLIMIT = 1023; TIMER=10(50ms)) this speed change would occur within 50ms. However if this same command was sent with a VELOCITYLIMIT=1, it would take

$$TIMER * 5MS * \frac{Change\_In\_Speed}{VELOCITYLIMIT} = 50MS * 1000 = 50seconds$$

The best way to get smooth ramping is to adjust the VELOCITYLIMIT register. For example to configure the speed ramp to from 0 to 1000 in 500ms set VELOCITYLIMIT=10, TIMER=1. This would cause the motor to ramp from 0 to 1000 in steps of 10 each occurring 5ms apart. Beyond experimenting with motor speed ramping this application note shows how to send forward, reverse, and stop motor speed commands to the Motion Mind.



AN1011 Schematic

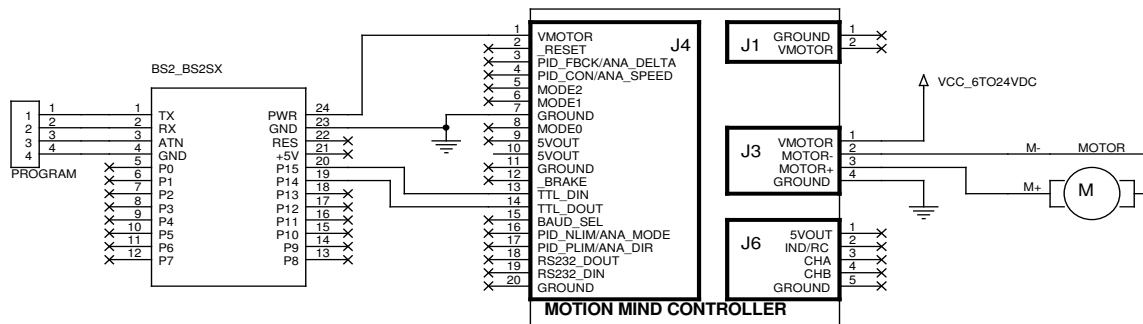


**AN1012 Single Wire Control of the Motion Mind with a BASIC Stamp 2:**

This application requires a DC brush motor. The Motion Mind should be configured for Mode 3, 9.6KBPS, and binary mode of communication. The TTL\_DIN and TTL\_DOUT lines on the Motion Mind are open-collector and are tied together in this application. This allows the BASIC Stamp 2 to implement bi-directional serial communication with a single i/o line. This will not work with the ASCII mode of communication.



AN1012 Schematic



**AN1013 Manual Speed Ramping with a BASIC Stamp 2:**

This application requires a DC brush motor. The Motion Mind should be configured for Mode 3, 9.6KBPS, and binary mode of communication. AN1011 showed how to configure the Motion Mind VELOCITY LIMIT and TIMER registers to perform ramping for you. However, you might want to manually ramp the motor speed using serial commands.

The Parallax BASIC stamp2 doesn't handle FOR...NEXT loops well when using 2's compliment numbers. The code for this application note shows how to hard code in FOR...NEXT loops using the decimal 16 bit equivalent of the desired 2's compliment speed value.

**Two's Compliment Number System**

This Motion Mind can accept negative numbers to represent reversed speed settings. In order to generate a two's compliment negative value take the binary or hexadecimal representation of the absolute value of the number, compliment it (every 1 becomes a 0 and every 0 becomes a 1) and add 1 to the result.

**Two's Compliment Examples**

Number	Absolute Value	Hexadecimal	Compliment	Two's Compliment
-1	1	H'0001'	H'FFFF'	H'FFFF'
-512	512	H'0200'	H'FDFF'	H'FE00'
-1000	1000	H'03E8'	H'FC17'	H'FC18'

MS Windows Calculator in Scientific mode can be used to convert decimal to hexadecimal values.

