**The MiniSSC II**

|  |  |
| --- | --- |
| The Mini-SSC (mini ssc) perfect for servo motion control through the PC or RAPU.  Used with Permission from www.seetron.com | A SSC is a device used to control several servos from a PC across a serial line.  [VSA](http://www.brookshiresoftware.com/vsa_overview.htm)has been specifically engineered for the Scott Edwards Electronics [MiniSSC II](http://www.seetron.com/ssc.htm), the Robot Store's [SMI](http://www.brookshiresoftware.com/rd_how_sscs_work.htm#smi), the Pontech [SV203](http://www.pontech.com/products/sv200bc/index.htm), the [Parallax Servo Controller](http://www.brookshiresoftware.com/rd_how_sscs_work.htm#pololu), and DMX devices.  We will start with the MiniSSC II.  The MiniSSC connects to the PC's communications port at 2400bps or 9600bps.  It accepts formatted data from the PC and converts it into [PWM](http://www.brookshiresoftware.com/rd_how_servos_work.htm) signals for the servos.  Each SSC can control up to eight servos and they can be linked together to control more servos.  The MiniSSC requires two power sources: 9-12v for circuitry and 5v for the servos.  Visit the [MiniSSC II](http://www.seetron.com/ssc.htm) homepage for ordering and more information. |

**Interface**

The MiniSSC requires three bytes to command one servo to a position.  First, a SYNC byte must be sent indicating the start of a sequence.  For the MiniSSC, the SYNC byte is 255 (or 0xFF) and it always represents a new sequence.  Next, the SERVO byte is sent.  The SERVO byte indicates which servo needs to be moved (it cannot be 255).  Finally, the POSITION byte is sent which tells the MiniSSC where to move this servo (it cannot be 255).

[ SYNC ] [ SERVO ] [ POSITION ]

For example, sending "255, 1, 127" would move servo #1 to position 127 (the center point).

The SSC command architecture is simple enough, but consider this: if your robot has 16 servos and runs for a period of just 3 minutes, you will need as many as 86,400 commands -- 259,200 bytes of data.  [Click here to learn how VSA](http://www.brookshiresoftware.com/vsa_overview.htm) can handle the data in the background and create the illusion of a variable speed servo.

**Timing**

Notice that each command sequence requires 3 bytes, or 24 bits, of data.  At 9600 bps (bits/second), the MiniSSC can process 400 commands/second (or a command every 2.5mS).  If there are 8 servos on each MiniSSC, each servo can receive 50 commands/second (or every 20mS).

Timing is by far the greatest challenge for any animator.  Action must occur at just the right times at just the right speeds or the end result is simply poor.  Even if you spend hundreds of man hours perfecting your actions on one machine, when you go to run it on the presentation machine -- everything might be off.  [Click here to see how VSA](http://www.brookshiresoftware.com/vsa_overview.htm) keeps the actions synchronized regardless of the computer.