Controlling Speed & Rate with SconScript

SconScript is designed to be flexible. It is not necessary to specify the speed (rate of servo rotation) during a move. Scon will control and synchronize the rate of all servos for you using a default rate. This default system rate can be changed easily to suit most purposes. SconScript supports additional powerful rate controls that can be used if desired.

It is recommended that the reader have a good understand of how moves are accomplished prior to reading about rate control.

There are three methods to control rate with SconScript: Dynamic Synchronized rate control, Synchronized rate control and Table rate control.

Synchronized Rate:

With Synchronized Rate, all servos will rotate at a rate that causes them to finish the move at the same time. For example, if you specify a rate of 100, Scon will apply that rate to the longest servo move (the servo that rotates the most during the move), then slow down all other servos used in the move so that they all complete the move at the same time. This will make most movements operate in a linear fashion. Synchronized rate is passed to Scon using the “Rate=” keyword in the move instruction.

Example 1: Move 1,000 Rate=100
Example 2: Move 1,000 Rate=250
Example 3: Move 1,023 Rate=10

Dynamic Synchronized Rate (DSR):

This is the default method. Like Synchronized Rate, Dynamic Synchronized Rate (DSR) will cause all servos to rotate at a rate that causes them to finish the move at the same time, however, Dynamic Synchronized Rate is not set in the move instruction. Instead, Dynamic Synchronized Rate uses variables for speed reference. This allows flexible speed control; for example if a move routine in a program uses DSR1 for every move, a change in DSR1 will cause the entire move routine to change speed by changing a single value. There are three sets of DSR variables. The Default System rate references DSR1. All three DSR variables are loaded with the default speed upon power-up or reset. All three speed variables, DSR1, DSR2 and DSR3 can be changed under program control, while your program is running. DSR variables can be operated on using the =, +, or – operators. See example 4 below for an example.

Example 1: Using the default DSR method and default speed

```
Move 10,000              /Move using DSR1
```

Example 2: Using the default DSR method and specifying speeds

```
DSR1=50                  /Set DSR1 to 50
Move 10,000              /Move using DSR1
```

```
DSR1=250                /Change DSR1 to 250
Move 10,001             /Move using DSR1
```

Example 3: Using the default DSR method on one move and DSR2 with another move

```
DSR2=50                 /Set DSR2 to 50
Move 10,000      DSR2   /Move using DSR2
Move 10,001             /Move at the standard system DSR1 rate
```

Example 4: Using the default DSR method and modifying speeds

```
DSR1=50                 /Set DSR1 to 50
Move 10,000             /Move using DSR1 (default)
DSR1+10                 /Increase the speed by 10
Move 10,000             /Move using DSR1 (default)
```

Example 5: Vary the speed, the “yo-yo” program

```
Here                    /Label
DSR1=50                 /Set DSR1 to 50
Do 40                   /Set up a loop counter
Move 10,000             /Move using DSR1 (default)
Move 10,001             /Move using DSR1 (default)
DSR1+10                 /Increase the speed by 10
Next                    /Back to the Do until finished

Do 40                   /Set up a loop counter
Move 10,000             /Move using DSR1 (default)
Move 10,001             /Move using DSR1 (default)
DSR1-10                 /Decrease the speed by 10
Next                    /Back to the Do until finished
Goto here               /Go back and start over
End

Address 10,000          /Set the servo positions for two servos
Position s1=12,500 s2=17,500
Position s1=17,500 s2=12,500
```

Table Rate:

Table rate uses the values form a memory element to specify the rate for each servo. Servo movements are not synchronized under this method, they must be set individually. Speed values are placed in an element just like position values except that when the instruction executes, the data is applied as speed for each servo. Each element contains rate information for all servos just as position data is kept. The primary benefit to using the table rate method is that the move calculates much faster because Scon does not process the data to determine the required rate for each servo. The Element address is passed to Scon using the “Table=” keyword in the move instruction.

```
Example: Move 10,000 Table=32,000        /Move using the speeds at element 32,000
```

Speed and rate details

The actual rotation rate depends on the servos travel and pulse range. These vary by servo manufacturer. For a typical servo with 90 degree full rotation per 1ms pulse change, the following formula can be used: Speed=R/2: Speed is the rotation rate in degrees/second and R is the rate number you pass to Scon. If it seems complicated just start with a number like 50, this will cause full servo rotation in about 3.6 seconds, it’s easy to experiment with speed later.

Rate values are limited to a range of 1 and 1,024. For a typical servo, the lower limit of 1 will cause a full rotation in about 180 seconds. The upper value of 1,024 will cause the same change to occur in .18 seconds, faster than most servos are capable of moving. A rate of 0 will cause an instant pulse change.

The S1=, S1+, or S1- operators (or any servo number) control the PWM module values directly and do not use the move engines. This causes instant changes (or as fast as the servo can travel) in servo position.

Due to servo limitations, most RC servos will not rotate smoothly at many rotation rates. There are several reasons for this: High-speed rate issues result from the 18MS “standard” update cycle for most servos. Scon is compliant with this 18MS standard. With a rate of 500, there are only 20 rotation positions for full servo rotation. A fast servo will stop at each one during the rotation cycle. Slow rotation rate issues are a result of “dead-band”, this is present in all servos. Dead-band is an area (rotation position) that the servo considers the correct position for a given pulse width range. A typical servo will have a dead-band of 7.5us or 75 Scon pulse units. This is quite a large number when you are operating at a travel rate of 1. Many servos will make small incremental changes every 7.5us even though they are sent uniform pulse width change. There are high quality servos available with a dead-band range of 1us or so.