

MRCs Dual Three Position Semaphore Controller V4.1 Slow Motion / Block Signal Software

Software modified by Barry Draper 06-21-2022

This software (Sketch) is designed to operate in the Model Railroad Control Systems Dual 3-way Semaphore Controller board, available at <https://www.modelrailroadcontrolsystems.com/dual-semaphore-servo-controller/>

Basic Installation and Instructions

The hardware And Sketch loading instructions are available at <https://github.com/SethNeumann/MRCs-Dual-3-Position-Semaphore-Controller/blob/master/MRCs-Dual-3-Position-Semaphore-Controller%20v03a.pdf> Please read and refer to these instructions. This document covers the changes made in the Semaphore4.1_BKDSlowmove.ino Arduino Sketch available for download at <https://github.com/barrydraper/MRCs-Arduino-Sketchs>

Loading and Setting Up this Arduino Sketch

Refer to section 5 of Seth Neumann's instruction (above) for installing the Arduino IDE app and connecting to the Arduino board. Download the 4.1BKD Sketch and open it in the IDE. Before uploading the Sketch to the Arduino there are a few simple set ups you will want to adjust

Blade Movement Speed. *Line 27 and 28.* SpdToGrn sets the speed moving toward Green, and SpdToRed sets the speed moving toward Red. Change the numbers (50 & 10) to set your desired speed. A larger number gives slower movement. The defaults result in a fast drop to red and a slow move up to green. Basically the number is the time in milliseconds between 1 degree moves of the servo, so total time will depend on degrees of movement.

Bounce on Return to Red *Line 35 & 37* The BounceArc value determines how far the blade will bounce back as it hits Red. Larger value sets a smaller bounce. A value of 0 sets no bounce. BounceDly value sets how long the bouncing will continue. Large means longer.

US&S Style T-2 semaphores had a direct drive so speed in both directions was nearly the same, and they did not usually bounce, but their Style S had a chain drive with a gravity fall damped by an air cylinder, so they dropped much faster than they were pulled up by the small motor. The Style S also tended to bounce as they hit Red.

Input Functionality. The inputs can be selected to three modes, Block Signal, Latching, or Auto Return to Red.

Block Signal: If *Line 26* Blksig is true (default) when the red input is pulled down that signal will go to red; if red input is high and yellow input is pulled down that signal will display yellow; and if both the red and yellow inputs are high the signal will display green. The green input is not used. This mode is compatible with occupancy detectors with an open collector output such as the MRCS cpOD. A simple Automatic Block Signal system can be assembled with no additional logic if desired. Both red and yellow inputs may be pulled down at the same time, the semaphore will then go to red.

Latching: If Blksig is changed to false and *line 44* latch is true (default) each semaphore will change position following a momentary pull down of each input, and stay at that position until a different input is pulled down. Note that only one input can be pulled down and any time.

Auto Return to Red: If both Blksig and Latch are changed to false each semaphore will default to red, and will only change to red or yellow while the respective input is pulled down. Again, only one input can be pulled down.

In all modes movement speed is controlled by SpdToGrn and SpdToRed.

NOTE: All inputs are 5 volt logic. If any other part of the signal system can pull up to higher than 5 volts each of the inputs of this board **MUST** be protected by a 1N4148 diode with the cathode (band end) facing away from the board.

Setting Blade Positions. With this sketch the pots that set blade position can only be adjusted during the 15 second setting time after a reset using the button on the Arduino (or on power up). Any pot may be set in any order, and the 15 second timer is reset by any adjustment, so you do not need to rush. Mirror is no longer needed. Red, yellow, and green can be set at any point within the range of the servos. The sketch will automatically determine from the pot settings which direction and speed is required.