



To program PWM 1 you will need to do the following math:

$$y = mx - b$$

Where: y is the PWM value to program the FMD1616

m is the slope of the line. For PWM 1 the slope is -0.54

x is the throttle position in the range of 0..100

b is the intercept of the line. For PWM 1 the intercept is 81

The math that you need to perform for PWM 1 looks like this:

$$y = -0.54 * x - 81$$

Now you must take into account the math capability of the FMD PLCs:

The FMD PLCs do not support floating point arithmetic so the slope value of -0.54 is not directly usable. The solution is to use scaled arithmetic.

Multiply both sides of the equation by 100 so the equation becomes:

$$y * 100 = (-0.54 * x - 81) * 100$$

This gets the equation to:

$$y * 100 = -54 * x - 8100$$

The next little FMD issue is that the TBASiC statement, SETPWM expects the PWM percentage to be a scaled integer that is 100 x the PWM duty cycle. For 50% you'd use a value of 5000.

This gets the equation scaled down to:

$$y = -54 * x - 8100$$

This makes the TBASiC statement to program the FMD PWM output #1:

SetPWM 1, -54*x - 8100, 200

Where x is the throttle input in the range of 0..100