Analog I/O Scaling

Scaling for analog inputs uses the traditional slope-intercept formula y=mx+b. Calculation of slope would be m=(y2-y1)/(x2-x1). Once slope (m) is calculated, solve for intercept (b).

Scaling for analog outputs would theoretically be the reverse of this process, meaning subtract intercept then divide by slope. However, the implementation was coded incorrectly. The ValuPoint correctly subtracts intercept first, but multiplies by slope instead of divide. Modifying firmware would result in the same model device in the field behaving two different ways. Rather than create this dilemma, we will simply revise the calculation needed.

Calculation of slope for analog outputs still uses the standard m=(y2-y1)/(x2-x1) formula. But when solving for intercept, the formula is y=(x-b)*m.

Analog outputs produce 0-20mA given a raw value of 0-100 (with percent being implied). To generate a traditional 4-20mA signal, a non-zero value is needed to get the 4mA offset. The 4mA offset corresponds to a raw value of 20 (or 20% of 20mA). Therefore, the Y values in the slope calculation should be 100 and 20, with X values corresponding to whatever range you wish to scale to for 20mA and 4mA respectively.

Qualifier 1 for the analog output should be the value of the dropping resistor at the other end of the loop. If left set to 0, it will assume 500 ohms by default. If the value is anything other than 500, enter that number as the qualifier.

Article ID: 15
Created On: Fri, Oct 17, 2014 at 2:03 PM
Last Updated On: Fri, Oct 17, 2014 at 2:05 PM