Application Note - AN-116

Non-Linear Curve Response

NOTICE

Application Description

This application controls a Pulse Width Modulation (PWM) output based on changes to a variable that represents RPM (in this program it is the Integer RPM). The actual PWM signal is non-linear and is divided into segments. The PWM output is controlled by a slope calculation with each segment. Each segment is defined with a minimum RPM input and a minimum PWM output. To change the 'curve', you will need to edit the defaults of several variables (X_RPM_1 through X_RPM_6 and Y_PWM_1 through Y_PWM_6).

The PWM output can be used to drive a proportional valve for speed control. The RPM variable can be replaced for fed by other circuits as needed (such as shown in AN-112 (Speed Measurement).

Please see the ladder diagram project for calculations and formulas.

Equipment Used

Harsh Environment 1XXX Series		
Controller Part #:		HEC-1000-E-R
Programming Software:		EZ LADDER Toolkit
Digital I/O:		On -Board
Application Filename:		AN116-HEC1X.dld
Programming Cable:		HEC-910 & Null Modem

This Application Note applies to any Divelbiss Controller that programs with EZ LADDER Toolkit, can calculate RPM (high speed counter) and Pulse Width Modulation (PWM) Outputs. To use other targets, generally, only a few program changes are required (typically I/O assignments). Some of the controllers are: PCS, HEC-2000 and HEC-4xxx.

Input / Output Description

PWM1: Pulse Width Modulation Channel 1. This channel is configured for 16 bit and the output signal is on the GPO1 pin.

Divelbiss Corporation • 9778 Mt. Gilead Road • Fredericktown, Ohio 43019 • 1-800-245-2327 • www.divelbiss.com

Program Variables

Big	An integer value with a large default used during calculations. Large value used to prevent divide by zero.
Curve_Index	This Integer variable represents which segment of the response curve is needed based on the RPM input.
Zero	Integer with default of 0. Used for comparisons, setting values and calculations.
One	Integer with default of 1. Used for comparisons, setting values and calculations.
Two	Integer with default of 2. Used for comparisons, setting values and calculations.
Three	Integer with default of 3. Used for comparisons, setting values and calculations.
Four	Integer with default of 4. Used for comparisons, setting values and calculations.
Five	Integer with default of 5. Used for comparisons, setting values and calculations.
Six	Integer with default of 5. Used for comparisons, setting values and calculations.
RPM	Integer value of the actual RPM input signal (in this example 0 - 3200 RPM).
RPM_A	Integer variable that acts as a temporary variable for calculating the slope and response.
RPM_B	Integer variable that acts as a temporary variable for calculating the slope and response.
RPM_D	Integer variable that acts as a temporary variable for calculating the slope and response.
X_RPM_1	Integer that represents the 1st RPM point on the response curve.
X_RPM_2	Integer that represents the 2nd RPM point on the response curve.
X_RPM_3	Integer that represents the 3rd RPM point on the response curve.
X_RPM_4	Integer that represents the 4th RPM point on the response curve.
X_RPM_5	Integer that represents the 5th RPM point on the response curve.
X_RPM_6	Integer that represents the 6th RPM point on the response curve.
PWMOUT	Real Variable that represents the calculated PWM duty cycle for the output.
PWMA	Real Variable that acts as a temporary variable for calculating the slope and response.
PWMB	Real Variable that acts as a temporary variable for calculating the slope and response.
PWMD	Real Variable that acts as a temporary variable for calculating the slope and response.
R_RPM_A	Real Variable that is conversion of RPM_A (integer to real).
R_RPM_D	Real Variable that is conversion of RPM_D (integer to real).
R_Zero	Real with default of 0. Used for comparisons, setting values and calculations.
Temp1	Temporary Real variable used for calculations.
Temp2	Temporary Real variable used for calculations.
Y_PWM_1	Real that represents the 1st PWM point on the response curve.
Y_PWM_2	Real that represents the 2nd PWM point on the response curve.
Y_PWM_3	Real that represents the 3rd PWM point on the response curve.
Y_PWM_4	Real that represents the 4th PWM point on the response curve.
Y_PWM_5	Real that represents the 5th PWM point on the response curve.
Y_PWM_6	Real that represents the 6th PWM point on the response curve.
b	Real Variable used for calculations.
m	Real Variable used for calculations.

Divelbiss Corporation • 9778 Mt. Gilead Road • Fredericktown, Ohio 43019 • 1-800-245-2327 • www.divelbiss.com

Program Description

Rungs 4-6: These are just a convenient listingof the RPM and PWM setpoints used in the response curve.

Rungs 7-8: The Integer RPM (simulated RPM input) variable is converted to a Real variable. This wil be used for calculations.

Rungs 11-13: The RPM is compared to the 1st Segment of the RPM Curve. If RPM is in the 1st curve segment then the Curve index is set to 0. This is used to load the proper curve values.

Rungs 14-16: The RPM is compared to the 2nd Segment of the RPM Curve. If RPM is in the 2nd curve segment then the Curve index is set to 1. This is used to load the proper curve values.

Rungs 17-19: The RPM is compared to the 3rd Segment of the RPM Curve. If RPM is in the 3rd curve segment then the Curve index is set to 2. This is used to load the proper curve values.

Rungs 20-22: The RPM is compared to the 4th Segment of the RPM Curve. If RPM is in the 4th curve segment then the Curve index is set to 3. This is used to load the proper curve values.

Rungs 23-25: The RPM is compared to the 5th Segment of the RPM Curve. If RPM is in the 5th curve segment then the Curve index is set to 4. This is used to load the proper curve values.

Rungs 26-28: The RPM is compared to the 6th Segment of the RPM Curve. If RPM is in the 6th curve segment then the Curve index is set to 5. This is used to load the proper curve values.

Rungs 29-31: The RPM is compared to the 4th Segment of the RPM Curve. If RPM is in the 7th curve segment then the Curve index is set to 6. This is used to load the proper curve values.

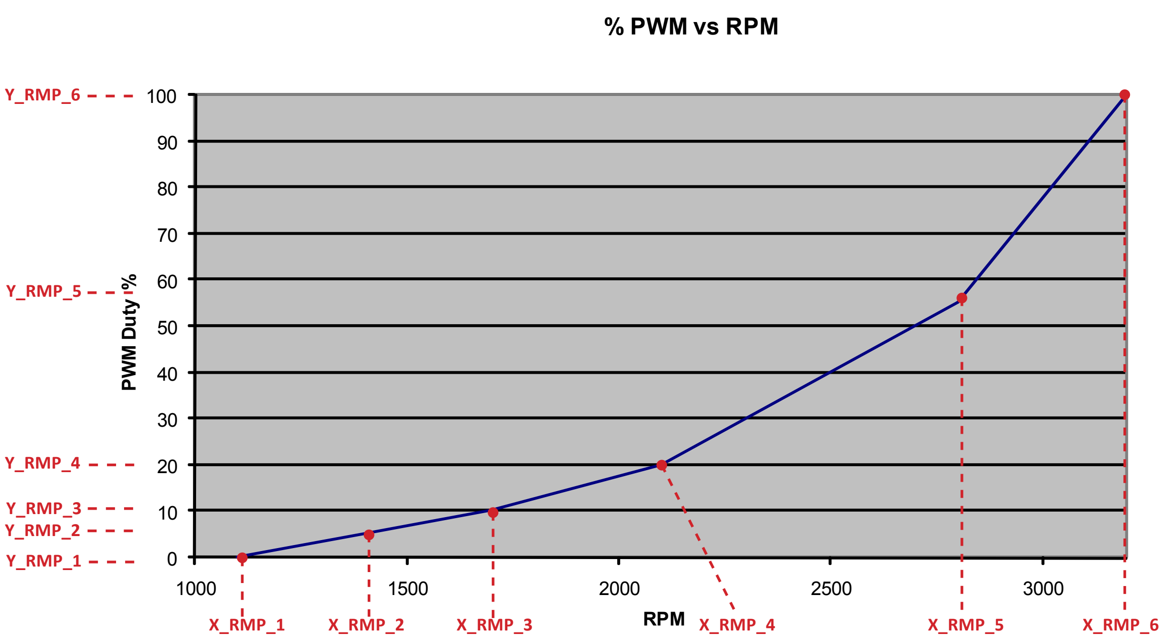
Rungs 34-42: The multiplexers load the RPM (lower end of segment) and the PWM (lower end of segment) into temporary variables that will be used for calculating the slope of the segment.

Rungs 43-51: The multiplexers load the RPM (upper end of segment) and the PWM (upper end of segment) into temporary variables that will be used for calculating the slope of the segment.

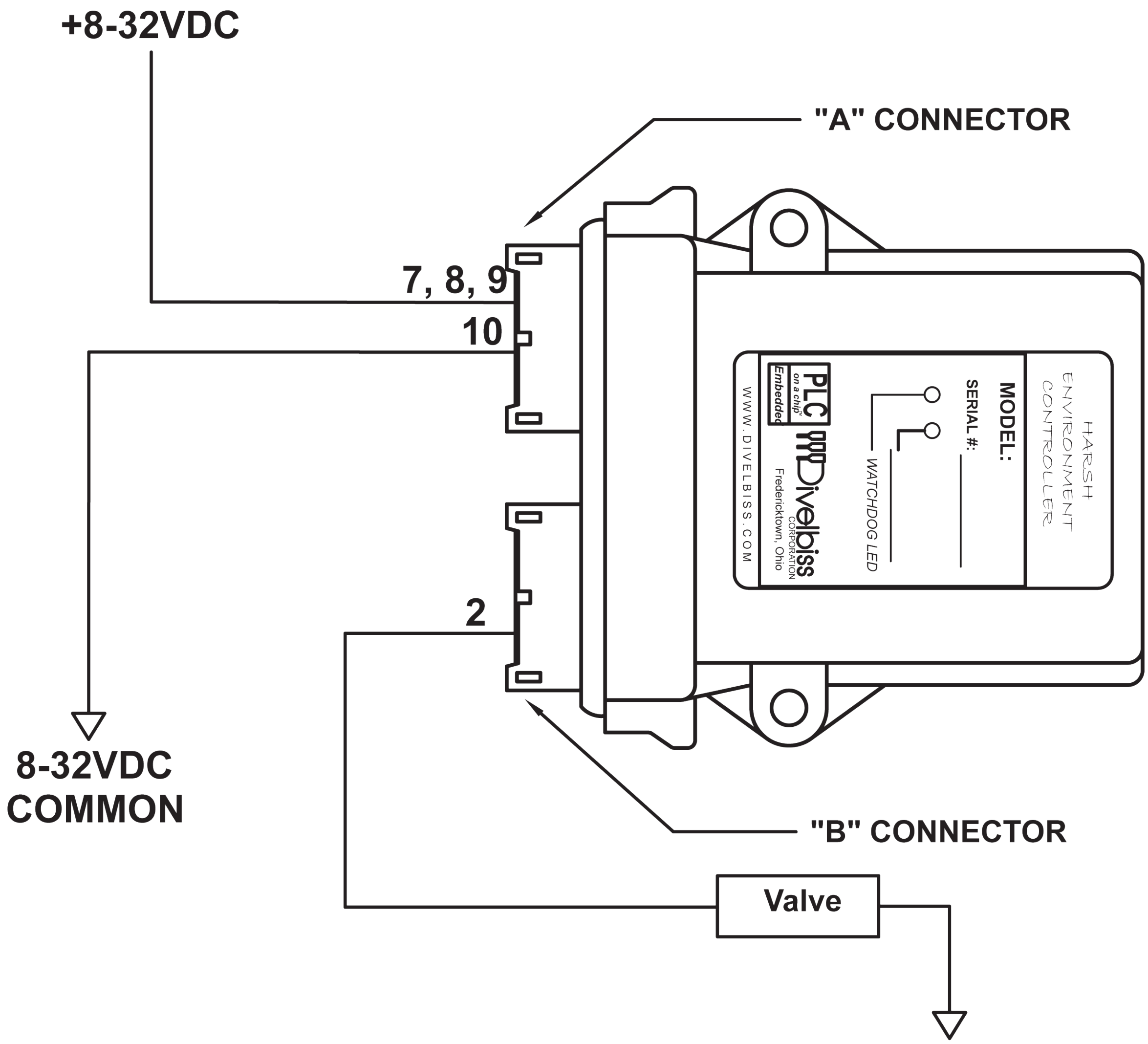
Rungs 52-71: The Slope is calculated using the variables that are gathered from the multiplexers.

Rungs 72-71: Actual PWM output is controlled using the duty cycle (0-100%) based on the PWMOUT calculated in rungs 52-71.

Slope Diagram

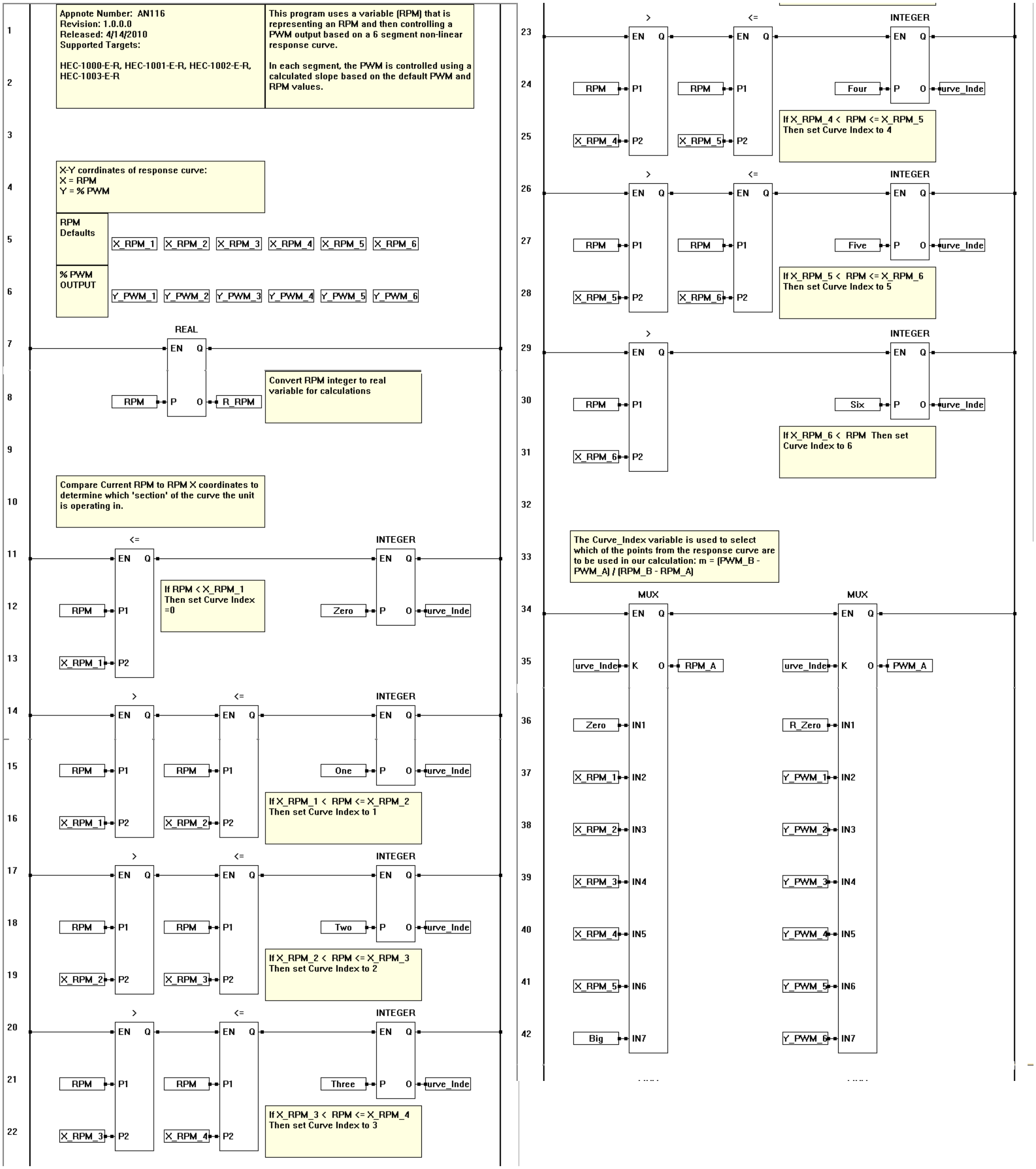


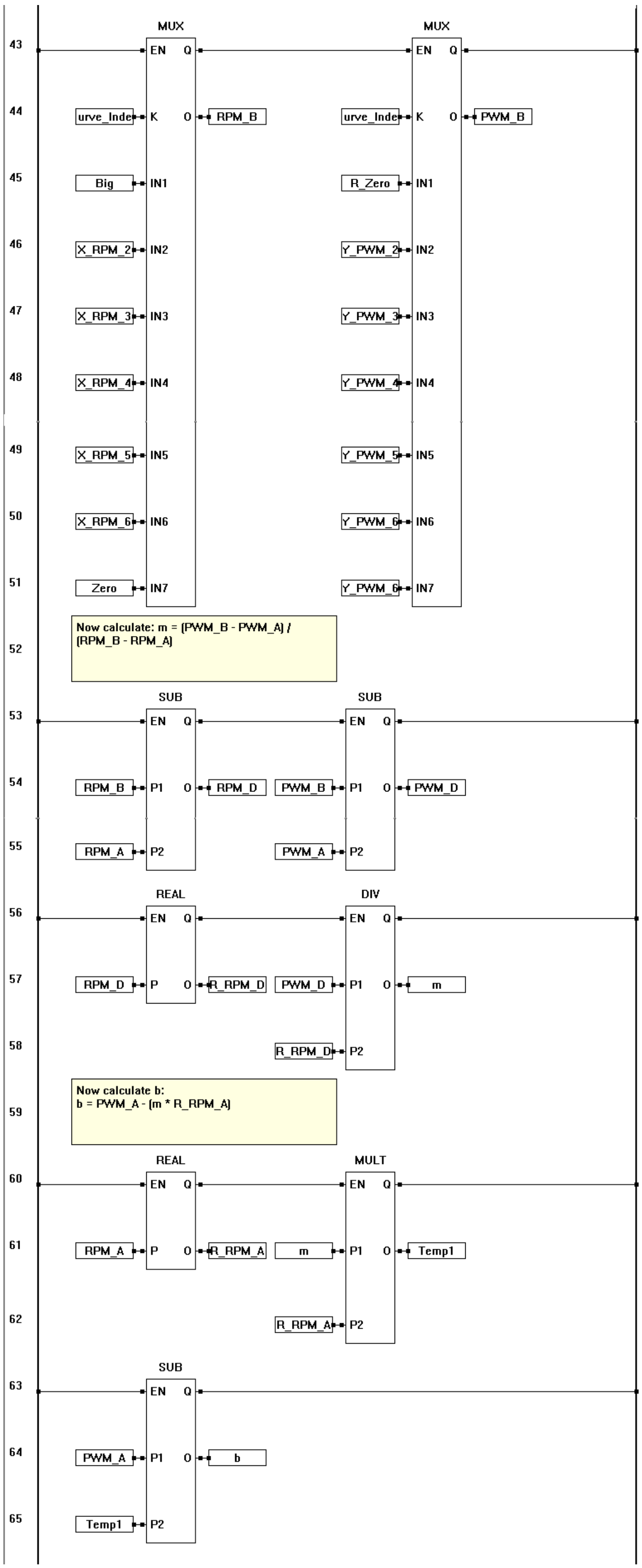
Connection Diagrams

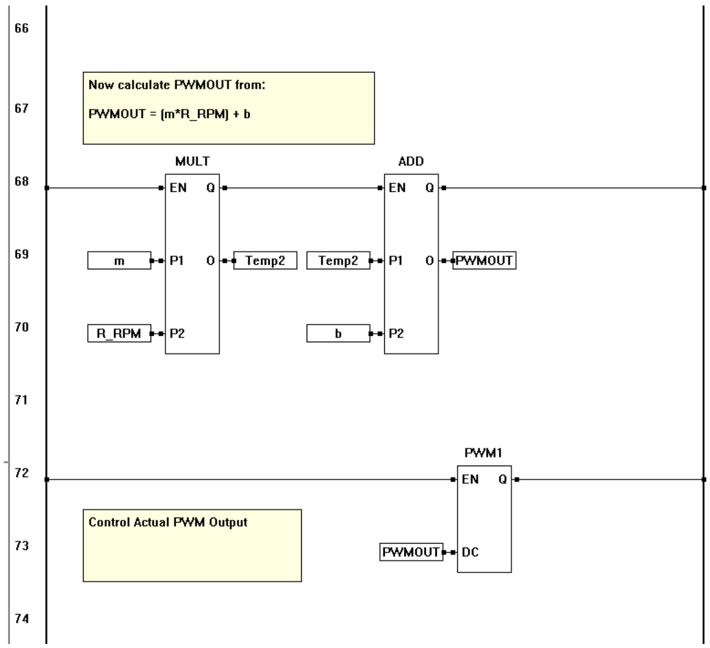


Divelbiss Corporation • 9778 Mt. Gilead Road • Fredericktown, Ohio 43019 • 1-800-245-2327 • www.divelbiss.com

Ladder Diagram




MUX
EN Q
urve_Inde K O RPM_B
Big IN1
X_RPM_2 IN2
X_RPM_3 IN3
X_RPM_4 IN4
X_RPM_5 IN5
X_RPM_6 IN6
Zero IN7
MUX
EN Q
urve_Inde K O PWM_B
R_Zero IN1
Y_PWM_2 IN2
Y_PWM_3 IN3
Y_PWM_4 IN4
Y_PWM_5 IN5
Y_PWM_6 IN6
Y_PWM_6 IN7
Now calculate: m = (PWM_B - PWM_A) / (RPM_B - RPM_A)
SUB
RPM_B P1 O RPM_D
RPM_A P2
SUB
PWM_B P1 O PWM_D
PWM_A P2
REAL
RPM_D P O R_RPM_D
DIV
PWM_D P1 O m
R_RPM_D P2
Now calculate b:
b = PWM_A - (m * R_RPM_A)
REAL
RPM_A P O R_RPM_A
MULT
m P1 O Temp1
R_RPM_A P2
SUB
PWM_A P1 O b
Temp1 P2


Now calculate PWMOUT from:
PWMOUT = (m*R_RPM) + b
MULT
m P1 O Temp2
R_RPM P2
ADD
Temp2 P1 O PWMOUT
b P2
PWM1
EN Q
Control Actual PWM Output
PWMOUT DC