Application Note - AN-115

Joystick Input

NOTICE

This application note is provided for use as a general example and a guide. Divelbiss assumes no responsibility, liability or warranty regarding this application, its use, functionality or reliability to meet application needs. User assumes all responsibility to ensure all safety precautions are taken when using this application note. This application must not be used alone in applications which would be hazardous to personnel in the event of a failure. Precautions must be taken by the user to provide mechanical and/or electrical safeguards external to this application and controllers shown.

Application Description

This application monitors the voltage from a joystick and controls two Pulse Width Modulation (PWM) channels based on it's position.

The joystick outputs a signal of 1-4VDC where 1VDC is full reverse, 4VDC is full forward and 2.5VDC is center and neutral (no movement). The analog input is converted, scaled and used to drive two Pulse Width Modulation (PWM) channels; one forward and one reverse. As the joystick is moved forward, the larger the duty cycle to the forward PWM channel (scaled 0-100% based on 2.5VDC to 4.0VDC). As the joystick is moved backward, the larger the duty cycle to the reverse PWM channel (scaled 0-100% based on 2.5VDC to 1.0VDC).

Equipment Used

Harsh Environment 1XXX Series		
Controller Part #:		HEC-1010-E-R
Programming Software:		EZ LADDER Toolkit
Digital I/O:		On -Board
Application Filename:		AN115-HEC1X.dld
Programming Cable:		HEC-910 & Null Modem

This Application Note applies to any Divelbiss Controller that programs with EZ LADDER Toolkit, has a 0-5VDC analog input 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 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.

PWM3: Pulse Width Modulation Channel 3. This channel is configured for 16 bit and the output signal is on the GPO3 pin.

AN0: Analog input 0. Connected to joystick provides a signal 0-5V total (1-4V joystick) and outputs an integer value of 0-1023 (0=0VDC, 1023 = 5VDC).

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

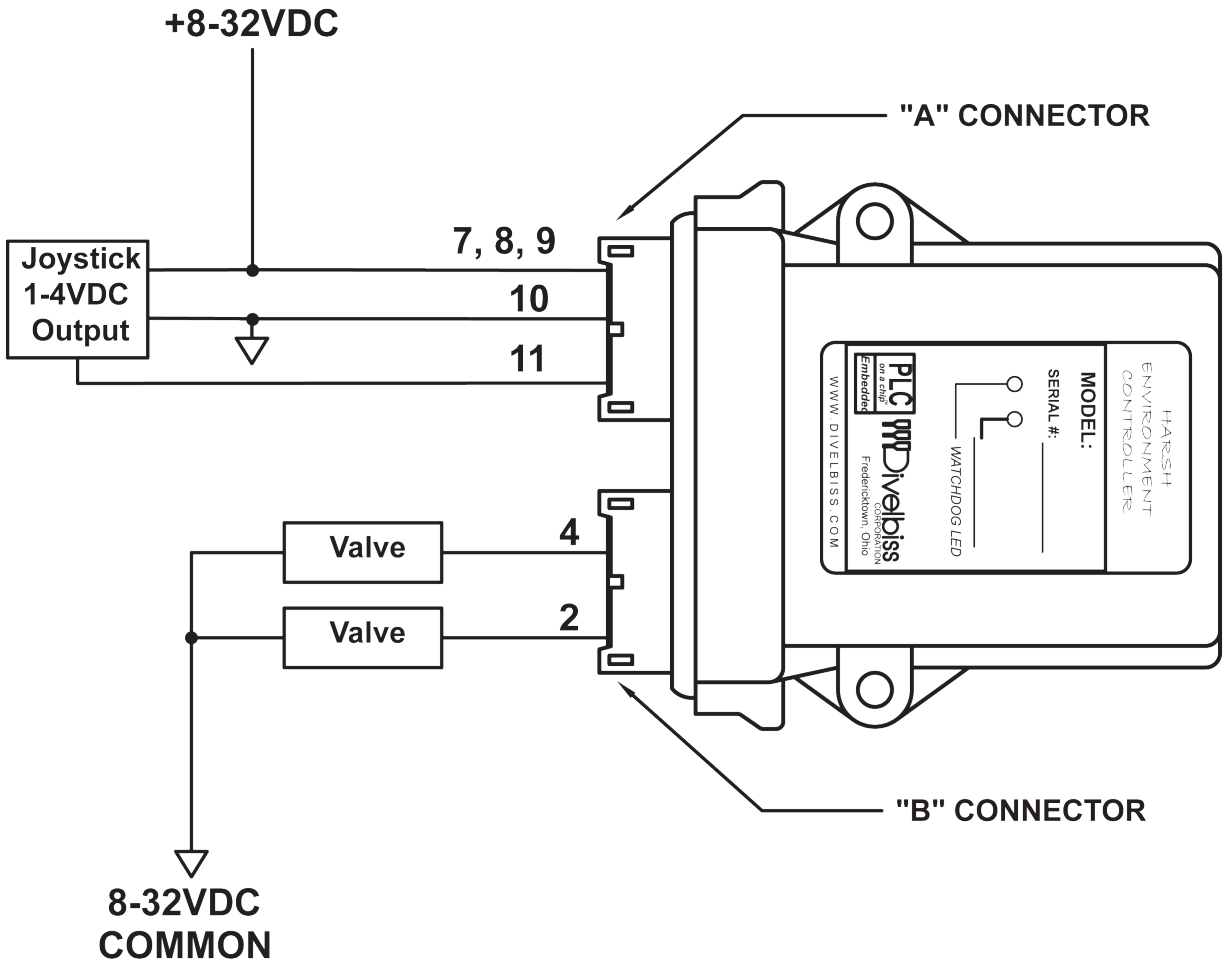
Program Variables

AN0:	Analog Input 0 as an integer (0-1023).
AN0avg	Analog input 0 moving average (10x) as an integer.
FwdPWM:	Integer value representing the duty cycle (0-100%) input of PWM1 (forward).
RevPWM:	Integer value representing the duty cycle (0-100%) input of PWM3 (reverse).
I_100:	Integer with a default value of 100. Used for comparisons and calculations.
I_Zero:	Integer with a default value of 0. Used for comparisons and calculations.
Mulitplier:	Calculated multiplier used to calculate the duty cycle based on the analog input.
R2_5:	Real variable with a 2.500 default value. Used for comparisons and calculations.
R_1:	Real variable with a 1.000 default value. Used for comparisons and calculations.
R_100:	Real variable with a 100.00 default value. Used for comparisons and calculations.
R_1023:	Real variable with a 1023.00 default value. Used for comparisons and calculations.
R_5:	Real variable with a 5.000 default value. Used for comparisons and calculations.
R_AN0avg:	Averaged Analog Input 0, converted to a real variable type.
R_Temp1:	Temporary Real Variable used for calculations.
R_Temp2:	Temporary Real Variable used for calculations.
R_Temp3:	Temporary Real Variable used for calculations.
R_Temp4:	Temporary Real Variable used for calculations.
R_Temp5:	Temporary Real Variable used for calculations.
R_Temp6:	Temporary Real Variable used for calculations.
Vout:	Actual calculated voltage from the analog input (0-5VDC).

Program Description

Rungs 5-6:	Causes a moving average of AN0 and then converts to a real variable.
Rungs 7-9:	Divides the analog input by 1023 and multiplies by 5.00 to scale the analog input 0-5VDC.
Rungs 12-14:	Compares the Joystick voltage (Vout) to 2.5. If greater than then subtracts 2.5 from Vout to get actual voltage above 2.5V and then multiplies by the multiplier to calculate a forward PWM signal (real variable).
Rungs 15-17:	Converts the Real variable PWM forward signal to an integer and compares if > 100. If greater than 100 then limit to 100.
Rungs 18-20:	Compares Vout to 2.5 and if it is greater, always set PWM reverse signal to 0.
Rungs 23-25:	Compares the Joystick voltage (Vout) to 2.5. If less than then subtracts 1.0 from Vout to get actual voltage above 1.0V and then multiplies by the multiplier to calculate a temporary variable (real variable).
Rungs 26-28:	Subtracts the temporary variable from 100 to get the actual real variable duty cycle for reverse direction and then converts it to an integer.
Rungs 29-31:	Compares the reverse duty cycle and if greater then 100 limits it to 100.
Rungs 32-33:	If Joystick Vout voltage is < 2.5 then always set forward PWM to 0.
Rungs 34-37:	Acutal PWM enable function blocks.

Connection Diagrams



Ladder Diagram