Application Note - AN-112

Speed Measurement

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 example is provided as a starting point. It can used individually or used as part of a larger application program. This application note illustrates how to calculate speed based on input pulses from a device such as a hall-effect speed sensor connected to our controller's high speed counter input.

This example is based on a certain set of specifications. You may be required to make variable or even ladder logic changes to accommodate your specific specifications and needs. Multiple controller examples are provided. You must ensure that your pulse rate does not exceed the maximum input frequency of the controller's high speed counter input (varies between controller family).

Speed is determined by a calculation based on the number of teeth on your flywheel or gear; with a 100 millisecond scan rate. Increasing the number of teeth or reducing the scan rate will increase accuracy. Accuracy for each final application should be considered when selecting the number of teeth and scan rate.

Equipment Used

Harsh Environment 1000 Series		
Controller Part #:		HEC-1000
Programming Software:		EZ LADDER Toolkit
Digital I/O:		On -Board
Application Filename:		AN112-HEC1X.dld
Programming Cable:		HEC-910 & Null Modem

This Application Note applies to any Divelbiss Controller that programs with EZ LADDER Toolkit and has a High Speed Counter Input. Some of which are: HEC-1XXX, HEC-2XXX, HEC-4XXX, Solve-It!, Micro Bear, Enhanced Baby Bear and PCS. To use these other targets, generally, only a few program changes are required (typically I/O assignments).

Input / Output Description

CNT 1 : Counter Channel 1. This is where the sensor is connected to read the number of pulses (teeth).

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

Program Variables

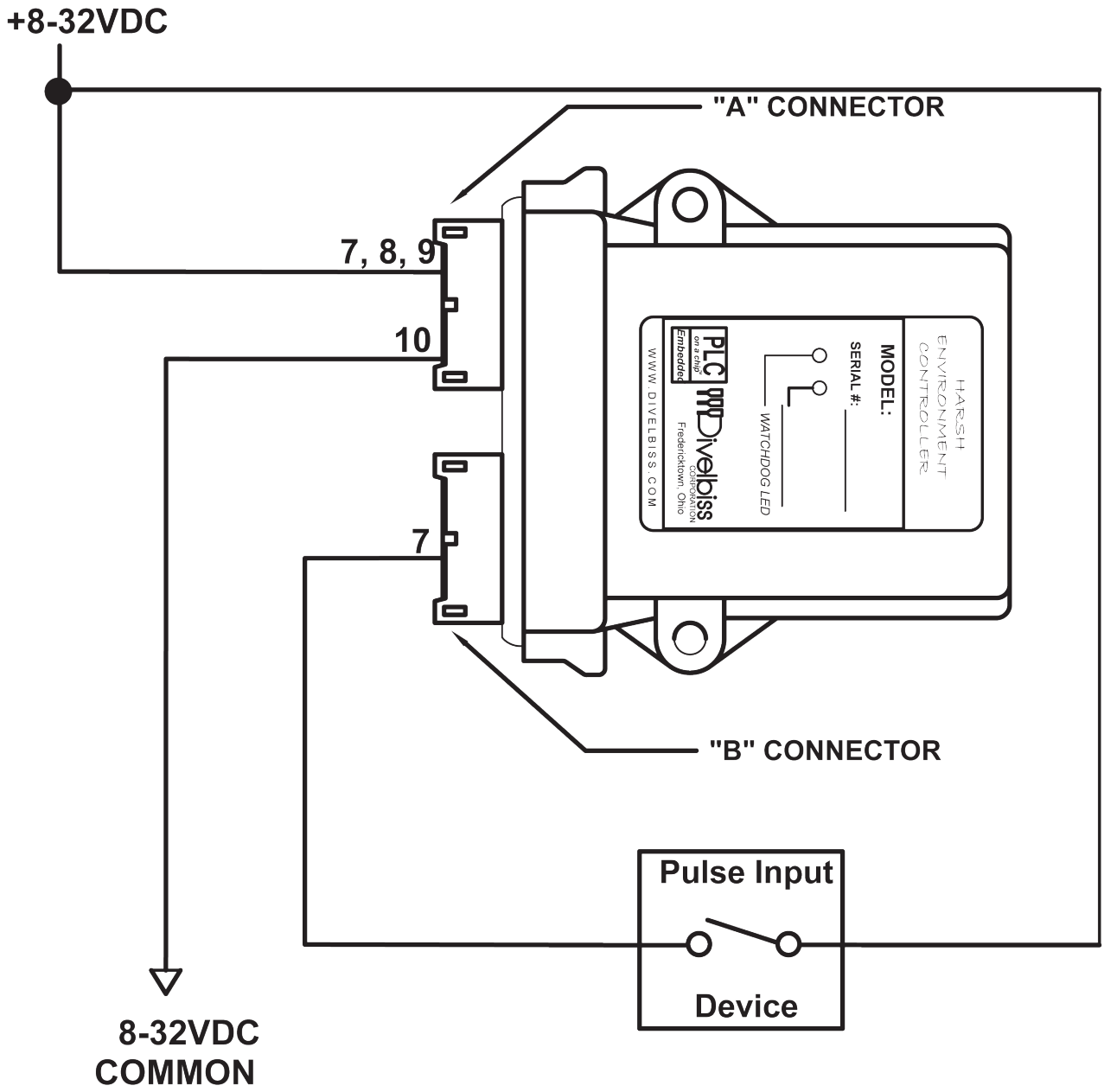
NumTeeth:	This variable is the actual number of teeth on the flywheel or gear. This value is defaulted to 60. To change this value, change the *Default Value* of the NumTeeth variable (using the Variable Edit Dialog).
R_NumTeeth:	This is a the converted value of the NumTeeth as a real variable type.
R_600:	Real variable with a default of 600. Used for calculations.
Multiplier:	Real variable that represents a multiplier needed to calculate speed based on the number of teeth and number of samples per minute.
CR1:	Boolean variable that is used to reset a high speed timer block causing it to restart timing.
CR2:	Boolean variable that is used to reset a high speed timer block causing it to restart timing. This also controls when the current number of pulses is scanned for the calculation as well as resetting the CntTmr function.
TimVal1:	Integer variable used to hold the high speed timer output.
I_500:	Integer variable with a default of 500. Used for comparisons of the high speed timers.
TimVal2:	Integer variable used to hold the high speed timer output.
PulseCnt:	Integer variable used to hold the actual number of teeth counted since the last counter reset.
Cnt_Store:	Real variable used to hold the actual number of teeth counted (converted from PulseCnt). This variable is only updated every 100mSec (when CR2 is triggered).
R_Speed:	Real variable representing the calculated speed.
Speed:	Integer (converted from Real) variable represeting the speed (RPM).

Program Description

Rungs 3-5:	The number of teeth (integer) is converted to a real variable and then the Multiplier is calculated by dividing the number of samples per minute (600) by the number of teeth. The Multiplier will be used in Rung 19 to calculate the Speed.
Rungs 7-9:	The High Speed Timer function will begin timing (free runs) and when it reaches 500 (50mSec), CR2 is set TRUE.
Rungs 10-12:	When CR2 is TRUE this High Speed Timer function will begin timing (free runs) and when it reaches 500 (50mSec), CR1 is set TRUE. (This causes the High Speed Timer in Rung 7 to reset on the next scan which intern clears CR1 itself).
Rungs 13-14:	On the Rising Edge of CR2, the PulseCnt (actual count) is stored as a real variable (Cnt_Store).
Rungs 15-16:	On the Rising Edge of CR2, the CNTRTMR function block is reset.
Rungs 17-19:	The stored count in the last 100 mSec (Cnt_Store) is multiplied by the Multiplier (calculated earlier) to get the current Speed (R_Speed) as a Real variable. This is then converted to an Integer variable (Speed).

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

Connection Diagrams



Ladder Diagram