

How to HART-enable Your Analog Input Module



Garrett Satterfield

The Internet of things (IoT) has implications for Industry 4.0, motivating smart sensor transmitters and similarly capable programmable logic controller (PLC) analog input modules. A [previous blog post](#) illustrated how to achieve digital communication by integrating the Highway Addressable Remote Transducer (HART) protocol into a sensor transmitter design. But that's only half the story. At the other end of the current loop sits the analog input module, which must convert the analog current and HART signals into digital data for processing.

To summarize the previous post, a HART modem couples a frequency-shift keying (FSK) current or voltage signal onto the current loop in a field transmitter. The primary variable (PV) is represented by the DC current value and other data is transferred by the AC FSK waveform. The question is, how do you separate and digitize these signals at the receiver end?

Before designing a HART-enabled analog input module, it is useful to visualize the signal content in the frequency domain. [Figure 1](#) shows the frequency content of a HART-modulated current loop. The analog signal representing the PV is very low frequency, while the HART signal is 1.2kHz-2.2kHz. [Figure 1](#) shows the approximate filtering requirements to separate the signals and avoid interference.

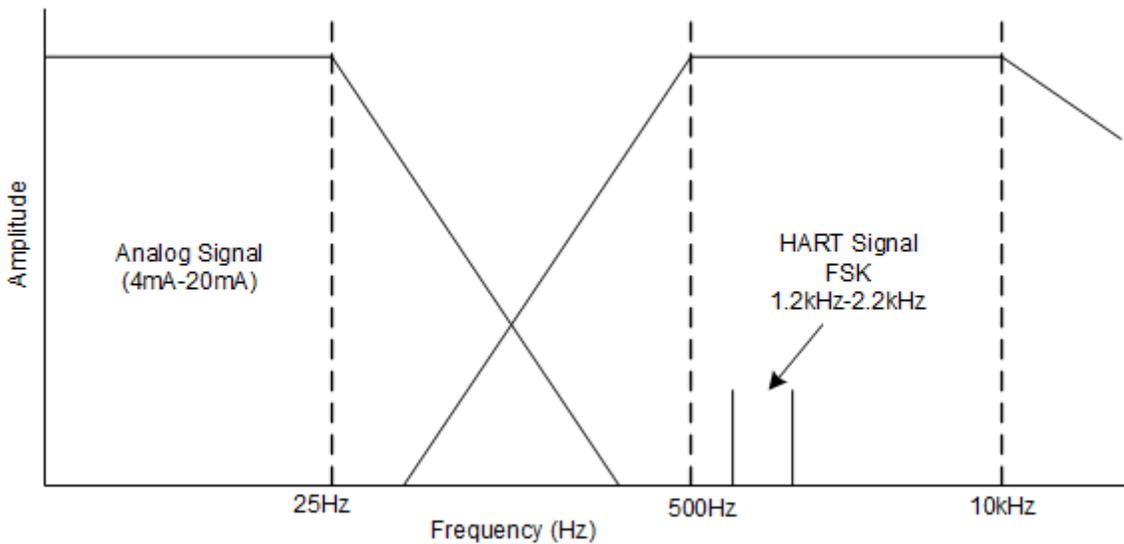


Figure 1. Analog Signal and HART Frequency Bands

Now, you just need a circuit to perform the filtering and convert the signals into the digital domain. [Figure 2](#) shows a HART-enabled 2-wire analog input module schematic. The input module uses the DAC8740H HART modem to demodulate the FSK signal and the ADS8689 to digitize the primary variable.

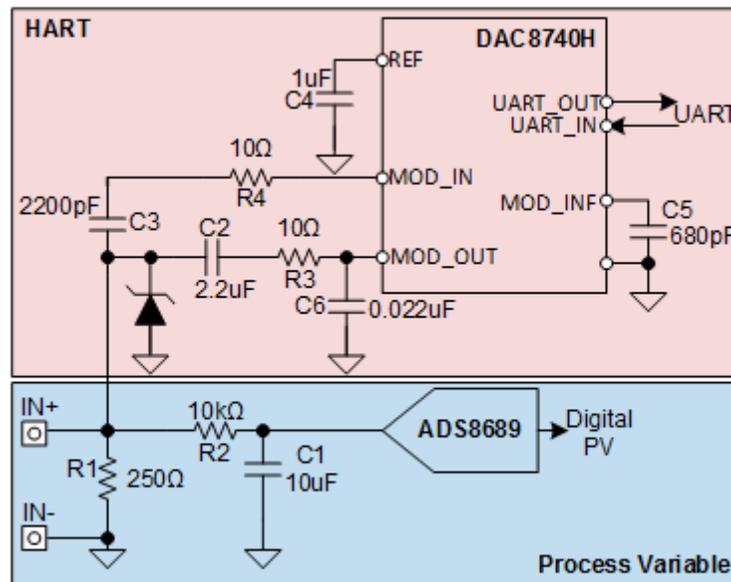


Figure 2. HART Enabled Analog Input Module Diagram

Let's look at how the process variable is captured in this circuit. R1 is the load resistor used to measure the current in the loop. The signal is filtered by a resistor-capacitor (RC) low-pass filter with a very low cut-off frequency. This ensures complete attenuation of the AC HART signal, leaving only the process variable. An analog-to-digital converter (ADC) like the ADS8689 then digitizes the shunt voltage and calculates the current in the loop using Equation 1:

$$I_{loop} = \frac{V_{measured}}{250\Omega} \quad (1)$$

The ADC used to digitize the process variable must have high input impedance in order to avoid altering the effective load resistance, causing errors in the loop-current measurement. The ADS8689 features high input impedance, an integrated analog front end allowing high input voltages and an internal reference.

The denoted HART portion of the circuit consists of the DAC8740H HART modem, with input and output pins AC-coupled to the high side of the load resistor. The DAC8740H's input (MOD_IN) demodulates the received FSK signal and the output (MOD_OUT) modulates an FSK signal to communicate with the field transmitter. Designers often mistakenly think that a current waveform is used for bidirectional communication in HART, but the field transmitter is regulating the loop current and the input module has no control over the current. Instead, the input module uses a voltage waveform to communicate back to the field transmitter. Since HART is half-duplex, both the input and output can couple to the same node.

You can achieve additional filtering of the HART signal within the DAC8740H to isolate the signal before demodulating. The device contains an internal fourth-order bandpass filter that is enabled by connecting a 680pF capacitor to MOD_INF.

Now you know how to achieve two-way HART communication through the analog input module. For more information about HART, 2-wire sensor transmitters and the devices mentioned in this post, see the links below.

Additional Resources

- Learn more about TI's portfolios of precision [DACs](#) and [ADCs](#).
- Jump-start your design with the [highly-accurate, loop-powered, 4mA to 20mA field transmitter with HART® modem reference design](#).
- Read the first blog post in this series, "[Back to the basics: what is HART protocol and how does it work?](#)"

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](https://www.ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2023, Texas Instruments Incorporated