FSK Detection Using PSoC

Dennis Seguine
Member of Technical Staff, Applications Engineering
Cypress Semiconductor, Lynnwood, WA

Abstract

An efficient (almost) all-hardware method for detecting FSK signals using a correlator and analog signal processing in the PSoC is demonstrated. Combined with a UART, this detector forms the core of a half duplex modem.

Introduction

FSK (Frequency Shift Keying) is the modulation cornerstone of a number of digital data transmission systems. The transmit signal is relatively easy to generate. Receive processing is tolerant of a wide range of signals and relatively immune to a large class of interfering signals. This Application Note outlines a simple, almost-all hardware method for detecting FSK signals using the analog signal processing capabilities of the PSoC®.

Operating Frequencies

The FSK signal is represented by:

$$v(t) = V_P \sin(2\pi(f_L + data(f_H - f_L))t) \quad \textbf{(1)}$$

f_L and f_H are the respective low and high frequency modulation frequencies and the data value is logical 0 or 1. (Alternatively, the low frequency can represent a digital 1 and the high frequency a digital 0, as determined by whichever standard is used). Modulation frequencies come in standard pairs for many applications, as listed in Table 1. The example used in this article has operating frequencies of 4,500 and 5,500 Hz modulated at a 600 baud rate.

Detection Basics

The core element in the FSK detector is the correlator, consisting of a delay line and multiplier, shown in Figure 1.

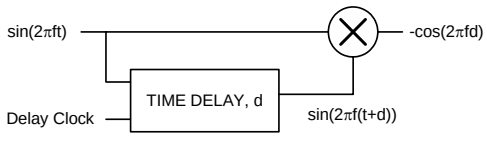

Figure 1. Correlator Block Diagram

The delay line delivers a time-delayed replica of the source signal. The multiplier multiplies the input signal by this delayed replica.

The mathematics of the multiplier are simple:

$$m(t) = \sin(2\pi ft)\sin(2\pi f(t+d)) \quad \textbf{(2)}$$

We'll start with the simplifying assumption that the levels are "unit" sized and fixed. Recalling trigonometric identities from high school, and applying a little algebra, we find the multiplier output:

$$m(t) = \frac{1}{2}(\cos(2\pi f(-d)) - \cos(2\pi f(2t+d))) \quad \textbf{(3)}$$

The DC value represents the data, the 2π*2f term (double the input frequency) is eliminated with a low pass filter. The waveforms for an optimized delay for a typical system are shown in Figure 2.

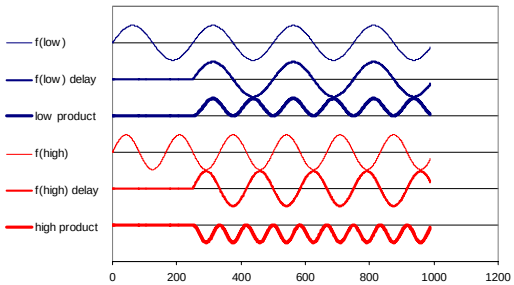

Figure 2. Correlator Waveforms, Optimized Delay

The product waveforms for the two freequencies are distinctly different. The correlator's multiplier output is a DC level that is a clear function of the

operating frequencies and the delay. Eliminating the scaling for simplicity, the difference between the high and low frequency delay products is:

$$Diff = \cos(2\pi f_L d) - \cos(2\pi f_H d) \qquad \textbf{(5)}$$

The design intent is to find a delay that will result in one multiplier output value for the low frequency and a very different value for the high frequency. That is, to find d so as to maximize $Diff$.

An example for f_L=4500 and f_H =5500 is shown in Figure 3. Creative use of a spreadsheet or MatLab shows the correlator output as a function of the delay.

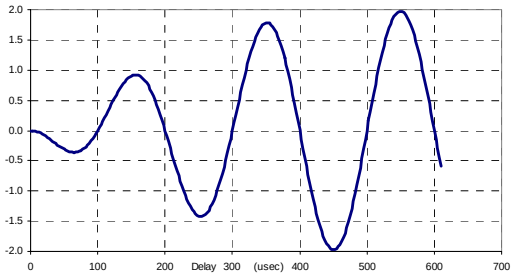

Figure 3. Detector Difference vs. Correlator Delay

So, what constitutes "large enough?" Since we defined these as being unit-sized waveforms, the maximum and minimum signal values after the multiplication operation are 1/2 and -1/2 times the correlator input signal level. We need to set a threshold between these values to determine level detection. We could set the threshold at 0, thus, any positive value represents one frequency and any negative value represents the other. In this case, the difference must be greater than 1/2. While a difference of 1/4 could be enough, there is nothing to assure that both values are not positive, say one at 0.15 and the other at 0.4. In this case, setting up a comparator to find the appropriate non-zero level is less straight-forward than a zero-crossing detector.

As a practical matter, a delay that yields the LARGEST possible difference simplifies the filtering and detection process. For the example chosen, the first peak that meets the minimum separation requirement occurs at a delay of 254 usec with a difference value of 1.5; the next peak at 352 usec with a value of 1.75, and the first FULL peak at 450 usec with a value very close to 2.0.
Detection with the minimum delay will result in the highest detection bandwidth or baud rate and allow the most time for low pass filtering to remove the 2*carrier frequency term from the multiplier.

Correlator Implementation

The multiplication in the correlator is implemented with an analog modulator, shown in Figure 4.

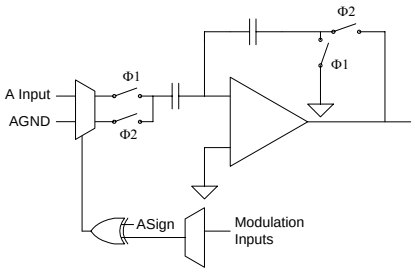

Figure 4. Modulator Schematic

The modulator bit changes the gain of the switched capacitor block from +1 to -1 at the modulation rate. Since the correlator is looking for timing information, not amplitude, the sinusoidal input waveform is turned into a digital waveform with a zero-crossing comparator. The delay function is made from a shift register.

Two easy simplifications are made taking advantage of PSoC block and connection features. The comparator is built into the input band pass filter. The multiplying ASC block, requiring a post-filter, is replaced with a low pass filter (LPF2) User Module with the modulator connection routed from the delay line via software. The completed block diagram is shown in Figure 5.

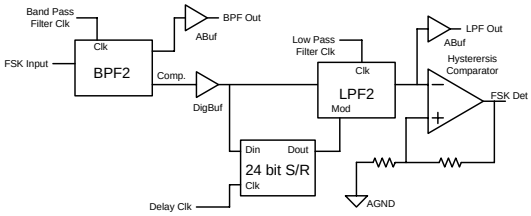

Figure 3. PSoC FSK Detector

Using the filter's comparator, the multiplication process is even easier than with sinusoidal waveforms. The correlator waveforms and output are shown for both frequencies in Figure 6.

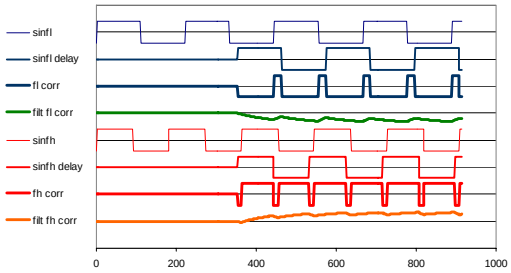

Figure 6. Digital Correlator Waveforms

For the delay selected (450 usec), the peak difference is negative so that at the low frequency the difference signal is negative, and at the high frequency the difference signal is positive. The multiplier output waveform appears as a digital signal with a low duty cycle at f_L and a high duty cycle at f_H. This is filtered to recover the DC level, clearly different for the two modulating frequencies.

Now it's just a simple matter routing the blocks, designing the filters and finding a workable set of clock frequencies to fit both filter and delay line requirements.

PSoC Implementation

The PSoC project is built in a CY8C27xxx part, using relatively little of the part's block, routing and code resources. The analog portion of the block layout is shown in Figure 7.

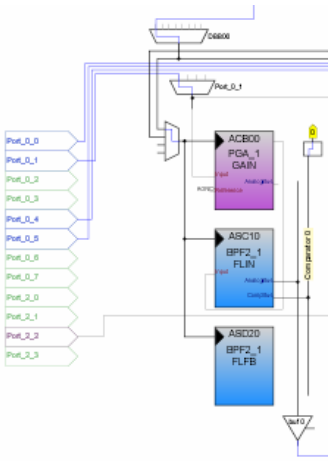

Figure 7. Analog Input

The input to the filter is through a PGA, which allows addition of gain for a wider input dynamic range. The band pass is a single-pole pair filter initially designed for 0 dB gain (V/V = 1.0) and centered at the geometric mean of the two modulating frequencies. The bandwidth is set somewhat wider than the frequency difference. The filter is implemented in two switched capacitor blocks and has a balanced loss of about 1.5 dB at both modulating frequencies.

A four-pole band pass filter, could be used to provide considerably steeper slopes, better out-of-band rejection and a flatter pass-band, . The trade-off is between improved selectivity and the combination of increased block count, increased power and slightly longer filter settling times.

In the example of Figure 5, the analog output of the filters are brought to an analog column outputs so that filters response can be monitored, but these connections are not required.

There is no direct connection from a comparator's logic output to the analog ACap input of the multiplying ASC block, as required in the block diagram of Figure 5. The comparator bus output for column 0 is routed to the Input1 of a DigBuf (digital buffer) User Module; the output of the DigBuf is then routed to a pin. This pin is externally wired to Port2[2], an analog input to the LPF2.

The DigBuf output is also connected to the shift register. The shift register is a modification of the PRS User Module. This user module has selectable feedback taps that are exclusive-ORed back to the input, normally used to generate a maximum length digital sequence. For this application, the polynomial routes the single tap at the end of the shift register chain back to the PRS input. The PRS is hard-wired to invert the output fed back to the input using an exclusive-OR, as shown in Figure 8.

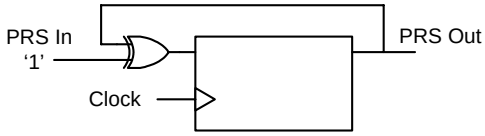

Figure 8. Basic PRS

Exclusive-ORing the S/R input with the PRS output yields a shift register by un-inverting the input to the PRS's internal S/R. The input connection is modified by adding another exclusive-OR as shown in Figure 9.

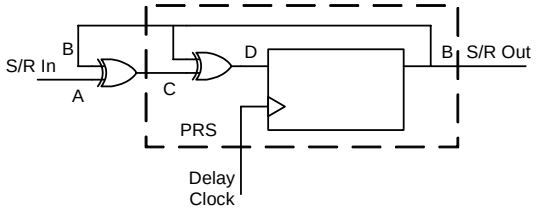

Figure 9. Shift Register Implementation with PRS

The added exclusive-OR is accomplished in the output row LUT. The output of the DigBuf, the input to the shift register, drives one row, which is also routed to a pin. The output of the PRS is exclusive-ORed in the row LUT and connected to a global output bus. This bus is routed to a global input bus via an output-to-input connection, shown on the bottom of the digital routing section. The global input bus connects to a row input. This is connected to the data input of the PRS via software, as the PRS data input is normally wired high for the pseudo-random feedback function. The PRS output on the global bus is selected in software as the modulator source to block ASC23. The complete digital block routing is shown in Figure 10.

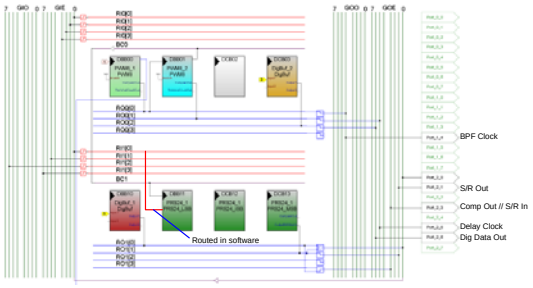

Figure 10. Digital Block Structure

The shift register clock is derived form the 24 MHz system clock. With the delay set to 352 microseconds (selected from data of Figure 5), and a 24-bit shift register, we calculate a delay clock of f_{CLK} = 1/352us/24 = 68.182 kHz. divided down by two PWMs.

The length of the shift register is set by selecting the proper tap on the PRS24 using the WritePolynomial API. In this case, the maximum length is set with dwPoly = 00800000h.

Two signals are routed in software because these settings are not available in the PSoC Designer Device Editor. The PRS24_1 input is re-routed from logic '1' to the row input by modifying the LSB input register. The modulator control source for column 3 is set by writing the appropriate value to register AMD_CR1.

The multiplier is implemented in the input of the low pass filter, as shown in Figure 11.

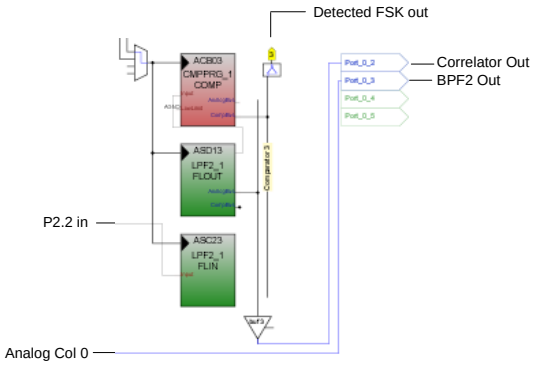

Figure 11. Multiplier and Output

The multiplier output goes nearly from rail to rail. The signal swing is a function of the degree of match to the required correlator delay and the gain of the low pass filter. The filter is designed to have the same sample clock as the band pass filter, synchronous with the delay clock to preclude aliasing problems. The result is a two-pole Butterworth filter at 725 Hz with a rise time of approximately 250 microseconds.

Added to the correlator delay, the response time to a step change in frequency is approximately 450 microseconds.

In order to eliminate false logic triggers caused by noise, the hysteresis comparator shown in Figure 12 is used.

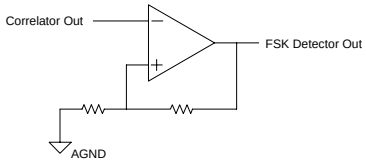

Figure 12. Hysteresis Comparator

The resistor ratio is set to provide a difference between positive-going and negative-going thresholds of 0.5 volts. This comparator is built by modifying the standard CMPPRG User Module in software, altering the selections on the input multiplexers.

The rest of the software simply sets power levels and starts all of the user modules.

Design Demonstration

A separate project was generated in CY8C27443 to provide continuous FSK data for testing the detector, in lieu of using an arbitraty waveform

generator. The project provides a 300 Hz signal and modulates the frequencies at 4.5 and 5.5 kHz. The outputs are a filtered 2.5 Vpp sine wave, and the digital modulation signal at 300 Hz for scope triggering.

The output of the FSK generator is biased at analog ground, so that the FSK detector input needs no additional biasing. Obviously, in the "real world," capacitive coupling to a suitable bias point near analog ground for the detector would be utilized.

The modulator timing sync waveform (from the FSK generator) and the detector's filter output (on P0[3]) are shown in Figure 13. There is a small bump in the envelope as the frequency slides from the low to the high frequency.

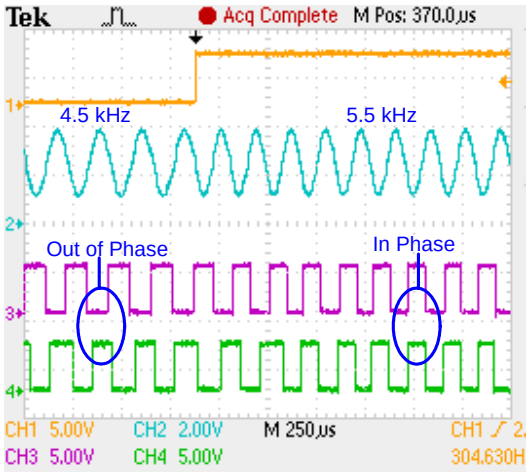

Figure 13. Correlator Waveforms

The comparator output is shown in trace 3, and the delayed copy in trace 4. For the lower frequency, trace 3 and 4 are clearly seen to be inverted relatively to each other. The product of these waveforms is negative. At the higher frequency, trace 3 and 4 are clearly in phase. The product of these waveforms is positive.

The filtered product of the two waveforms is shown in trace 3 of Figure 14. The detection comparator output is shown in trace 4.

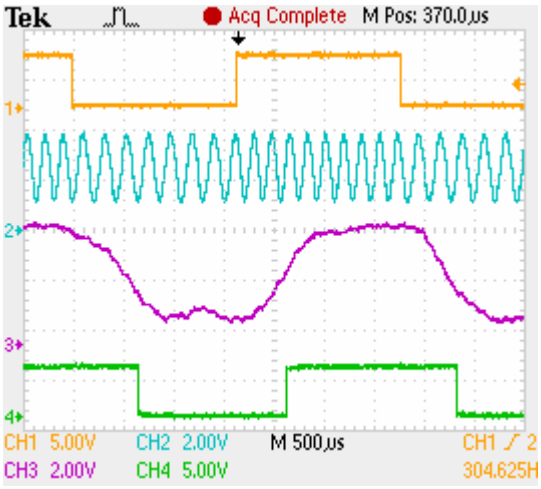

Figure 14. Correlator Output Waveforms

There is a finite time required for the multiplier to get from one level to the other. The transition time is determined by the RMS sum of the input band pass filter response time, correlator delay and the output low pass filter response time. The response time is dominated by the correlator delay. The total response time is approximately 450 usec. The high-going and low-going comparator detection thresholds can be clearly seen.

Design Opportunities

A bare essential FSK detector design has been shown. It uses less than half of the analog capability of the CY8C27xxx, most (but not all) of the digital capacity of the chip, only 36 lines of source code, no run-time manipulations and no interrupts.

There are numerous opportunities for design improvements and adaptation to other operating frequencies, including increasing gain, improving filter selectivity and completion of the connection of the recovered serial digital data via a UART Rx to the CPU's bus.

Acknowledgement

The author wishes to thank his co-worker, the always inventive Principal Applications Engineer **Dave Van Ess**, for the creative connections that generated a functional shift register.

About the Author

Name: Dennis Seguine
Title: Member of Technical Staff, Applications Engineering
Background: BSEE, MSEE, University of California Berkeley.

Manager of Cypress Semiconductor Applications Engineering since 2000.
Contact: seg@cypress.com

Cypress Semiconductor
2700 162nd Street SW, Building D
Lynnwood, WA 98087
Phone: 800.669.0557
Fax: 425.787.4641

http://www.cypress.com/