**Application Note Abstract**
To operate in the presence of noise, many comparators require some hysteresis to allow graceful output transitions. Combining programmable comparator user modules and internal digital programmable logic (LUTs), it is possible to build a comparator with independently selectable threshold levels. A method of building such a comparator is explored and a sample project is included.

**Introduction**
The comparator is the most fundamental building block of mixed-signal design. It has two analog inputs and a single analog output. It is essentially a differential amplifier with an extremely high open loop gain.

For practical purposes, as shown in Figure 1, \( V_p \) greater than \( V_n \) causes the output to be driven to the positive supply rail. And conversely, \( V_p \) less than \( V_n \), causes the output to be driven to the negative supply rail. What about where \( V_p \) equals \( V_n \)? Well “equal” is not a term analog guys have much use for. (You won’t find an analog guy say “exactly 5 volts.”) A good working definition for “equal” is that brief period in the transition from “less than” to “greater than.” (Or for that matter, from “greater than” to “less than.”) In this very narrow (typically 50 \( \mu \)V) transition span, the output is in its linear range. For the ideal comparator, this transition width is zero.

A comparator allows the transformation of an analog signal to a digital form. When coupled with the appropriate digital circuitry it is possible to measure frequency. Paired with a DAC, it becomes a successive approximation ADC. It can, in itself, be thought of as a two-state (single-bit) ADC.

One fundamental problem is that for slow moving input, noise can cause this signal to dance around the threshold, forcing the output to perform a digital cha-cha-cha. One solution is to get rid of noise, a task no harder than ridding the world of hunger. A more practical solution is to have two thresholds. One for transitions from low to high, and a lower threshold for transitions from high to low. The hysteresis Shown in Equation 1 is defined as the difference of these two thresholds.

\[
V_{\text{hysteresis}} = V_{\text{HighThreshold}} - V_{\text{LowThreshold}}
\]  

Equation 1

If the hysteresis is greater than the noise seen at the input then the transition is guaranteed to be smooth. Many circuit topologies exist to generate comparator hysteresis, however most generate a plus and minus value equidistant away from some common threshold. Such topologies make it difficult to independently change either of the threshold levels. It is desirable to utilize a topology that allows independent alteration of the individual thresholds.
How About Two Comparators and Some Digital Feedback

A straightforward method that allows two isolated threshold levels is having two actual comparators and some digital logic.

The desired operation is:

- When the input is greater than the high threshold ($V_{in} > V_H$), the output is always high.
- When the input is less than the low threshold ($V_{in} < V_H$), the output is low.
- When the output is low, the output will not go high until the input is greater than the high threshold.
- When the output is high, it will not go low until the input is less than the low threshold.

Equation 2 is the Boolean representation of these requirements.

$$\text{Out} = \text{High} + (\text{Out} \cdot \text{Low})$$

The implementation is shown in Figure 2.

Figure 2. Hysteresis Comparator Schematic

A PSoC Solution

This example is implemented with a CY8C29466-24PXI, but most any of the PSoC® family of parts could have been used. This part was chosen because it allows the project to use the parts shipped in the PSoC Evaluation Kit (CY3210-Eval1). This project can easily be cloned to other packages or PSoC families. This is an exercise left to the reader.

This project consists of an analog input on P0[7] and a digital output on P0[6]. The high threshold is set to 75% of the supply voltage while the low threshold is set to 25% of said supply.

A PSoC solution requires two parts:

- Two programmable comparators
- Logic to implement the digital section

From the parameters in Figure 4 it is apparent that the high threshold is set to 75% of the supply voltage while the low threshold is set to 25%.

Figure 3. Dual Comparator Block Placement

Figure 4. Comparator Parameters

Also, each comparator is connected to its respective column comparator bus.

Implementing the digital logic requires two DigBuf user modules renamed “HighBuf” and “LowBuf.” These buffers allow the comparator buses access to the digital rows. They also serve as interconnects allowing input rows to be connected to particular output rows. The block placement and interconnects are shown in Figure 5.

Figure 5. Digital Block Placement and Routing
These buffers, along with the interconnect feedback and digital row logic (LUTs), allow the required logic to be implemented. Following the connections through the LUT shows that the Boolean expression stated in Equation (2) is implemented.

All the software required is code to start the analog user modules. The digital buffers default to an “on” power on state. The program is shown below in example Code 1.

Code 1
//------------------------
// Hysteresis Example
 //------------------------
#include <m8c.h>
#include "PSoCAPI.h"
void main()
{
    HighThreshold_Start(HighThreshold_HIGHPOWER);
    LowThreshold_Start(LowThreshold_HIGHPOWER);
    while(1);
}

The pinout for this example project is shown below in Figure 6.

Figure 6. Pinout for Example Project

The output does not go high until the input exceeds 3.75V and only returns low when the signal is less than 1.25V. There is 2.5V of hysteresis.

Quickly changing the lower threshold to 50% and running under the same conditions, Figure 8 shows that the lower threshold has changed.

Summary

Comparator hysteresis allows for threshold detection in the presence of noise. The two comparator and digital logic allows for independently adjustable thresholds. PSoC, with its programmable comparators, digital block buffers, and LUTs, make easy implementation of this circuit topology. Under software control, each threshold can independently be changed.

Note: This project is configured to operate at 5V. The comparator thresholds are set to 75% and 25% of the supply voltage. A 4.6-V 1-kHz sinusoid, centered around 2.5V, stimulus is fed into the signal input. This signal along with the digital output, is displayed in the oscilloscope screen shot shown in figure 7.

Figure 7. 75% High Threshold, 25% Low Threshold

The output still goes high only when the input exceeds the 3.75V threshold, but now only goes low when the input drops 2.5V. There is now a 1.25V hysteresis.

Figure 8. 75% High Threshold, 50% Low Threshold

Proof is in the Pudding

This project is configured to operate at 5V. The comparator thresholds are set to 75% and 25% of the supply voltage. A 4.6-V 1-kHz sinusoid, centered around 2.5V, stimulus is fed into the signal input. This signal along with the digital output, is displayed in the oscilloscope screen shot shown in figure 7.
About the Author

Name: Dave Van Ess
Title: Principal Application Engineer
Cypress Semiconductor

Background: An Engineer by training, a poet by temperament, and an outlaw in Nebraska. Dave is capable of abstract thought, concrete analysis, and ruthless implementation. BSEE from University of California, Berkeley. More than 28 Years experience in circuit, signal processing, digital, software, analog, and system design. Holder of six U.S. Patents (plus three pending) for medical systems, signal processing, and digital block enhancements. Author of numerous Application Notes, web casts, and technical articles. Joined Cypress MicroSystems at the dawn of the Millennium.

Contact: dwv@cypress.com

In March of 2007, Cypress recataloged all of its Application Notes using a new documentation number and revision code. This new documentation number and revision code (001-xxxxx, beginning with rev. **), located in the footer of the document, will be used in all subsequent revisions.

PSoC is a registered trademark of Cypress Semiconductor Corp. "Programmable System-on-Chip," PSoC Designer, and PSoC Express are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
http://www.cypress.com/

© Cypress Semiconductor Corporation, 2005-2007. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

This Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.