AN60220

PSoC® 3 / PSoC 5LP Multiplexed Comparator

Author: Rajiv Vasanth Badiger
Associated Project: Yes
Associated Part Family: All PSoC 3 and PSoC 5LP parts
Software Version: PSoC® Creator™
Related Application Notes: None

If you have a question, or need help with this application note, contact the author at nidh@cypress.com

AN60220 describes a way to multiplex a comparator such that it can monitor multiple signals. The PSoC® 3 and PSoC 5LP device class has either two or four comparators. Multiplexing offers a solution as long as the number of slow varying signals is greater than the available number of comparators in the device. The solution uses configurable digital and analog hardware that is present in PSoC 3 and PSoC 5LP, and does not require CPU power for continuous operation. This document includes a PSoC Creator™ project that demonstrates the comparator multiplexing.

Introduction

Analog resources such as comparators, amplifiers, ADCs, and DACs in a device are precious resources. They are generally not available in large numbers in a single device. The PSoC 3 and PSoC 5LP (hereafter referred to as PSoC) devices have a maximum of four comparators in their analog arsenals that can be used for analog signal comparisons. However, in some cases, multiple (more than four) signals must be compared to a reference signal. With PSoC 3 and PSoC 5LP, you can do this in several ways:

- **Using opamps**

  The PSoC 3 and PSoC 5LP device have up to four opamps. Because these opamps can be operated in open loop, they can be used as comparators. However, this requires two extra pins if the output needs to be routed back to the digital section.

- **Using special I/Os (SIOs)**

  The PSoC 3 and PSoC 5LP devices have SIOs that can be used for comparator applications. Either four or eight SIOs are available depending on the specific device. For more information, see AN60580.

- **Multiplexing comparator**

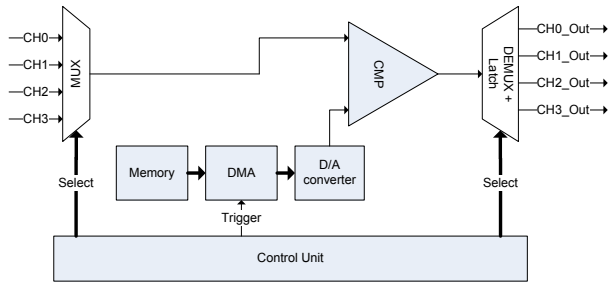
  Comparators can be multiplexed in a similar way as ADCs, in which each channel is scanned one at a time. This solution is useful for low frequency signal applications.

This application note describes a technique for multiplexing the comparator for multiple signals. This technique is a complete hardware solution that uses 0% CPU time.

Multiplexed Comparator Design

In this design, a four-channel multiplexed comparator is implemented as shown in the block diagram in Figure 1. This design is easily scalable for different number of channels. Each channel is selected one at a time and compared with the DAC value. There is a unique DAC value for each channel. This means, for each channel, a unique threshold value can be set. The output of the comparator is demultiplexed to individual outputs. The demultiplexer is synchronized to the multiplexer. Threshold voltage is set by the DAC, which is updated by direct memory access (DMA) hardware. DMA takes the threshold value from memory set by the user.

Figure 1. Block Diagram of Multiplexed Comparator



Features

- Unique threshold voltage for each channel
- Flexibility in adjusting the number of channels
- Non overlapping channel switching (break before make switch)
- No CPU involvement in channel multiplexing
- Configurable scanning rate
- Dedicated Latch for each channel
- Option of CPU reading the channel status
- Option of triggering interrupt on each channel output

Implementing Multiplexed Comparator in PSoC 3 and PSoC 5LP

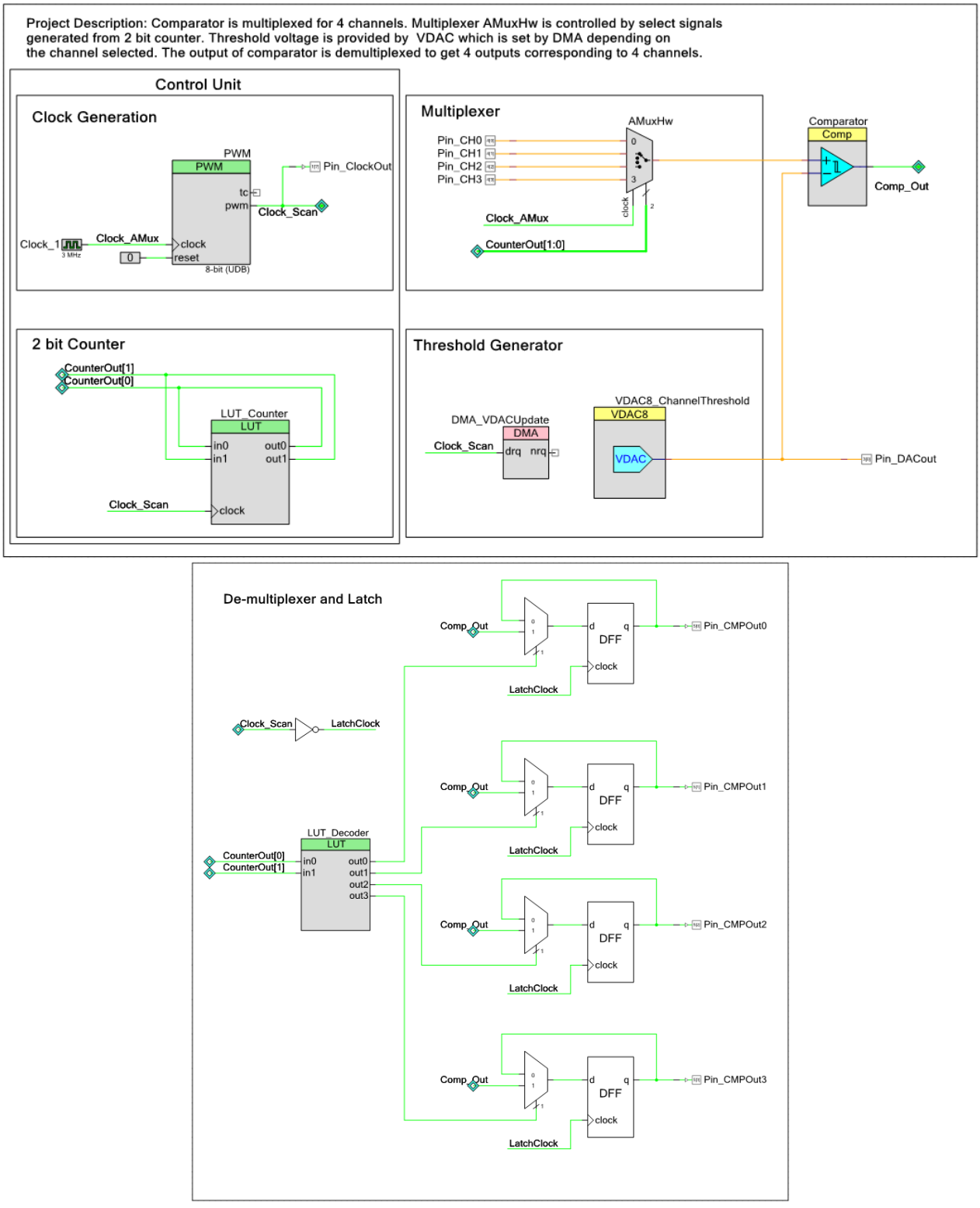
As shown in block diagram in Figure 1, the following resources are required to build a complete solution:

- Control unit
- Multiplexer for analog signals
- Voltage DAC
- Direct memory access (DMA)
- Comparator
- Digital demultiplexer and latch

The PSoC 3 and PSoC 5LP provide all the necessary hardware to implement all of these functions. PSoC Creator simplifies the design task with the customizable components.

Figure 2 shows the multiplexed comparator design made in PSoC Creator.

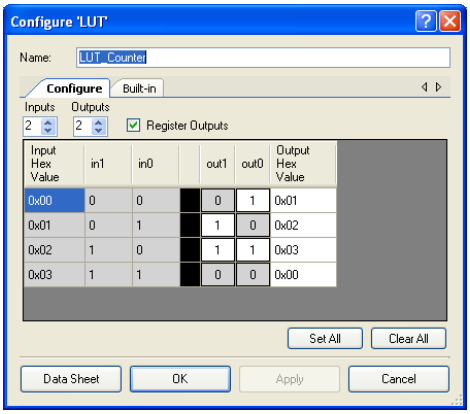
Figure 2. Multiplexed Comparator Design



Control Unit

This section consists of clock generation logic and a 2-bit counter implemented using a LUT component as shown in Figure 2. The output of the 2-bit counter is used to select the channel in the multiplexer and demultiplexer.

The following screenshot shows the LUT configuration for implementing the 2-bit counter.



LUT output increments for every clock that it receives.

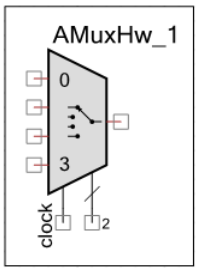
Clock generation logic consists of a PWM component. An input clock of 3 MHz to the PWM component also drives the clock terminal of the hardware AMUX component. Output PWM frequency sets the scanning speed of input channels. It is given by the following equation:

$$\frac{Scanning\ rate}{Channel} = \frac{PWM\ Output\ Frequency}{Number\ of\ channels}$$

In the current project, the PWM output frequency is set to 100 kHz and four channels are used. Therefore, scanning rate or channel becomes 25 kHz. Duty cycle setting gives configurable time for the DAC voltage and input channel voltage to settle on the internal analog bus. In the present project, duty cycle of output PWM is set to 50 percent.

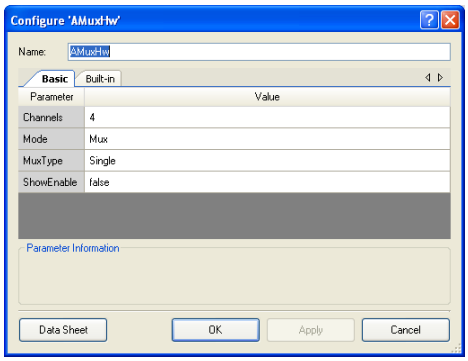
Multiplexer for Analog Signals

The analog hardware multiplexer (AMuxHw) component is used for the multiplexing function.



This makes use of Analog Global bus (AG) and Analog MUX (AMUX) bus switches, which are used for routing the analog signal in and out through the ports. These switches are controlled by digital signals. To synchronize the switching of channels with the other digital logic, a clock terminal is provided. A much higher clock, relatively, is selected for the multiplexer compared to channel scanning frequency because the multiplexer takes two clock cycles to change the channel. This allows the channel to be changed almost immediately as the select lines values change.

The following screenshot shows the AHwMux component configuration.



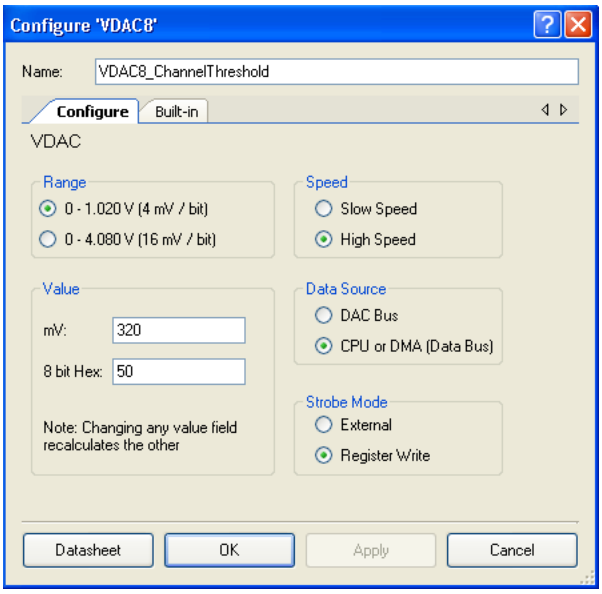
Four channels are configured. The mode of the component is set to MUX. This means that one channel can be connected to the output at a time. Setting it to switch will enable multiple input channels to be connected at a time. If input analog signals are single-ended (referenced to PSoC ground), select MuxType as single. For differential signals, select MuxType as differential. In this project, single-ended input signals are used.

For more details about the analog hardware multiplexer component, see the component datasheet.

Voltage DAC (VDAC)

Threshold voltage for the comparator is provided by an 8-bit D/A converter (VDAC) that is updated by DMA. DMA takes the threshold values from the memory set by the user. Because data resides in memory, threshold values can be changed in runtime.

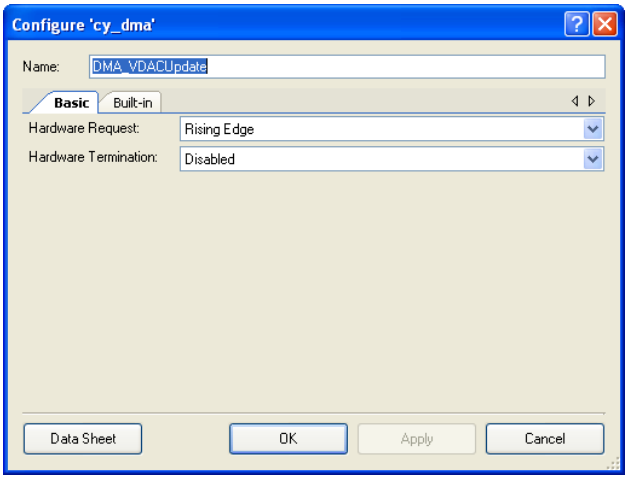
The following screenshot shows the VDAC configuration.



The VDAC output range is set to 1.020 V. Speed is set to Fast because VDAC update time affects the maximum scanning rate that can be achieved. This is explained later in this document. Because DMA will be configured to update DAC, Data Source is set to the CPU or DMA option.

DMA Configuration

DMA trigger terminal DRQ is enabled. A rising edge signal at this pin will initiate a data transfer. This terminal is driven by the PWM output clock. This is used to keep the VDAC voltage (threshold to the comparator) in sync with the channel selected.

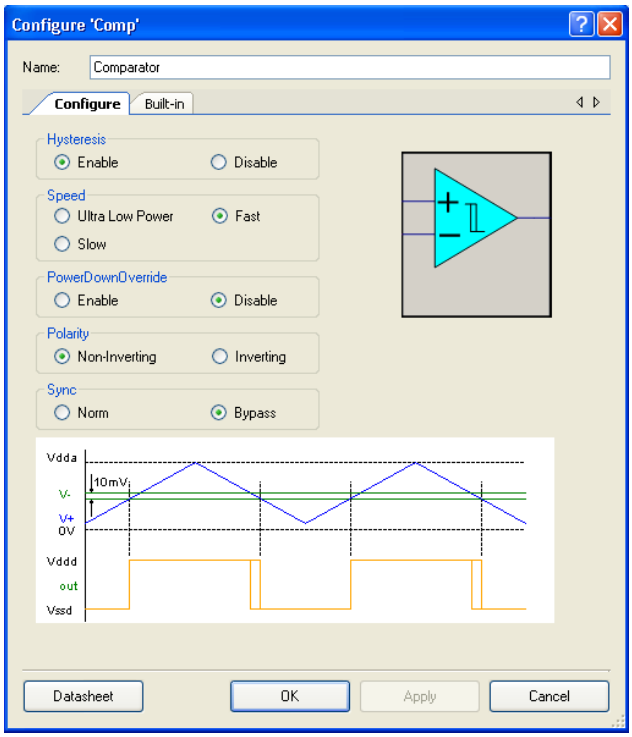


Single transaction descriptor (TD) is defined to carry out transactions or transfers from memory to VDAC. Because there are four channels set in this application, four 8-bit transactions are required. Four threshold values will be stored in memory. Single TD can perform this by using the auto source address increment feature where source address can be automatically incremented for the defined number of bytes. This requires the threshold values to be stored in continuous locations in memory. An array should be defined for threshold values which will ensure continuous memory allocation.

DRQ trigger signal and TD configuration enables the DMA to transfer 8-bit data from memory to VDAC on receiving each trigger. After every trigger, source address, which points to threshold values in memory, is incremented to keep it in sync with the selected channel. After transferring 4 bytes (starting from channel 0 through to channel 3), the source pointer returns back to channel 0. However, there will be a slight offset in synchronization. This is because, when DMA is triggered for the first time, the analog hardware multiplexer would have already selected channel 1 (second channel of the available four channels). In the firmware, the array declared for the threshold value should hold the channel 1 threshold value at '0' index, channel 2 threshold at '1' index, channel 3 threshold at '2' index, and channel 0 threshold at '3' index.

Comparator Configuration

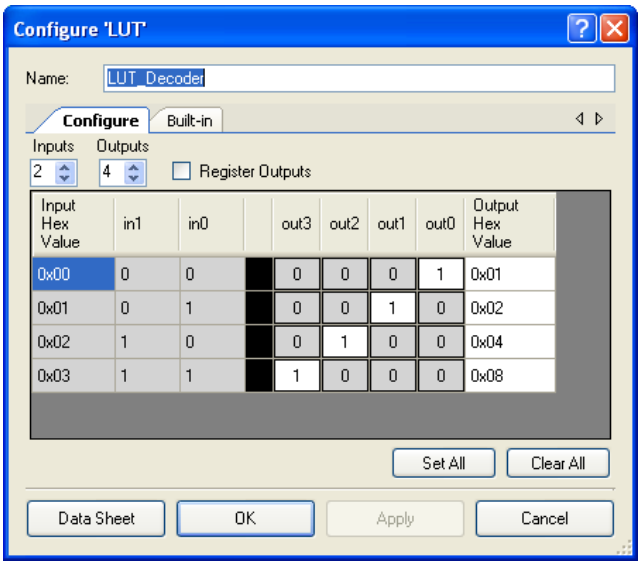
The following screenshot shows the comparator configuration.



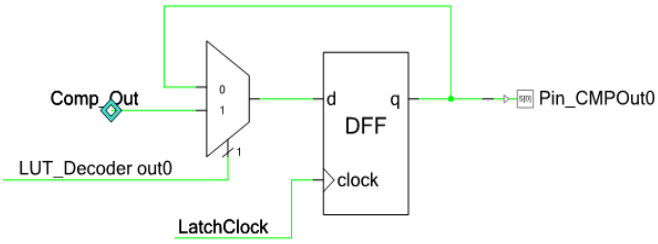
Hysteresis (approximately 10 mV) is enabled and speed is set to fast, which is essential because the scanning rate will be high.

Demultiplexer and Latch

LUT_decoder implements a 2:4 demultiplexer, which distribute the single comparator output to multiple channels in order to have individual outputs for each channel. The following screenshot shows the LUT configuration.



The output of this decoder is used to select a particular line for latching the comparator output. The following figure shows the logic for channel 0.



LatchClock is the inverted version of the PWM output. Because of this, comparator output gets latched at negative edge. This gives approximately ½ PWM cycle time for the signal to settle on the bus after connection. If LUT decoder output for a channel is 0, then the previous value is maintained at the output. When channel 0 is selected, LUT decoder output becomes 1 and comparator output is routed to input of latch. On negative edge, flip flop will store the comparator status. This logic is repeated for the remaining three channels.

Figure 3. Timing Diagram

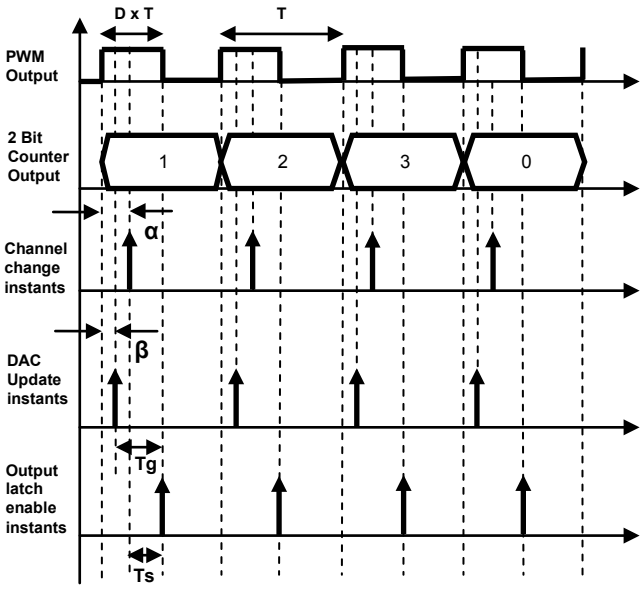


Figure 3 shows the alignment of various functions relative to each other. The 2-bit counter output increments for every clock cycle. Notice the channel change, DAC update, and output latch enable instants.

'**α**' is the delay caused in channel connections by using the AHwMux component. There is a delay of two clock (Clock_AMux) cycles in connecting the channel; one clock is used to disconnect the previous channel and another clock is used to connect the new channel. In the present project, Clock_AMux is set to 3 MHz. This means that **α** is equal to (2 / 3 MHz= 0.66 us).

Comparator output is latched at the negative edge of the clock when the channel is selected. Time (Ts) given for the signal to settle down on the bus can be calculated as follows:

Ts = D x T - α

D is the duty cycle and T is the time period of PWM output. Note that if the source that is driving the channel has higher impedance, more time should be given for the signal to settle down. There are two options available to increase the allowed time for settling. You can increase the PWM duty cycle D; this will delay the negative edge at which comparator output is sampled by the output latch. In this case, the scanning rate will remain same. Another option is to reduce the PWM frequency; this will reduce the scanning rate. Select one of these options depending on the source impedance and scanning rate requirement.

'**β**' is the delay caused in VDAC update by DMA.

In the present application in which one DMA channel is used to transfer 8-bit data from memory to VDAC, it takes approximately 6 bus cycles to update the DAC.

Therefore, **β** = (6 / Bus Clock Frequency). In the project, Bus Clock Frequency is set to 24 MHz; β is approximately 0.25 us. DMA latency depends on various factors. For more information, see the Technical Reference Manual.

If the PWM output frequency that sets the scanning rate is less than 100 kHz, then β value will be small for most used cases as compared to period of T and thus can be neglected.

The time gap between the VDAC update and comparator output latch is:

Tg = D x T - β

VDAC should settle to the set value within Tg time. The maximum update of VDAC for the 1 V range is 1 Msps. Therefore, Tg should always be greater than 1 µs.

There are two options to ensure this:

- Increase duty cycle (D). This will push the negative edge at which comparator output is sampled thus widening time gap.
- Decrease PWM output frequency (T will increase)

In the present application, α = 0.66 us, β~0.25 us, D = 0.5.

VDAC range is set to 1 V.

With this data, the maximum PWM frequency which decides the maximum scanning rate can be calculated as:

Tg= D x T - β

Tg will be minimum for maximum PWM frequency and it is equal to max VDAC settling time; 1 us in this case.

So, 1 us = 0.5 x T – 0.25 us; therefore, T = 2.5 us.

Maximum PWM frequency is therefore 1/T = 400 kHz.

The following table summarizes the numbers for different VDAC settings. (Duty set to 50 percent, β = 0.25 us)

VDAC range	Maximum Update rate	Maximum PWM frequency	Maximum scanning rate / channel (number of channels = 4)
1 V	1 Msps	400 kHz	100 kHz
4 V	250 ksps	118 kHz	29.5 kHz

If the scanning rate is kept small, then β can be neglected.

Pin Selection

The following screenshot shows the pins selected in the application.

Figure 4. Pin Selection

Name	Pin
Pin_CH0	P4[0]
Pin_CH1	P4[1]
Pin_CH2	P4[2]
Pin_CH3	P4[3]
Pin_ClockOut	P1[7]
Pin_CMPOut0	P5[0]
Pin_CMPOut1	P5[1]
Pin_CMPOut2	P5[2]
Pin_CMPOut3	P5[3]
Pin_DACout	P3[0]

Testing

Test the project by applying a triangular waveform signal at the channel and observing the output.

Oscilloscope CH1- Threshold Levels

Oscilloscope CH3- Output

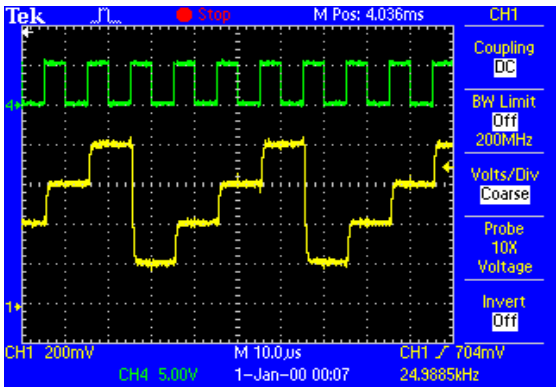
Oscilloscope CH2- Input Signal

Oscilloscope CH4- PWM output

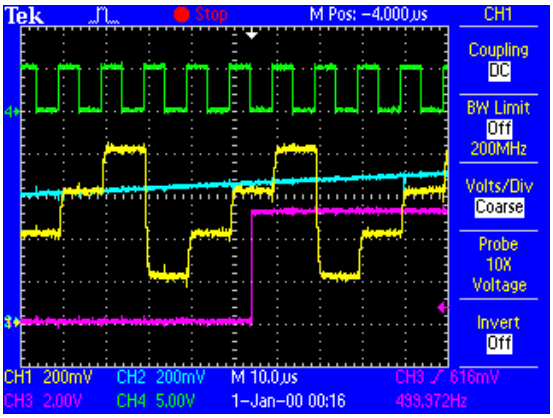
VDAC threshold levels that are currently set in the project are shown here.

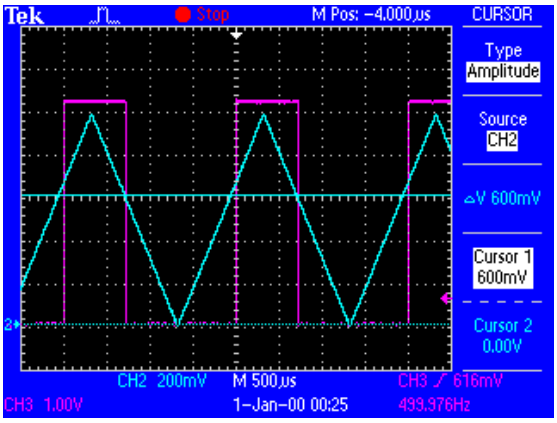
Channel	VDAC counts	VDAC voltage
CH0	200	800 mV
CH1	50	200 mV
CH2	100	400 mV
CH3	150	600 mV

DAC threshold stays at one particular value for one PWM output cycle. It changes at the positive edge of the clock as shown in the following DSO screenshot.



Provide a ramp signal of 1 V p-p and 500 Hz at CH3 input P4[3], and observe the output at P5[3]. As in the following two screenshots, the output goes high at the negative edge of the PWM when the input signal exceeds the CH0 threshold of 600 mV. You can repeat this for other channels as well.





There is slight offset in time when the output switches. This is because of finite scanning speed and comparator hysteresis.

Available Options

- **Checking the status of comparator using CPU.** If you need to check the comparator status for each channel using CPU, then a status register component can be connected at the output of the latch. The status register can then be read by the CPU to check the status for each channel.
- **Providing interrupt when the signal crosses threshold level.** The interrupt component can be connected at the output of the latch.
- **Increasing or decreasing the number of channels.** The counter used in control unit should be such that:
  - Modulus of the counter = number of channels.
  - LUT should be designed accordingly.
  - The number of D F/Fs in the output latch and analog hardware multiplexer dimension changes.
- **Changing the scanning rate.** Changing the PWM output frequency, which drives the counter, changes the scanning rate.

Summary

This application note demonstrated multiplexed comparator implementation. The design is structured in such a way that you can easily modify the design for your application, depending on how many channels needed, scanning speed, and other variables.

About the Author

Name:	Rajiv Vasanth Badiger
Title:	Sr. Applications Engineer
Background:	B.E. Electronics and Communication
Contact:	rjvb@cypress.com

Document History

Document Title: PSoC® 3 / PSoC 5LP Multiplexed Comparator – AN60220

Document Number: 001-60220

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	2893274	RJVB	03/16/2010	New application note
*A	3194626	RJVB	03/12/2011	Updated the design with analog hardware multiplexer Updated timing diagram and maximum scanning rate calculations
*B	3442273	RJVB	11/30/2011	Updated template and project files Component customizer screenshots updated.
*C	3556272	NIDH	03/20/2012	Added contact email ID.
*D	3809219	NIDH	11/12/2012	Updated for PSoC 5LP

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc cypress.com/go/plc
Memory	cypress.com/go/memory
Optical Navigation Sensors	cypress.com/go/ons
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/usb
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 5

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com