# Analog - Analog Multiplication with PSoC

# AN2159

**Author**: Victor Kremin
**Associated Project**: Yes
**Associated Part Family**: CY8C24xxx, CY8C27xxx
**Software Version**: PSoC® Designer™ 5.1
**Associated Application Notes**: AN2041, AN2044, AN2161

## Application Note Abstract

This Application Note demonstrates how to design a four-quadrant multiplier, which is capable of multiplying two analog signals. The multiplier does not require any CPU utilization during operation.

## Introduction

Analog multipliers are used in a wide range of applications. Possible examples in which multipliers are present include:

- Power meters. Multipliers are used to form the active and reactive power level signals.
- Phase detectors, for example, PLL systems or impedance meters.
- Instrumentation systems that measure complex signal RMS values, such as high-end digital multi meters.
- Modulation and demodulation functions in various systems. Note the modulation technique allows the reduction of influence of various offset voltages, 1/f noise in the range of instrumental systems, and detection of very low-level signals.
- Frequency conversion. Multipliers can be used as mixers in super-heterodyne receivers. A simple, wide-range harmonic signal frequency doubler can be built using the multiplier.
- Voltage-controlled variable gain amplifiers and filters.
- Non-linear automatic control systems.

PSoC® has no non-linear hardware (Gilbert or log/ expression cores) dedicated directly to performing multiplication of two analog signals. PSoC does have analog modulators capable of multiplying one analog signal by another binary signal to which only two values, ±1, can be assigned.

When both signals are analog, it is not suitable for direct multiplier implementation.

One can also build the multiplier using a double ADC and single DAC, but this approach requires CPU activity to read the ADC, multiply and normalize the conversion results, and write the final product to the DAC. This technique is not very useful in many practical applications because the CPU is busy with other tasks, and the DAC updating jitter/latency degrades the multiplier performance. Therefore, it is better to build the multiplier purely in the hardware, requiring no CPU activity for operation.

For a general explanation of switched capacitors, please read AN2041 “Understanding Switched Capacitor Blocks.” For a discussion on analog modulators, please reference AN2044 “Signal Rectification, using Switched Capacitor Modulators.”

## The Multiplier Idea

The multiplier idea lies in the conversion of one analog signal in the pulse-width-modulated (PWM) signal and multiplication of the second analog signal by the pulse-width-modulated signal. The resulting filtered signal yields a signal that is proportional to the product of two signals.

Figure 1 shows the multiplier flowchart

Figure 1. Converter Block Diagram

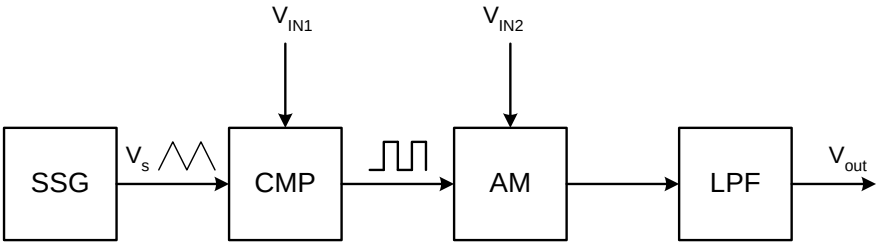
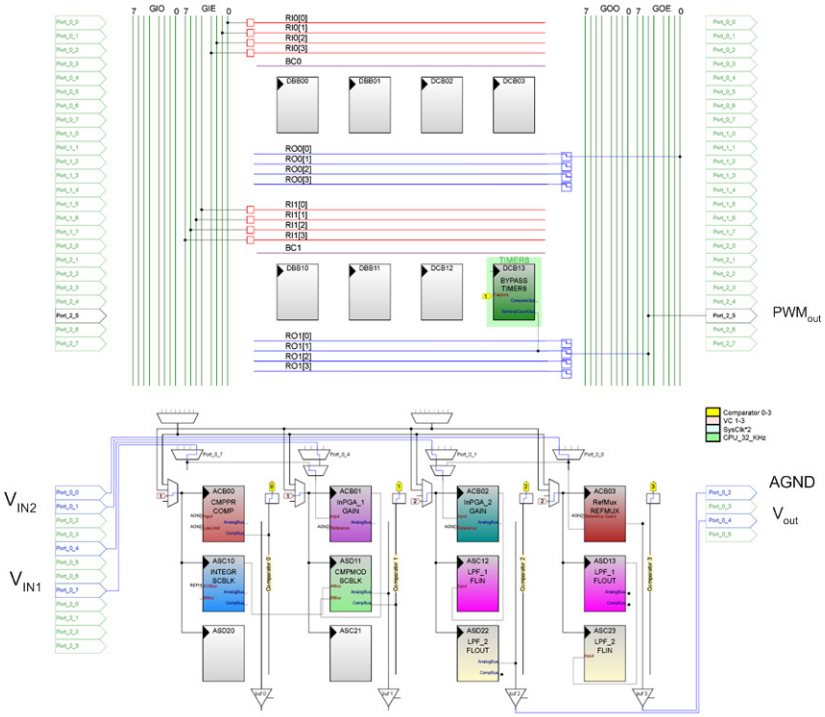



Figure 2. Multiplier Internals




The sawtooth signal generator (SSG) forms a symmetric sawtooth signal, which is sent to the first comparator (CMP) input. The first multiplier input signal is sent to another comparator input. The comparator works as an analog PWM, converting the analog signal into a duty-cycle-modulated signal. The comparator output signal drives the reference input of the analog modulator (AM), which multiplies the second signal by the PWM signal. The low-pass filter (LPF) removes the high-frequency SSG signal components from the AM spectrum and selects a relatively low-frequency product signal.

If the upper limit of the multiplier input signal's frequency spectrum is much less than the sawtooth generator output signal, the input signals are virtually unchanged during the sawtooth signal period.

The AM multiplies the input signal by +1 when the first multiplier input, $V_{IN1}$, is greater than the SSG $V_s$ signal, and by -1 when $V_{IN1}$ is less than the SSG $V_s$ signal. Because the sawtooth signal changes linearly with time and is symmetric relative to AGND, the +1 multiplication is directly proportional to $V_{IN1}$. As a result, the AM output signal average produces a signal that is proportional to the product of the two input signals. Note that the analog PWM produces a 50% duty-cycle output for zero input signals. The pulse width increasing/decreasing for positive/negative inputs allows for generation of a four-quadrant multiplier (all voltages are relative to AGND).

## Multiplier Implementation

The current implementation of the PSoC multiplier is depicted in Figure 2.

The sawtooth signal generator has been built using the switched capacitor integrator and Schmitt trigger. Generator operation is described in detail in AN2161 "The Voltage-to-Frequency Converter." The Schmitt trigger is placed in the ACB00, the integrator is placed in the

ASC10, and the analog modulator is used to change the integration sign. The input voltage is set REFHI with RefMux set to Vdd/2 ± Vdd/2. Those, with preset integrator settings, output the generator frequency at about 15 kHz.

The first input signal, $V_{IN1}$, comes to the first comparator input built around the configurable switched capacitor block in the ASD11 via InPGA_1. The second comparator input signal, $V_{IN2}$, comes from the generator integrator. The comparator output signal is passed to the ASC12 analog modulator via *Comparator_Bus_1*. To route this signal to an external pin for test/debug purposes, the Timer_8 is placed in the DCB13, module function is changed to the CRC, and bypass mode is turned on in the firmware code.

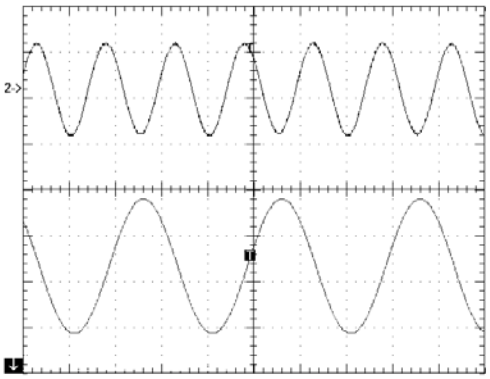
The second signal passes to the ASC12 analog modulator via InPGA_2. The modulator output signal is filtered by a 4-order LPF placed in ASC12, ASD13, ASD22, and ASC23. The filter roll-off frequency is set near 4 kHz. The reference multiplexer (RefMux) signal is passed to port pin P0[2] via column buffer 3 to provide DC bias voltage for multiplier inputs.

## Multiplier Tests

Figure 3 shows the result of using the proposed multiplier as a harmonic, signal-frequency doubler. The 1 kHz signal is sent to both inputs at the same time. The output is a 2 kHz sinusoidal signal. See Equation (1):

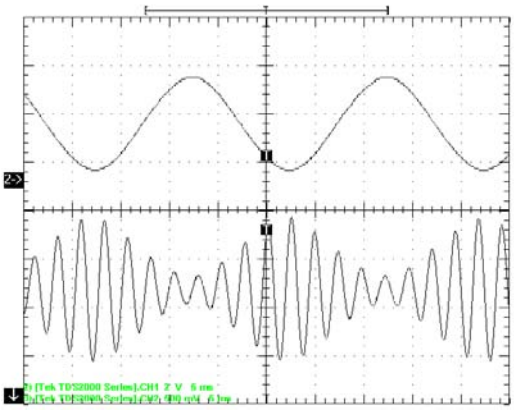
$$\cos^2(\omega t) = \frac{1}{2}\left(1 + \cos(2\omega t)\right) \qquad \text{Equation 1}$$

Figure 3. Using the Multiplier as a Frequency Doubler



In another example (Figure 4), the multiplier is used as an amplitude modulator, performing amplitude modulation of 500 Hz carrier signal. The modulation frequency is 50 Hz.

Figure 4. Using the Multiplier as an Amplitude Modulator



With the existing settings, the multiplier operation frequency is limited to 2.5-3 kHz. It is possible to increase the SSG operation frequency and LPF roll-off frequency to get a wider frequency range. The multiplier's linearity is better than 0.1% for DC signals on $V_{IN2}$. $V_{IN1}$ input accuracy is not as good due to a number of factors. The sawtooth signal echelon form limits the $V_{IN2}$ resolution because the different input voltages within the integrator output step are translated in the same way as the pulse width. With the preset integrator settings, the rising or falling sawtooth edge takes about 50 steps or column sample-signal periods. As a result, these 50 integration steps translate to the 2% error component caused by the discrete switched-capacitor integrator operation.

The multiplier resolution can be improved by using a continuous-type integrator (built around a continuous-type block and external R/C components) to form the sawtooth signal. The integrator leakage current limits the sawtooth signal linearity, which translates to multiplier non-linearity. This error component is near 1.5% full scale.

## Design Variations

This design has been implemented using the PSoC CY8C27xxx device family. The CY8C24xxx family can be used to build a multiplier as well because several possible implementations exist. For example, Column_0 can be used to build the sawtooth generator and Column_1 can be used to place the analog modulator and second-order vertical LPF. The input signal comes directly from port 2. The modulation comparator can be built using a second continuous-type block.

## Alternative Applications

The proposed multiplier has an analog PWM, which has numerous standalone applications, including PWM-controlled DC-DC converters and analog-digital control loops (valve, motors controlling). For example, the addition of the PID hardware implements a closed-loop control with PWM output. This system operates purely in the hardware; CPU is required only for start-up and possible control-loop-parameter optimization at runtime.

## Appendix

The multiplier main routine source code:

```
void main()
{
   CMPPR_COMP_CR0 |= BIT(2); //connect the RTopMux to opam output
   CMPPR_COMP_CR1 = (CMPPR_COMP_CR1 & 0xC0) | 0x2F; //set NMux and PMux connections
   CMPPR_COMP_CR2 &= BIT(6); //The output latch is always transparent
   CMPPR_Start(CMPPR_HIGHPOWER);

   AMD_CR0 |= 0x04 | 0x50;
   INTEGR_Start(INTEGR_HIGHPOWER);


   CMPMOD_Start(CMPMOD_HIGHPOWER);

   InPGA_1_Start(3);
   InPGA_2_Start(3);

   LPF_1_Start(3);
   LPF_2_Start(3);

   RefMux_Start(RefMux_HIGHPOWER);

   BYPASS_FUNC_REG = BYPASS_FUNC_REG & 0xFC | 0x02;
   BYPASS_CONTROL_REG |= 0x03;
   BYPASS_Start();

   while(1);
}
```

## About the Author

**Name**: Victor Kremin
**Title**: Associate Professor
**Background**: Victor earned a radiophysics diploma in 1996 from Ivan Franko National Lviv University, a PhD degree in Computer Aided Design systems in 2000, and is presently working as Associate Professor at National University "Lvivska Polytechnika" (Lviv, Ukraine). His interests involve the full cycle of embedded systems design including various processors, operating systems and target applications.

**Contact**: vkremin@lviv.farlep.net

## Document History

**Document Title: Analog - Analog Multiplication with PSoC**

**Document Number: 001-35342**

| Revision | ECN | Orig. of Change | Submission Date | Description of Change |
|---|---|---|---|---|
| ** | 1513664 | MEH | 09/26/07 | New Spec. |
| *A | 3087707 | MEH | 11/16/10 | The project was converted from PSoC Designer 4.1 to PSoC Designer 5.1. |

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.



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