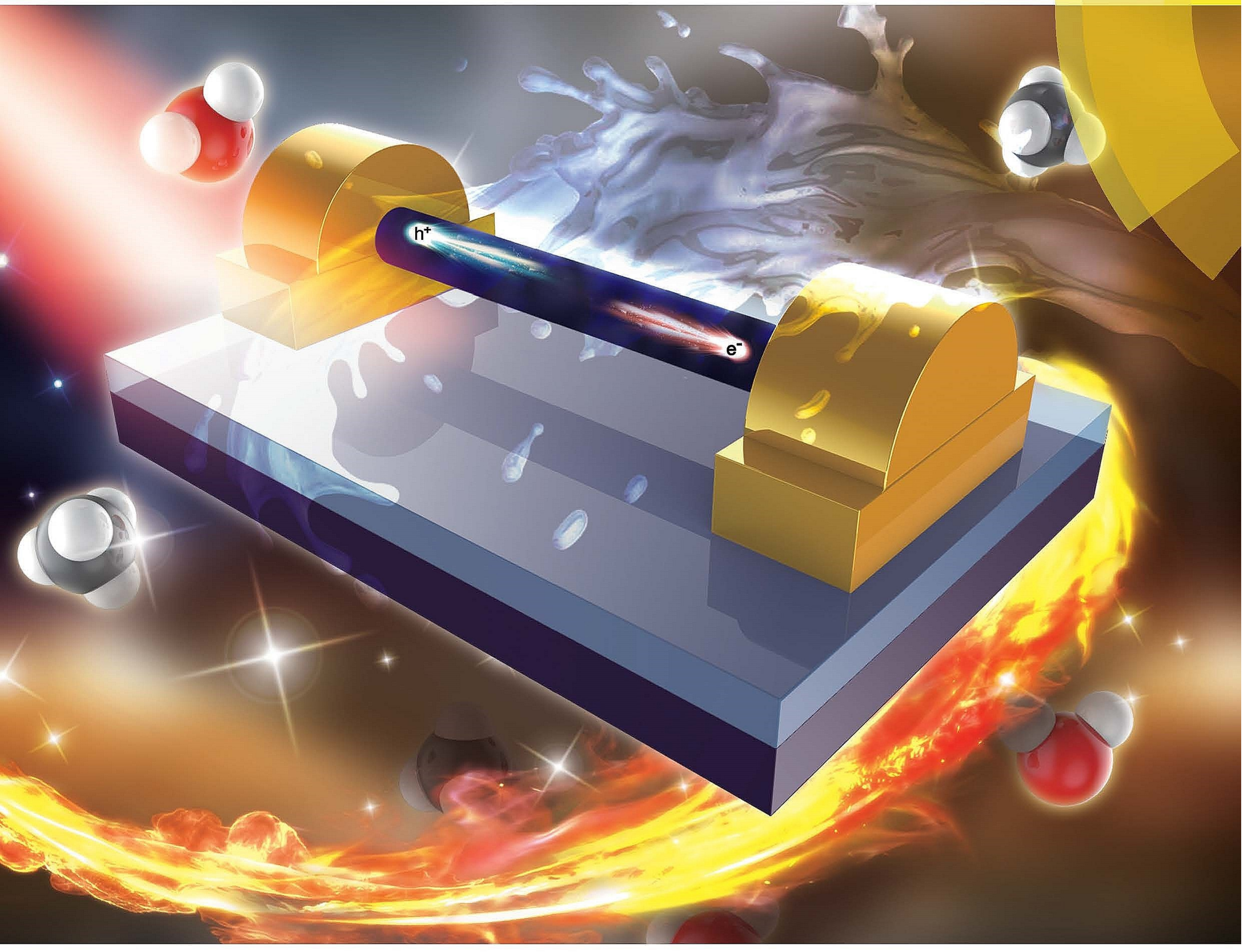
JB Institute of Engineering & Technology

TECHTRONICS



Electronics & Communication Engineering

Technical Magazine

May 2019

Technical Magazine

2018-2019

Editors:

Dr. Towheed Sultana
Dr. Anindya Jana
Dr. Prasanta Kumar Pradhan

Best Complements

from

The Dept. of

ECE

Massage from the Desk of Principal

I am very much pleased to see the keen interest and devotion of the students of ECE, JBIET towards the publication of the Department technical magazine. This is an indicator of the communication skills of the students, which is rising day-by-day. And the way students are putting across this massage to the reader shows that there communication skills are raising. TECHTRONICS is a magazine that provides you with everything you need to know.

Our college is one, that strives for excellence in every field, with academics, sports, cultural or co-curricular activities. It is true that your academic performance matters the most you go for an interview, but when your furnish details of how much you have participated in co curricular activities, apart from your academics may impress the interviews. So build your communication skills and vocabulary by participating co curricular activities.

Here, wishing all the happy reading.

Dr. Towheed Sultana
Principal, JBIET

Massage from the Desk of HOD

I am very glad to see my students dedicated for the issue of the TECHTRONICS magazine. Through this edition, I want my students to know about the present scenario. As it is month of exam pressure, I am feeling proud that the students discussed about how to overcome examination fever and to prepare for the exams.

I am feeling glad that my students have taken social responsibilities throughout the year.

Finally, I convey my wishes to all the students for their upcoming examinations.

Dr. S. Ibrahim Sadhar
HOD, ECE

Stress Management

Stress can come from many sources, which are known as "stressors." Because our experience of what is considered "stressful" is created by our unique perceptions of what we encounter in life (based on our own mix of personality traits, available resources, habitual thought patterns), a situation may be perceived as "stressful" by one person and merely "challenging" by someone else.

Simply put, one person's stress trigger may not register as stressful to someone else. That said, certain situations tend to cause more stress in most people and can increase the risk of burnout. For example, when we find ourselves in situations where there are high demands on us; where we have little control and few choices; where we don't feel equipped; where we may be harshly judged by others; and where consequences for failure are steep or unpredictable, we tend to get stressed.

Because of this, many people are stressed by their jobs, their relationships, their financial issues, health problems, and more mundane things like clutter or busy schedules. Learning skills to cope with these stressors can help reduce your experience of stress.

Stress can be effectively managed in many different ways. The best stress management plans usually include a mix of stress relievers that address stress physically and psychologically and help to develop resilience and coping skills.

The following are some effective stress management techniques, we should consider making us stress free.

Use quick stress relievers

Some stress relief techniques can work in just a few minutes to calm the body's stress response. These techniques offer a "quick fix" that helps you feel calmer at the moment, and this can help in several ways. When your stress response is not triggered, you may approach problems more thoughtfully and proactively. You may be less likely to lash out at others out of frustration, which can keep your relationships

healthier. Nipping your stress response in the bud can also keep you from experiencing chronic stress.
Quick stress relievers like breathing exercises, for example, may not build your resilience to future stress or minimize the stressors that you face, but they can help calm the body's physiology once the stress response is triggered.

Develop stress-relieving habits

Some techniques are less convenient to use when you are in the middle of a stressful situation. But if you practice them regularly, they can help you manage stress in general by being less reactive to it and more able to reverse your stress response quickly and easily.
Long-term healthy habits, like exercise or regular meditation, can help to promote resilience toward stressors if you make them a regular part of your life.3

Communication skills and other lifestyle skills can be helpful in managing stressors and changing how we feel from "overwhelmed" to "challenged" or even "stimulated."

Eliminate stressors when you can

You may not be able to completely eliminate stress from your life or even the biggest stressors, but there are areas where you can minimize it and get it to a manageable level. Any stress that you can cut out can minimize your overall stress load. For example, ending even one toxic relationship can help you more effectively deal with other stress you experience because you may feel less overwhelmed.

Dr. V.V. Rao
CEO, JB Group

Analog and Digital Implementation of a Synchronous Demodulator

P PREETHI
ROLL NO:16671A0470

ABSTRACT

In this article, we'll take a look at the analog blocks for implementing the square wave-based synchronous demodulator. We'll also briefly look at the FPGA implementation of the synchronous demodulation technique.

SQUARE WAVE-BASED SYNCHRONOUS DEMODULATOR IDEA IN DETAILS

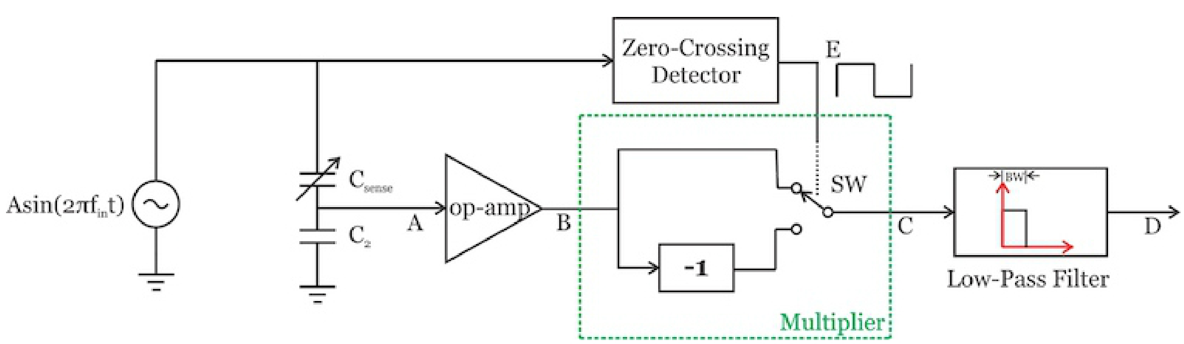


Figure 1. Square wave-based synchronous demodulato

The two blocks that we'll examine are the "zero-crossing detector" and the "multiplier."

ZERO-CROSSING DETECTOR

The “zero-crossing detector” converts the input sine wave into a square wave that drives the switch SW. This can be done using the circuit in Figure 2.

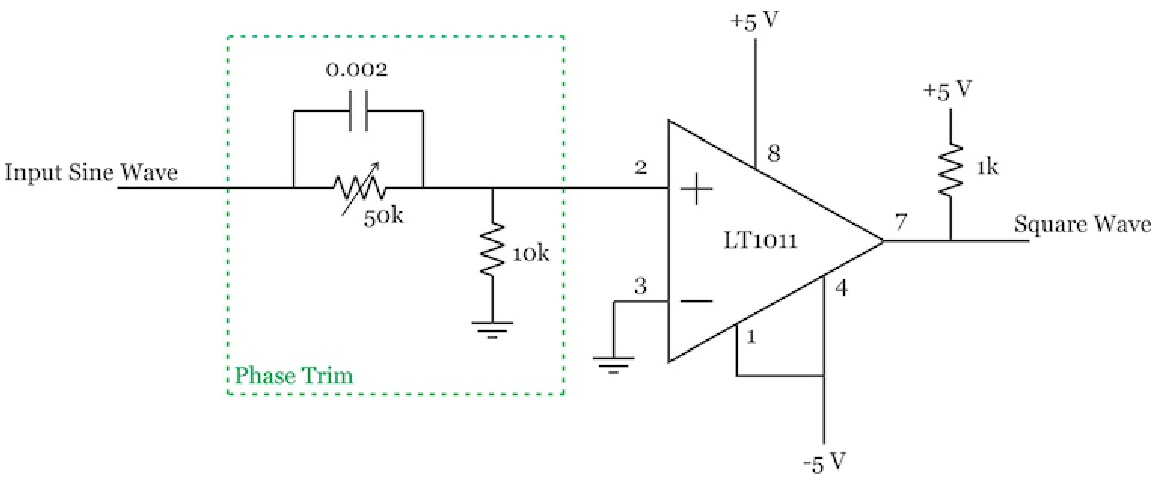


Figure 2. The zero-crossing detector block in a synchronous demodulator. Schematic used courtesy of Linear Technology

The LT1011 is a voltage comparator that compares the input sine wave with the ground level. The potentiometer is used to adjust the phase of the produced square wave so that it matches the phase of the sine wave at node B in Figure 1.

In this way, we can have a square wave that switches when the sine wave crosses 0 V. Recall that the signal amplitude at the output of the multiplier is a function of the phase difference between the two inputs of the multiplier. When the square wave is in phase with the sine wave, the phase relationship between the two signals is known and we can more easily interpret the voltage that appears at the output of the low-pass filter.

MULTIPLIER

One common implementation for the "Multiplier" block is shown in Figure 3:

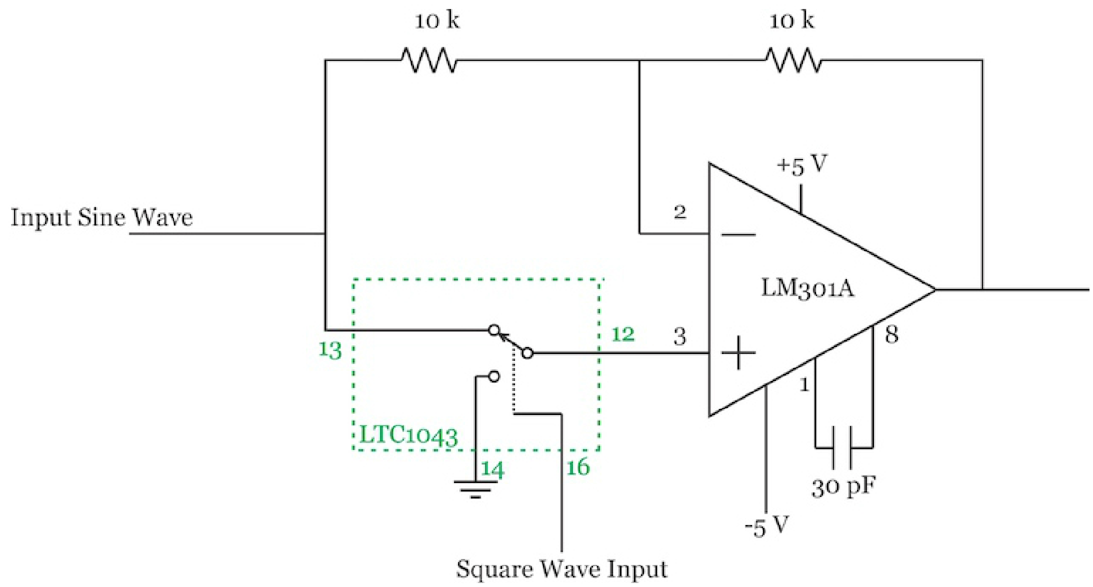


Figure 3. The multiplier block of the example synchronous demodulator.
Schematic used courtesy of Linear Technology

In this figure, the LM301A is a general-purpose operational amplifier. The LTC1043 is a building block originally designed for implementing discrete switched-capacitor circuits. Figure 4 shows a simplified block diagram of a portion of the circuitry inside the LTC1043.

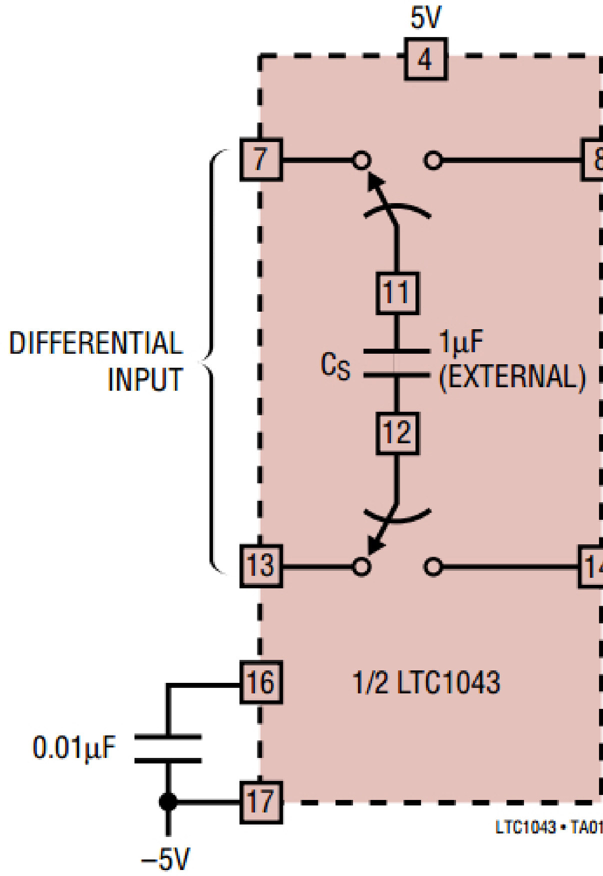


Figure 4. Diagram of the LTC1043. Image courtesy of Linear Technology

As shown in this figure, the differential input is connected to an external capacitor during the sampling phase. During the next phase, the charged capacitor is connected to the output port. The clock for the switches can be created either internally or through an external CMOS clock.

This apparently simple functionality allows us to use the LTC1043 in a variety of applications such as precision instrumentation amplifiers and switched-capacitor filters. However, with the schematic shown in Figure 3, the LTC1043 is actually used as a simple switch.

Let’s see how this circuit can multiply the input by a square wave. When the switch connects terminals 12 and 14 together, we have the following schematic.

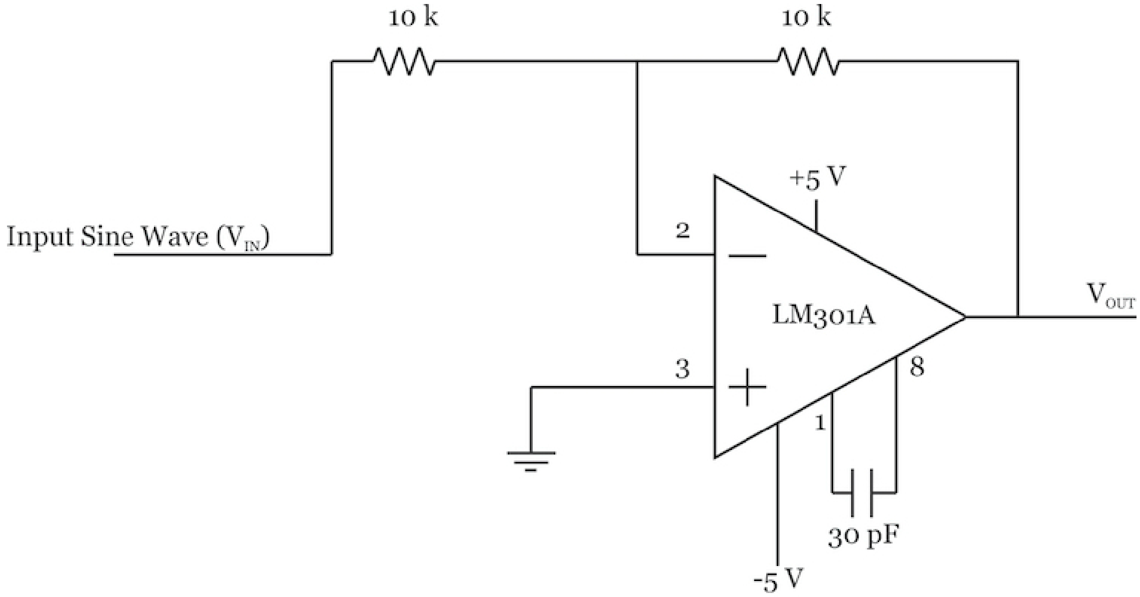


Figure 5

In this phase of operation, we have an inverting amplifier with a gain of VOUTVIN=−1VOUTVIN=−1. However, when terminals 12 and 13 of the LTC1043 are connected together, we obtain the following schematic:

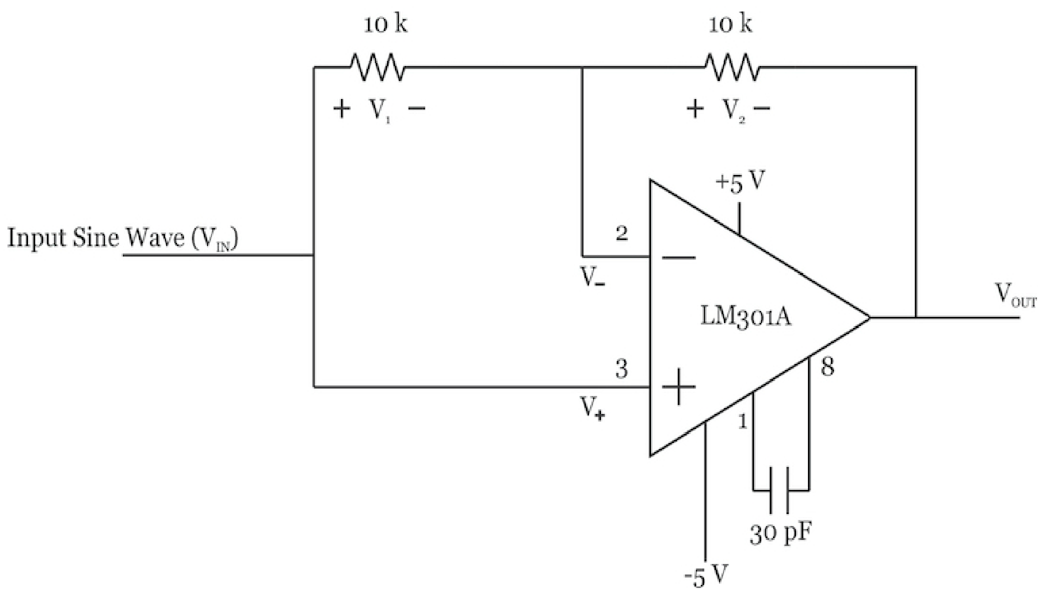


Figure 6. Connecting terminals 12 and 13 of the LTC1043

We know that due to the negative feedback path and the high gain of the op-amp, the two inputs of the op-amp have almost the same voltage: V−=V+V−=V+. Therefore, V1=0VV1=0V and no current flows through the 10 kΩ resistor on the left.

Assuming that the current drawn by the op-amp inputs is negligible, the current through the 10 kΩ resistor in the feedback path will be zero as well and we have V2=0VV2=0V. Therefore, in this phase of operation, we have a gain of VOUTVIN=+1VOUTVIN=+1. In other words, the input is multiplied by a square wave that toggles between ±1.

DIGITAL IMPLEMENTATION OF SYNCHRONOUS DEMODULATION

Rather than using analog building blocks, we can use digital circuits to implement a synchronous demodulator. The basic idea is shown in Figure 7.

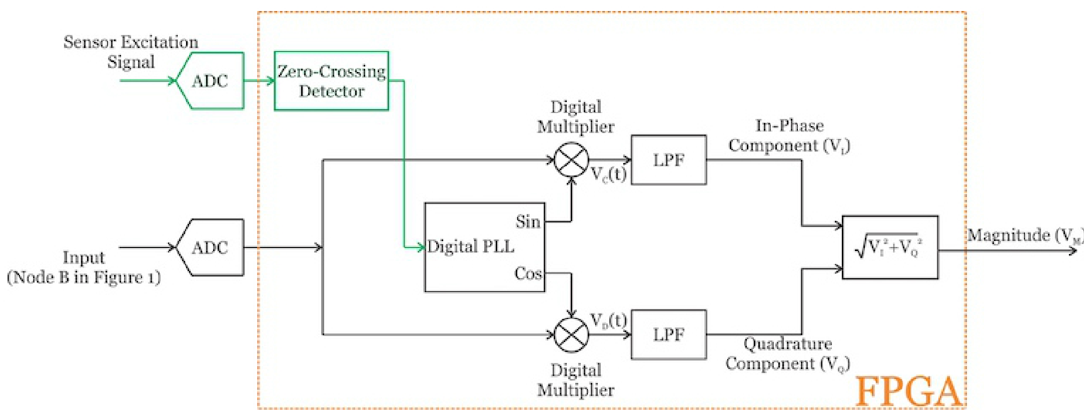


Figure 7. A digital synchronous demodulator circuit

Two A/D converters (ADCs) are used to digitize the input signal (node B in Figure 1) and the sensor excitation sine wave. As shown in the figure, the other blocks are digital and can be implemented by an FPGA.

IN-PHASE AND QUADRATURE COMPONENTS

In Figure 7, the digitized input is multiplied by sine and cosine waves produced by a digital phase-locked loop (PLL). Why do we need to multiply the input by both the sine and cosine waves? In the first part of this series, we examined multiplication by a sine wave. If we multiply

vB(t)=Bsin(2πfint+ϕ)vB(t)=Bsin(2πfint+ϕ) by Asin(2πfint)Asin(2πfint), we obtain:

vC(t)=Asin(2πfint)×Bsin(2πfint+ϕ)=12ABcos(ϕ)−12ABcos(4πfint+ϕ)vC(t)=Asin(2πfint)×Bsin(2πfint+ϕ)=12ABcos(ϕ)−12ABcos(4πfint+ϕ)

The first term is DC, however, the second term is at twice the input frequency. Hence, a narrow low-pass filter (LPF) can remove the second term and we have:

vI=12ABcos(ϕ)vI=12ABcos(ϕ)

The output is a function of the phase difference between the two inputs. This equation shows that a phase difference between the measured signal and the reference input can reduce the signal amplitude at the output of the LPF.

Additionally, we need to know the phase difference to be able to calculate the amplitude of the measured signal. (Remember that the analog solution discussed above used the "phase trim" circuitry to make the phase difference equal to 0). To circumvent these two problems, we incorporate a second multiplication stage that multiplies the input by a cosine wave:

vD(t)=Asin(2πfint)×Bsin(2πfint+ϕ)=12ABsin(ϕ)+12ABsin(4πfint+ϕ)vD(t)=Asin(2πfint)×Bsin(2πfint+ϕ)=12ABsin(ϕ)+12ABsin(4πfint+ϕ)

After the LPF, we have:

$$vQ=12AB\sin(\phi)vQ=12AB\sin(\phi)$$

We can calculate the magnitude of the input by taking the square root of the sum of the squares of the in-phase and quadrature components:

$$vM=\sqrt{}vI2+vQ2=12ABvM=vI2+vQ2=12AB$$

As you can see, the result is not a function of the phase difference.

DIGITAL PLL

It is important to note that the digital PLL in Figure 7 must generate samples of sine and cosine waveforms. We can accomplish this by using a PLL that has a direct digital synthesizer (DDS) as its digitally controlled oscillator (DCO). The basic idea is shown in Figure 8.

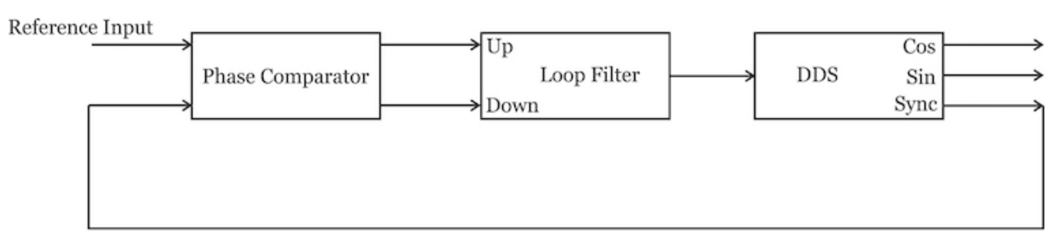


Figure 8. A PLL with a direct digital synthesizer. Image adapted from Cornell University

You can find more information about FPGA implementation of a synchronous demodulator here.

CONCLUSION

In this article, we examined both the analog and digital implementation of the synchronous demodulation technique:

For the analog implementation, we need a “zero-crossing detector” and a “multiplier”.

For the digital version, we can use two ADCs to digitize the measured signal and the sensor excitation waveform.

The other blocks can be implemented in an FPGA. By implementing both the in-phase and the quadrature paths, we can make the measurement independent of the phase difference between the multiplier inputs. For the digital implementation, we need a PLL that generates samples of sine and cosine waveforms. This can be achieved by a PLL that uses a DDS as a programmable oscillator.

Real Time Clocks (RTCs) in Microcontroller Timers

B RAJA RAJESHWARI
15671A0467

Real-time clocks (RTC) are timers dedicated to maintaining a one-second timebase. In addition, an RTC is often used to keep track of clock time and calendar date either in software or hardware. Many of the features of an RTC are very specialized and required for maintaining high accuracy and very reliable operation. There are RTC devices external to a microcontroller which interface with an I2C or SPI bus.

This article describes RTCs inside a microcontroller.

OVERVIEW OF REAL-TIME CLOCKS

The real-time clock's basic function is to produce intervals of one second and maintain a continuous count.

You can see a representation of this in the diagram below.

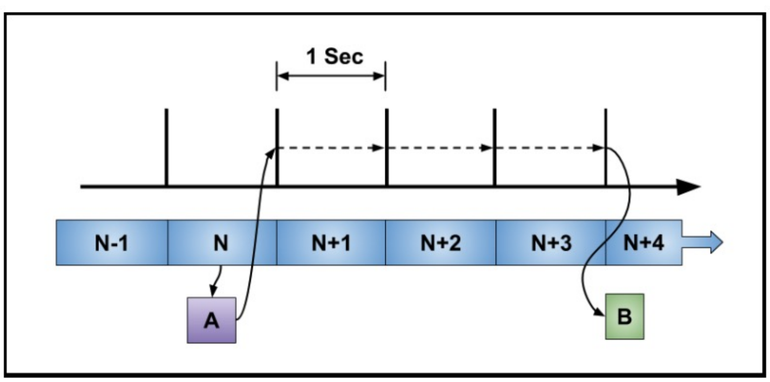


Figure 1. This timing diagram depicts the basic function of an RTC

Also shown is a program function, A, reading a seconds counter and scheduling an event, B, to occur three seconds in the future. This action is called an alarm. Notice the seconds counter runs continuously and does not stop and start. Two primary requirements of an RTC are accuracy and continuous operation.

The next diagram shows common hardware features of an RTC.

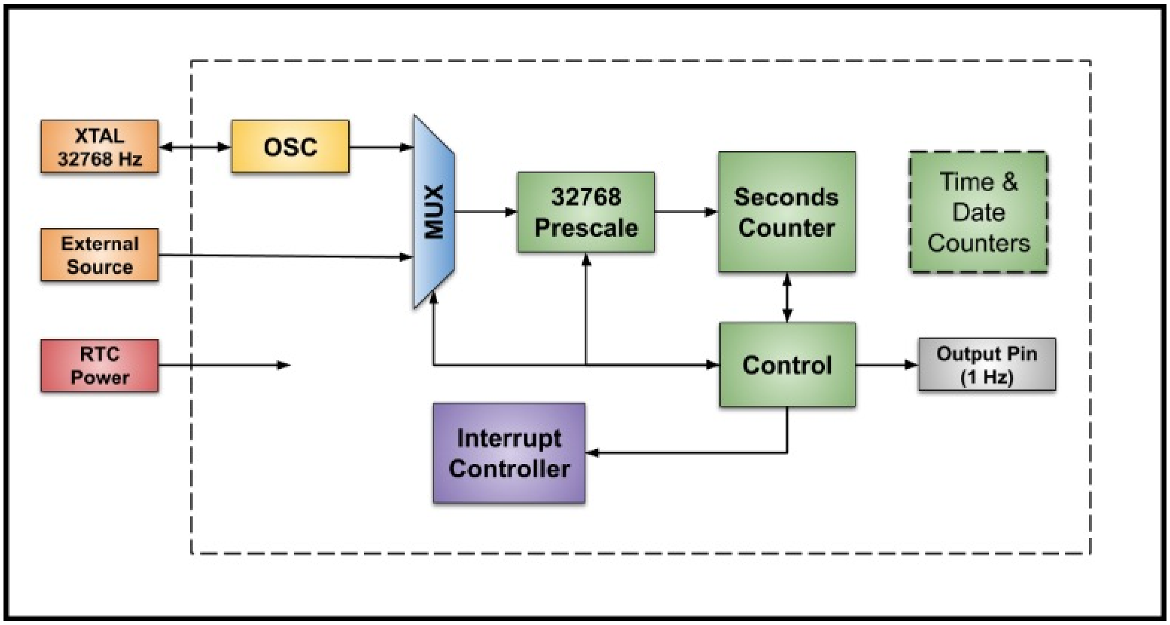


Figure 2. Real-time clock hardware features

An RTC often has its own internal oscillator with an external crystal and an option to use an external frequency reference. All clock sources run at 32,768 Hz. An external clock source allows the use of a very accurate and stable device such as a TCXO (temperature-compensated crystal oscillator).

A clock source is selected with a multiplexer and input to a prescaler which divides the clock by a factor of 32,768 (215) to produce a one-second clock.

A basic RTC has a seconds counter which is usually 32 bits or more. Some RTCs have specialized counters to keep track of the time of day and the calendar date.

A basic RTC without time and date counters uses software for this purpose. A common option is a 1 Hz square wave from an output pin. An RTC will have several possible events to generate a processor interrupt.

An RTC often has a dedicated power pin to allow operation when the rest of the microcontroller is powered down. This power pin is typically connected to a battery or separate power supply.

RTC ACCURACY AND FREQUENCY COMPENSATION

The accuracy of an RTC depends on the 32,768 Hz clock source. In a well-designed crystal oscillator, the major source of error is the crystal. An external TCXO can be used for highly accurate timing, or special frequency compensation techniques are used with less expensive crystals and the internal oscillator. There are three major sources of error from a crystal.

- Initial circuit and crystal tolerance
- Crystal drift with temperature
- Crystal aging

The graph below shows several concepts related to RTC accuracy.

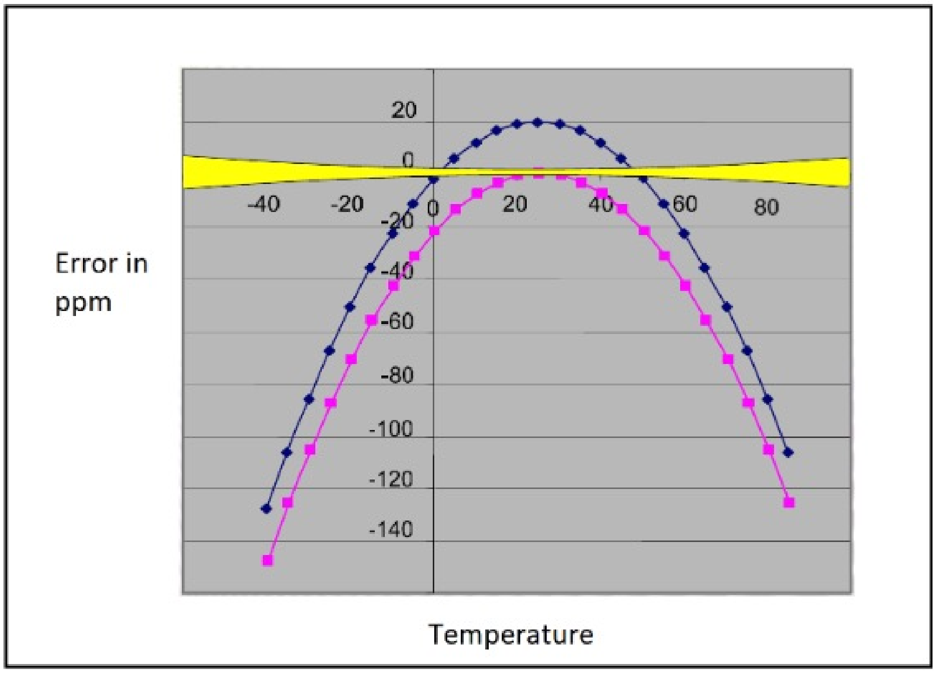


Figure 3. Graph showing error measurement using temperature used courtesy of Texas Instruments

The dark blue trace on this graph shows a typical initial tolerance and the change with temperature. The pink trace shows just the temperature error. The key to compensating for temperature is the fact that the behavior of a crystal is well known and predicted with a quadratic equation. If the initial error is measured after the circuit board is manufactured and the temperature is known, it is possible to compensate for the largest error sources.

The yellow band is a reasonable target for accuracy after careful compensation. Keep in mind 1 ppm over a year is about 30 seconds. Crystal aging is difficult to compensate for. Fortunately, aging is usually only a few ppm per year.

HOW TO CHANGE RTC TIMING

Here are two ways to change the timing of an RTC as part of a system to compensate for errors.

The first diagram (Figure 4) depicts the number of oscillator cycles counted by the prescaler for each period of the seconds counter.

The first two seconds are the usual 32,768 cycles. The software uses temperature readings and the initial error to determine that the oscillator is running a little fast and 32,768 cycles is actually a period of 0.99990 seconds. To compensate for this small error, the software tells the RTC to change the modulus of the prescaler to 32,781 for every fourth second to add some time.

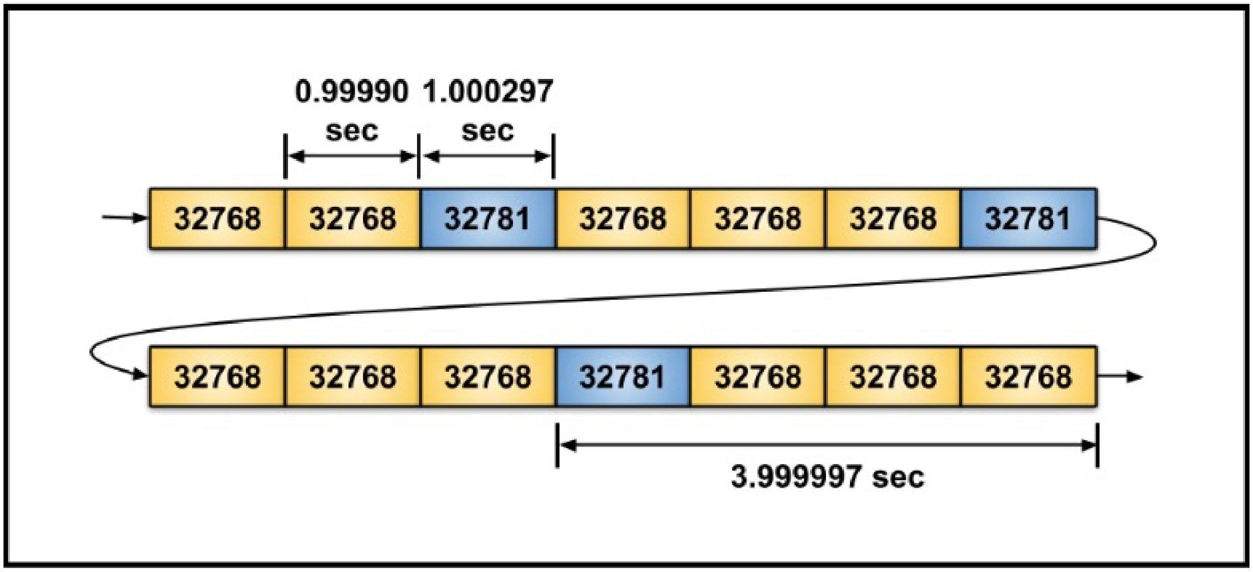


Figure 4. A representation of oscillator cycles counted by a prescaler

This technique has the advantage of a small change in the period from second to second. However, the technique requires an adjustable prescaler and additional registers to hold the special prescale count and the number of seconds between application of the special count. I think this is pretty cool. A little complicated but pretty cool.

What if the RTC does not have a special prescaler to adjust the timing? This diagram shows another method.

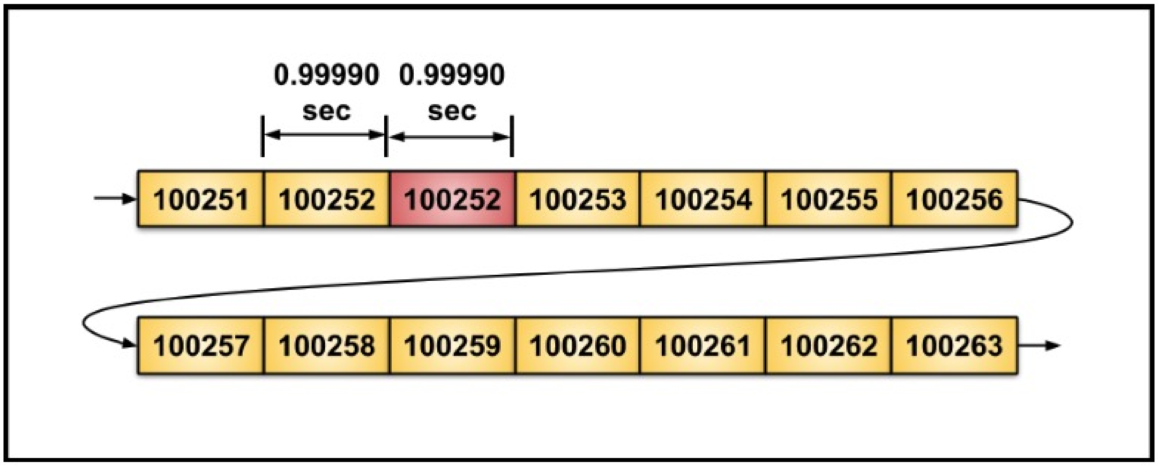


Figure 5. The same situation as shown in Figure 4, but without a prescaler

In this case, the number in the boxes is the seconds counter. The count shown is 100251 followed by 100252. The software has been continuously calculating the adjustment and keeping track of the RTC seconds count. When the error accumulates to exactly one second, the software adds or subtracts a second to adjust for the accumulated error.

A disadvantage of this technique is the change from second to second is large when an adjustment is done. This technique has the advantage of compatibility with any RTC.

SECURITY IN RTCS

Security is an interesting requirement. There are applications where time is used for billing customers for using a service or consuming a resource. There is an extensive body of practice around preventing or detecting hacks of RTCs. Techniques range from intrusion detection for enclosures to special features within the microcontroller.

The RTC on a microcontroller that I am currently using has special registers to allow software to permanently lock critical registers. Once locked, they cannot be changed and are protected from hacking or out-of-control code. Changing the time requires a complete reset of the microcontroller.

TIME AND DATE

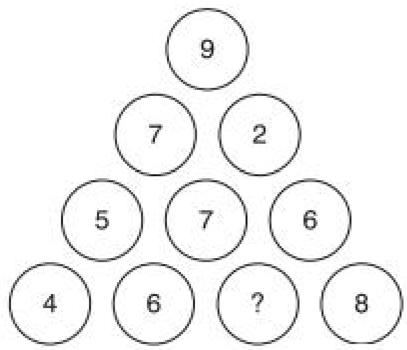
Some RTCs have hardware counters to maintain the time of day and calendar date. This requires counters for minutes, hours, days, months, years, and consideration for leap years. Time of day and calendar dates can also be kept by software.

A prominent example would be the functions in the C Standard Library seen in the time.h file. For a microcontroller, this system can be based on the seconds counter of an RTC. Four small, custom functions must be written to fully support the time.h library.

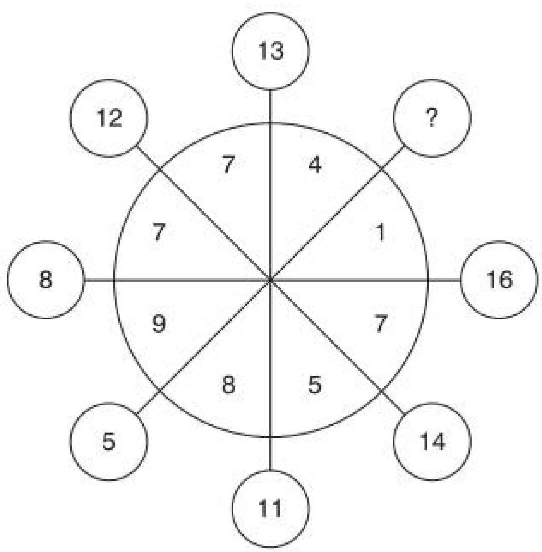
The one function of interest here is called by the time( ) function in the library, which returns the time as the number of seconds since a starting point called an "epoch," usually January 1, 1970. Typically, the custom function to read the hardware timer is named get_time( ) or a similar variation. All get_time( ) does is read the seconds counter and return the value. The library does the rest to turn this time in seconds to the current time of day and date.

Number Puzzles

1. What number should replace the question mark?



2. What number should replace the question mark?



3. What number should replace the question mark?

			14	
	22			
			34	
41				
		53		?

4. If A is substituted by 4, B by 3, C by 2, D by 4, E by 3, F by 2 and so on, then what will be total of the numerical values of the letters of the word SICK?

5. Two Fathers and Two Sons Riddle

   Two fathers and two sons sat down to eat eggs for breakfast. They ate exactly three eggs, each person had an egg. The riddle is for you to explain how.

6. A cuboid shaped wooden block has 6 cm length, 4 cm breadth and 1 cm height.

   Two faces measuring 4 cm x 1 cm are coloured in black.
   Two faces measuring 6 cm x 1 cm are coloured in red.
   Two faces measuring 6 cm x 4 cm are coloured in green.
   The block is divided into 6 equal cubes of side 1 cm (from 6 cm side), 4 equal cubes of side 1 cm(from 4 cm side).

   How many cubes will have green colour on two sides and rest of the four sides having no colour ?

   - [ ] A) 8
   - [ ] B) 24
   - [ ] C) 16
   - [ ] D) 4

Swami Vivekananda

- We are what our thoughts have made us; so take care about what you think. Words are secondary. Thoughts live; they travel far.

- The Vedanta recognizes no sin it only recognizes error. And the greatest error, says the Vedanta is to say that you are weak, that you are a sinner, a miserable creature, and that you have no power and you cannot do this and that.

- If faith in ourselves had been more extensively taught and practiced, I am sure a very large portion of the evils and miseries that we have would have vanished.

- The will is not free - it is a phenomenon bound by cause and effect - but there is something behind the will which is free.

- If you have assimilated five ideas and made them your life and character, you have more education than any man who has got by heart a whole library.