Siemens Digital Industries Software

From proof-of-concept to product - Initial design of a MEMS sensor

MEMS design

Executive summary

In previous papers, we have covered how to design and verify an IoT tank fluid-level monitoring system. We covered how to create a proof-of-concept and prototype. In this series of whitepapers, we explore the detailed product design of the MEMS pressure sensor within this system. In this initial whitepaper, we introduce the piezoelectric micromachined ultrasonic transducer (PMUT) sensor, show how to use finite element analysis on the device, and cover the analog frontend circuitry that converts the sensor waveform into the depth of liquid in a tank.

Ryan Diestelhorst, and Nicholas Williams

eda.sw.siemens.com

Contents

Introduction

In our popular series of IoT whitepapers, we described how to prototype an IoT edge device, implementing it with Siemens complete design flow (such as this whitepaper). We covered the overall system design including a pressure sensor, analog processing electronics, conversion of the pressure to digital, input into a microcontroller, and creation of firmware to monitor the level of liquid, sounding an alarm when it is too low.

Now, we start a new series of whitepapers where we focus on the detailed product design of the MEMS pressure sensor, the analog front-end electronics, behavioral modeling of the sensor, and MEMS-IC co-simulation. In this paper, we focus on the enhancement of the system level design by switching the pizeoresistive pressure sensor with a piezoelectric micromachined ultrasonic transducer (PMUT). This whitepaper covers the simulation and design of the sensor and its integration into the overall system, illustrating how the design infrastructure can be easily modified to accommodate design changes between prototypes. In subsequent papers, we look in more detail at the sensor physical implementation and the design of the electronics and software.

Understanding the system

We are developing a tank fluid-level monitoring system. This system can be deployed in breweries, wineries, and other beverage facilities at the IoT edge to collect the status of the fluid levels and to notify technicians proactively if there are any issues (such as leaks). We have modified the system (Figure 1) to utilize the PMUT sensor that monitors the fluid level in the tank and periodically transmits measurements to a gateway device.

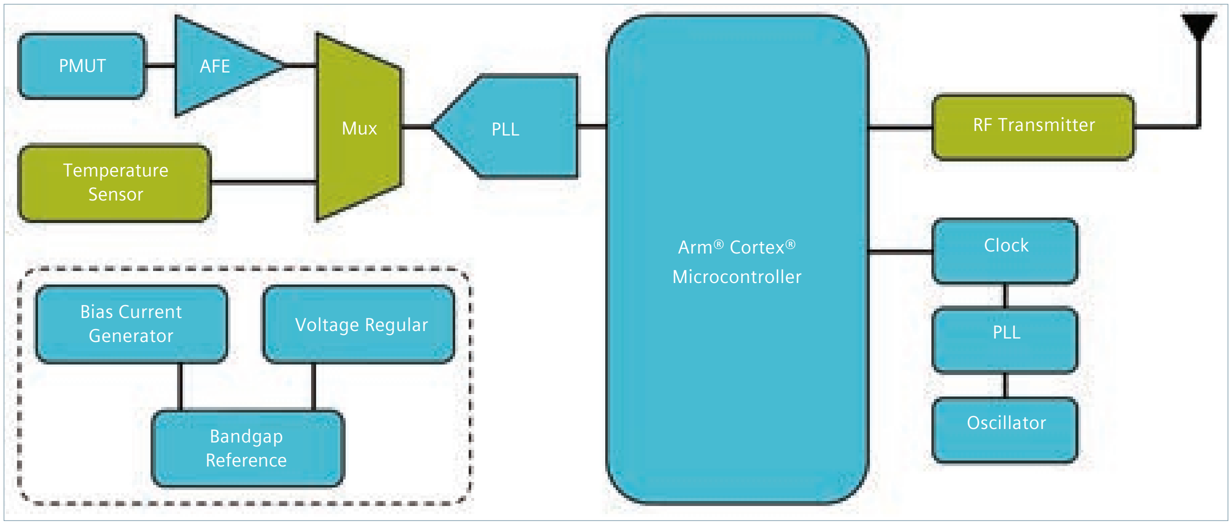


Figure 1: Block diagram of the tank fluid-level monitoring system.

The ultrasonic sensor produces acoustic waves in the tank and then measures the waves that get reflected by the liquid, resulting in a small analog waveform that requires amplification. The time difference between sending the wave and the measured reflected wave is proportional to the distance to the liquid. An analog front-end (AFE) amplifies the MEMS waveform and converts the signal to an integrated voltage representing time-of-flight or twice the depth to the liquid. An analog-to-digital converter (ADC) circuit converts the voltage to digital as an input into the microcontroller that runs the software. The clock, PLL, and oscillator circuits are supporting modules to the digital circuitry and the bias current generator, voltage regulator, and bandgap reference are supporting modules to the analog circuitry. The RF transmitter sends the data to the gateway. The Arm® Cortex®-M3 microcontroller interfaces to the analog circuits and RF transmitter. In the future, we plan on adding a temperature sensor which necessitates a multiplexer, but for this whitepaper we will not implement those two elements.

Understanding the sensor

Ultrasonic transducers use acoustic waves to detect the distance between the sensor and other objects. They operate by converting electrical energy into mechanical energy, and in most cases, converting mechanical energy back to electrical energy. It is this duality of function that allows them to produce pressure waves that travel to an object or interface and subsequently detect them when they are reflected back to the source. They are widely used in everyday applications, but have historically been bulkier, more power hungry, and more expensive than competing technologies. This has limited their use in, for example, consumer applications, but this is changing fast. New generations of miniaturized ultrasonic transducers coming to market consume up to an order of magnitude less power than their predecessors. There is no better example than the current development and implementation of PMUTs, which we use for the sensor in our system.

A PMUT consists of a piezoelectric thin-film membrane (typically PZT or AlN material) suspended above a cavity. When an electric impulse is applied, the membrane vibrates to produce acoustic energy in the medium directly contacting it. At its designed resonant frequency, a PMUT can produce a substantial amount of energy at very low power. Figure 2 shows a representative cross section of a PMUT, where the membrane thickness and diameter can be tuned to optimize the resonant frequency for a given medium. PMUTs are produced using well-established silicon semiconductor fabrication processes. Unlike many competing technologies, this allows them to be used in very high-volume applications at low costs, and more importantly, to be seamlessly integrated with CMOS to provide complete sensing systems on a chip.

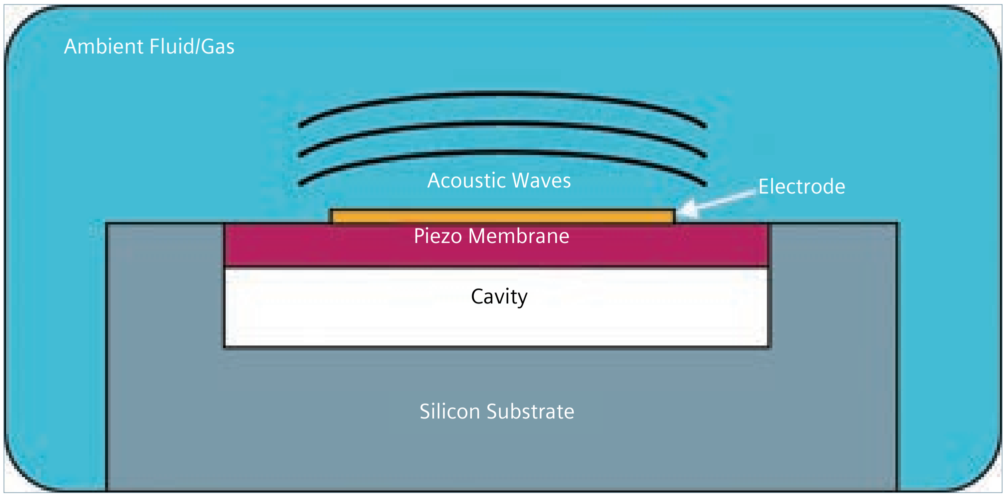


Figure 2: Representative cross section of a PMUT transducer (Source: OnScale®).

Simulating the PMUT using finite element analysis

As a first step in the PMUT design, we simulate the PMUT using a simulation technique called finite element analysis (FEA). By using FEA, intricate differential equations that describe the structure's behavior can be converted into algebraic expressions, which in turn can be solved numerically. FEA can be used several ways in our design: to explore design concepts, perform functional verification of design performance, and to optimize the design. FEA can explore the real world variations in non-ideal geometric shapes, fabrication processing, and material properties and it is well suited to problems with complex boundary conditions.

FEA tools rely on dividing the simulated structure into low-order finite elements or mesh. This process is assisted by auto-mesher functions that partition the structure efficiently. By refining the meshing where needed, the designer can get a set of accurate simulation results. By combining the mesh with other information from the electrical, mechanical, and thermal domains, coupled field modules can be used to simultaneously solve for electrostatics, coupled electro-mechanics, piezoelectric, piezoresistive, damping effects, and other characteristics.

Many important device characteristics can be simulated for ultrasonic sensors, including:

- Electrical impedance
- Mode shapes
- Pressure and displacement levels
- Beam patterns
- Directivity index
- Efficiency
- Pulse-echo response
- Crosstalk
- Bandwidth
- Material characteristics
- Mechanical shocks
- Bondline effects.

We utilize a powerful cloud-enabled, multi-physics FEA tool called OnScale, which provides all the capabilities mentioned above and executes them on a massively parallel Cloud infrastructure to allow design studies to be completed in hours instead of weeks.

Designing the sensor

The nature of piezoelectric ultrasonic transducers presents a unique set of challenges to solve using simulation techniques such as FEA. One of the most fundamental challenges is the relative size of the computational domain required. Accurately capturing the small spatial gradients in stress created as the wave propagates demands a fine mesh (such as 8-12 elements per wavelength) to be deployed over a long path (such as 10-250 wavelengths). The result is a large number of elements needed to generate efficient simulation grids. OnScale is uniquely suited for models of this size due to its highly efficient solver and advanced hybrid meshing techniques.

The PMUT used for this effort is modeled as a single 3D transducer with a nominal cavity width of 400 um, a top electrode diameter of 200 um, and a cavity depth of 40 um. The PMUT is excited with a series of voltage pulses applied to the electrode on top of the membrane. Figure 3 shows how the membrane deforms when a series of pulses at is applied at 32 V. The deformation is exaggerated for clarity.

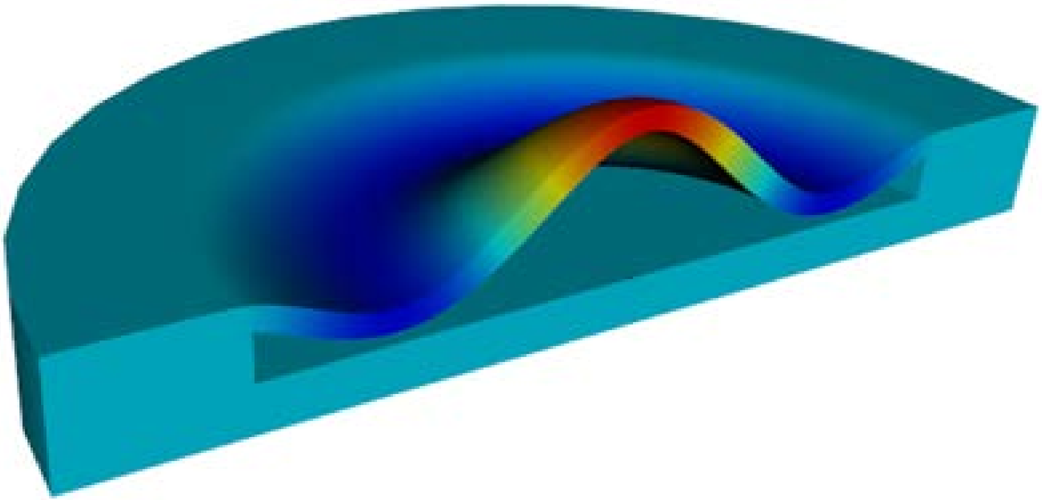


Figure 3: Deformation of the PMUT membrane during electrical stimulation (Source: OnScale).

To optimize this design, we need to maximize the mechanical energy received by the PMUT after reflection off of the fluid surface. We do this by building a design of experiments (DoE) that covers enough of the design space to contain our optimal solution. In this case specifically, we vary the design parameters (Table 1) to determine the best design.

Both the piezoelectric layer thickness and the membrane thickness are swept from 1.0 to 2.5 μm in 0.1 μm steps, yielding a total of 256 designs to be simulated. A subset of the results is shown in Figure 4, where resonant frequency is plotted against the swept parameters. These results inform the optimal dimensions of the design for our given environmental conditions prior to prototyping. In this case, the simulations indicate that power of the reflected wave is optimal at lower frequencies due to losses in air. We choose a piezoelectric layer thickness of 1.2 μm and a membrane thickness of 1.2 μm, which yields a low resonant frequency of 122 kHz, while keeping within the limits imposed by the manufacturing process. A design space of this size would be difficult to explore with a legacy FEA package, which is why OnScale was chosen for this analysis.

Table 1: Swept PMUT optimization parameters.

Parameter	Range	Steps
Piezoelectric Layer (PZT) Thickness	1.0 – 2.5 μm	16
Membrane (SiO_2) Thickness	1.0 – 2.5 μm	16

Resonant Frequency

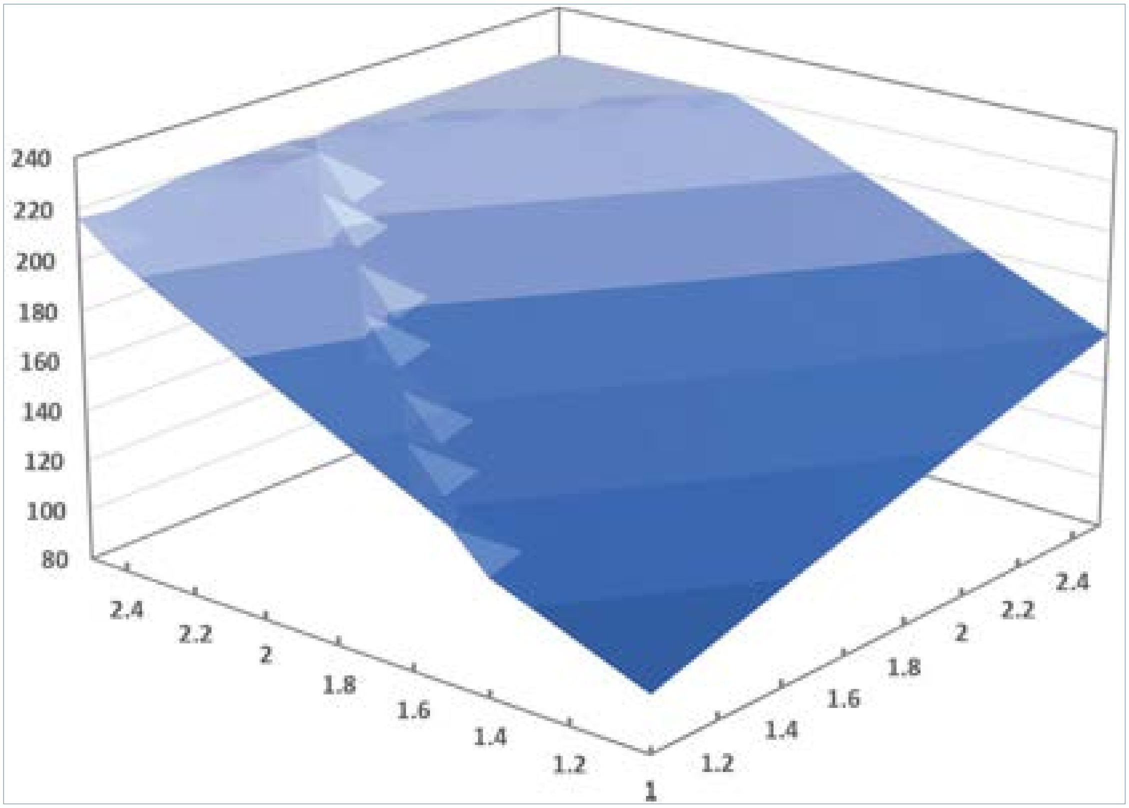


Figure 4: Resonant frequency optimization design space for the PMUT (Source: OnScale).

Designing the analog front-end

Figure 5 shows the analog front-end used in this design. The PMUT is schematically captured in S-Edit using a voltage source with the properties and parameters that match the output of the FEA study.

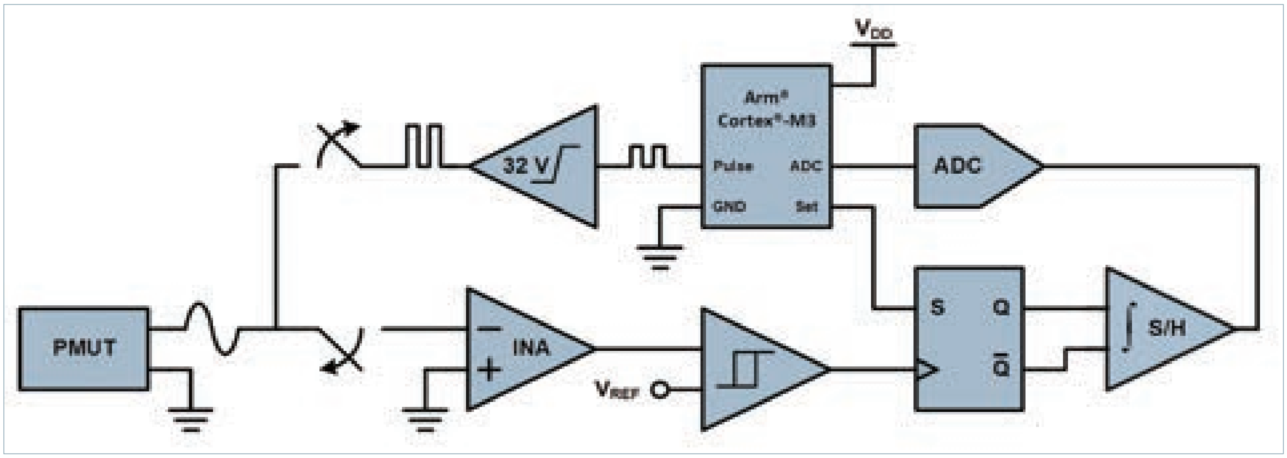


Figure 5: Analog front-end for the PMUT transducer.

The time-of-flight function begins when the microcontroller sets the SR latch, which begins accumulating charge on a sample-hold integrator. At the same time, a series of pulses is produced by the microcontroller at the designed resonant frequency of the PMUT (122 kHz). The pulse is amplified to 32 V using a charge pump DC-DC converter and a digital level-shifter because the core power voltage is 2.5 V and it has to be raised to 32 V as required by the PMUT. The transducer receives the signal pulses, creating a pressure wave in the tank that is reflected back at the fluid interface (Figure 6). This reflected wave is sensed by the PMUT, creating a voltage signal on the membrane with a peak amplitude on the order of 500 μV that is delayed based on the how long it takes for the pressure waveto travel to the top of the liquid level and be reflected back to the PMUT.

An instrumentation amplifier, with a gain of 70 dB, amplifies the PMUT signal and feeds it to a Schmitt trigger. The Schmitt trigger, with built-in hysteresis, resets the SR latch when the reflected amplified signal rises above the threshold voltage (VREF) of 1.25 V. The latch reset also locks the sample-hold integrator, which has now accumulated charge on its output capacitor during the time it took for the pressure wave to traverse the air volume of the tank and return to the PMUT. An ADC before the microcontroller converts the voltage output of the integrator to a digital value that is a linear measure of the time-of-flight, and by subtraction, the fluid-level inside the tank.

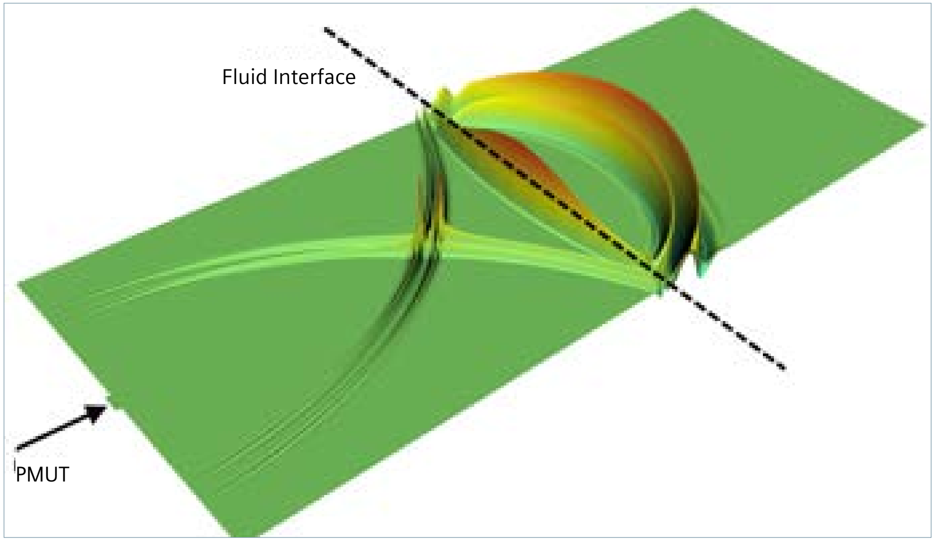


Figure 6: Simulated wave propagation inside the tank (Source: OnScale).

Performing initial system simulation

When launching the simulation, S-Edit creates the complete Verilog-AMS netlist and passes it to AFS (Analog FastSPICE). AFS automatically adds analog/digital connection modules and then partitions the design for simulation. AFS simulates the analog (SPICE and Verilog-A) and sends the RTL to Questa for digital simulation. Both simulators are invoked automatically and during simulation the signal values are passed back and forth between the simulators whenever there is a signal change at the analog/digital boundary. This means, that regardless of the design implementation language, the designer drives the simulation from S-Edit and the design is automatically partitioned across the simulators. Then, the designer can interact with the results using the Questa and AFS waveform viewers. Figure 7 shows the results of the simulation.

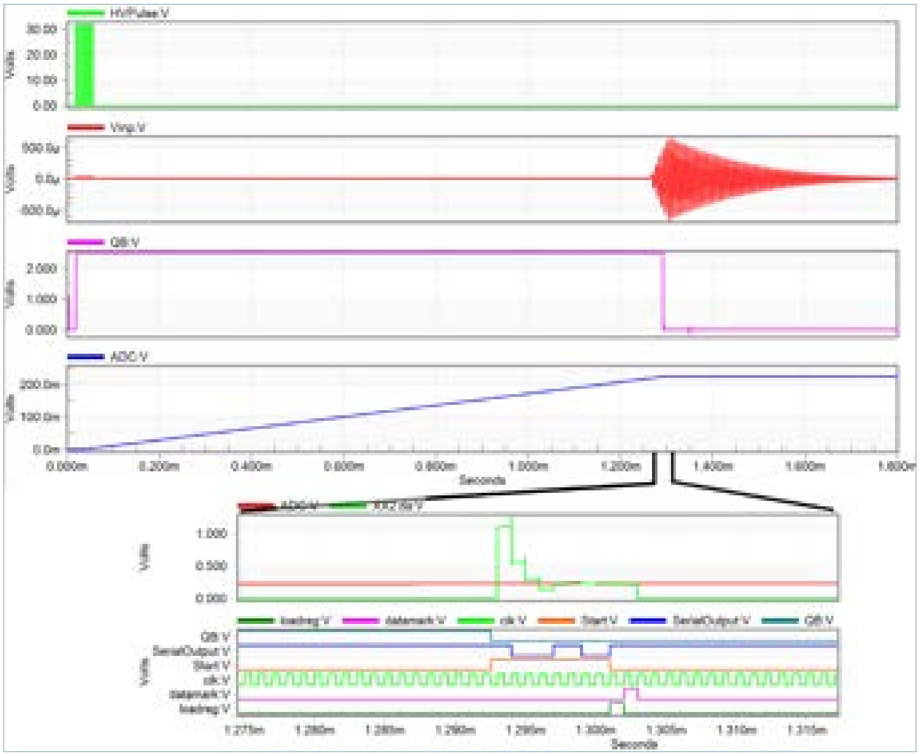


Figure 7: Initial system simulation results.

The waveform shows the high voltage pulse at 122 kHz (green). Next, the input to the instrumentation amplifier is plotted (red) with the PMUT response occurring around 1.2 ms after the initial pulse. Assuming a sound speed of 343 m/s in air, this corresponds to a fluid level 21 cm below full. The third waveform (purple) shows the output of the SR latch, which triggers when the PMUT output voltage exceeds ~400 µV (before amplification). Lastly, the output of the integrator is plotted (blue), showing a linear increase in voltage until the SR latch triggers the integrator to hold. The output is converted to digital by an 8-bit ADC before being sent to the microcontroller for calculation of the fluid level. For this design, the total tank depth is assumed to be 2.1 m. The maximum time-offlight can then be calculated to be 12.2 ms, which subsequently yields an integrator output of 2.17 V. This is close to the maximum input voltage of the chosen ADC (2.2 V), as designed. For different size tanks, the gain of the instrumentation amplifier can be adjusted accordingly.

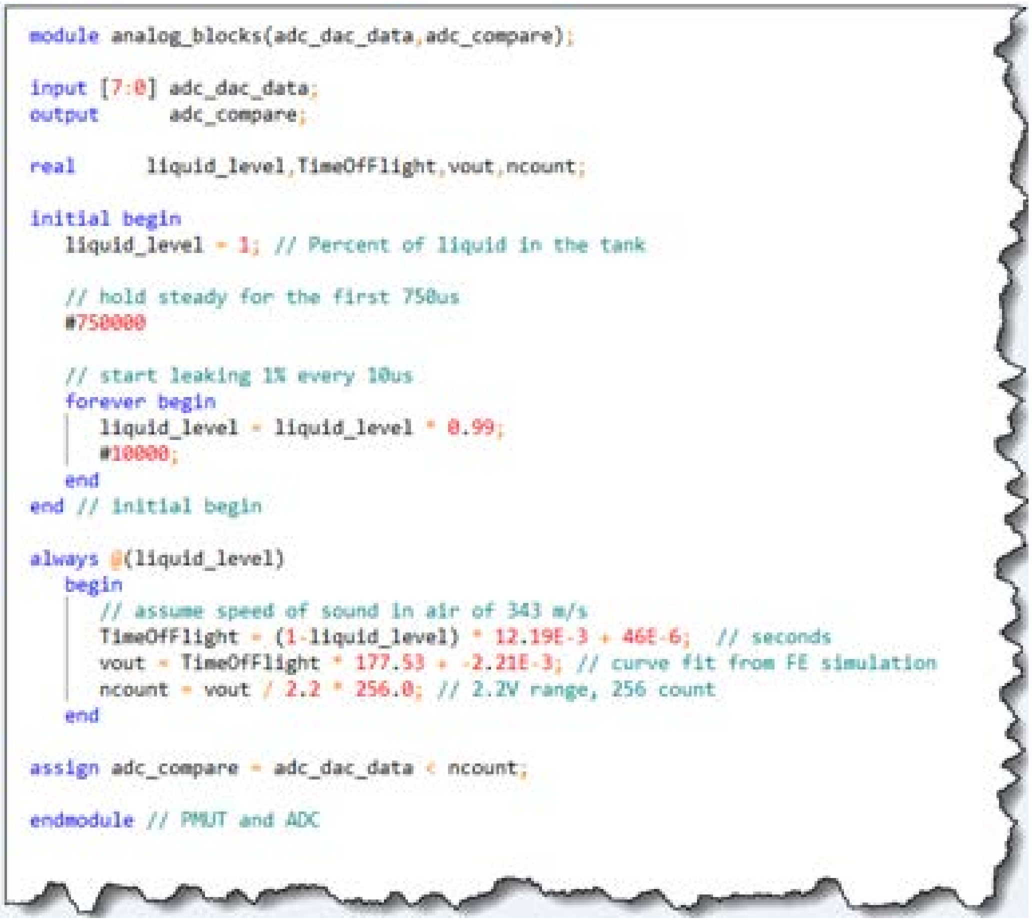
Developing the software

The software running on the microcontroller includes code to periodically convert the sensor output into fluid depth and to report any significant fluid level change to the monitoring system. Given that the integrator and ADC are approximately linear, the ADC output can be converted to time using a linear slope and intercept. These values are initially simulated and then verified through measurement of the analog simulation results. The time-of-flight is converted to air level by multiplying by the speed of sound in air (343 m/s). Finally, fluid level is derived as the total tank depth minus the air level.

Performing actual system simulation

Because we already verified the MEMS transducer, and we want to significantly save simulation time, we use a stand-in Verilog behavioral model (Figure 8) of the sensor that samples data using discrete simulation time steps instead of continuous analog sampling.

This model closely matches the analog sensor behavior, but samples the data every 1 us for faster system simulation, even though the ADC takes 12us for each conversion. Figure 9 shows the system simulation results. The transcript (A) displays the liquid level change warning and the waveforms show the liquid level and acoustic sensor values corresponding to the leak (B). The system spends most of the simulation time asleep, waking up to take depth readings periodically (C).



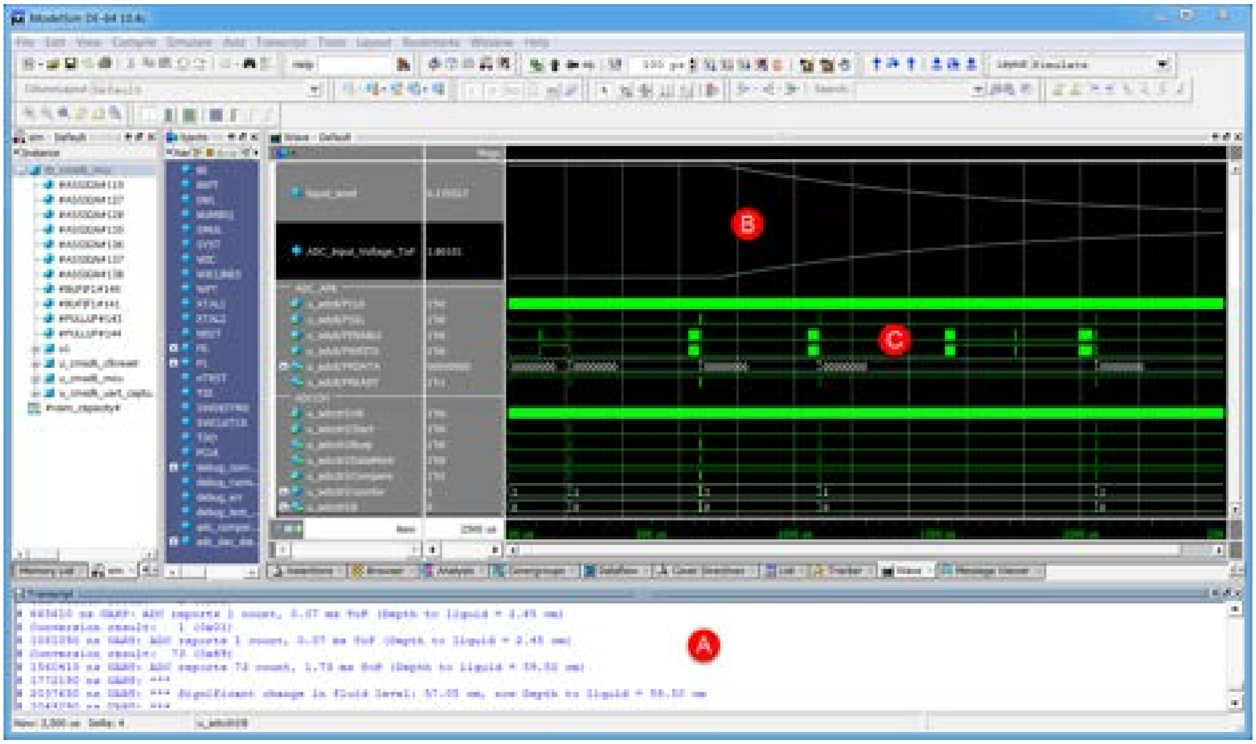
```
module analog_blocks(adc_dac_data,adc_compare);

input [7:0] adc_dac_data;
output      adc_compare;

real      liquid_level,TimeOfFlight,vout,ncount;

initial begin
   liquid_level = 1; // Percent of liquid in the tank

   // hold steady for the first 750us
   #750000

   // start leaking 1% every 10us
   forever begin
   |  liquid_level = liquid_level * 0.99;
   |  #10000;
   end
end // initial begin

always @(liquid_level)
   begin
   |  // assume speed of sound in air of 343 m/s
   |  TimeOfFlight = (1-liquid_level) * 12.19E-3 + 46E-6;  // seconds
   |  vout = TimeOfFlight * 177.53 + -2.21E-3; // curve fit from FE simulation
   |  ncount = vout / 2.2 * 256.0; // 2.2V range, 256 count
   end

assign adc_compare = adc_dac_data < ncount;

endmodule // PMUT and ADC
```

Figure 8: Stand-in simulation model for the PMUT.

Next steps

Some of the next steps are to flesh out the analog section, add an RF transceiver, and add a communication block and a device ID to the design in order to connect to the Internet. In addition, developing sophisticated software for the system and for the Web opens the doors to many possibilities, including potentially: two-way communication (directing the system to act), over-the-air software updates, preventative maintenance, and Cloud services. We also might move forward with adding a temperature sensor to the system to better calibrate the measurements. Look for future whitepapers on developing, implementing, and verifying this IoT system.

To learn more about the Tanner solution, view this website.



```
# [illegible] ns [illegible]: ADC reports 1 count, 0.07 ms ToF (Depth to liquid = 2.45 cm)
# Conversion result:   1 (0x01)
# [illegible] ns [illegible]: ADC reports 1 count, 0.07 ms ToF (Depth to liquid = 2.45 cm)
# Conversion result:  73 (0x49)
# [illegible] ns [illegible]: ADC reports 73 count, 1.73 ms ToF (Depth to liquid = [illegible] cm)
# [illegible] ns [illegible]: ***
# [illegible] ns [illegible]: *** Significant change in fluid level: [illegible] cm, new Depth to liquid = [illegible] cm
```

Figure 9: System simulation results.

Siemens Digital Industries Software

Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific
Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

About Siemens Digital Industries Software

Siemens Digital Industries Software is driving transformation to enable a digital enterprise where engineering, manufacturing and electronics design meet tomorrow. Our solutions help companies of all sizes create and leverage digital twins that provide organizations with new insights, opportunities and levels of automation to drive innovation. For more information on Siemens Digital Industries Software products and services, visit siemens.com/software or follow us on LinkedIn, Twitter, Facebook and Instagram. Siemens Digital Industries Software – Where today meets tomorrow.

siemens.com/software