

# Real-world, accessible electronic devices, test equipment, and software tools for engineering education

Presented By:  
Mark Thoren  
Nathan Frey  
Analog Devices, Inc.

©2023 Analog Devices, Inc. All rights reserved.

▶ This is the light cover slide option.

## For Reference: session abstract

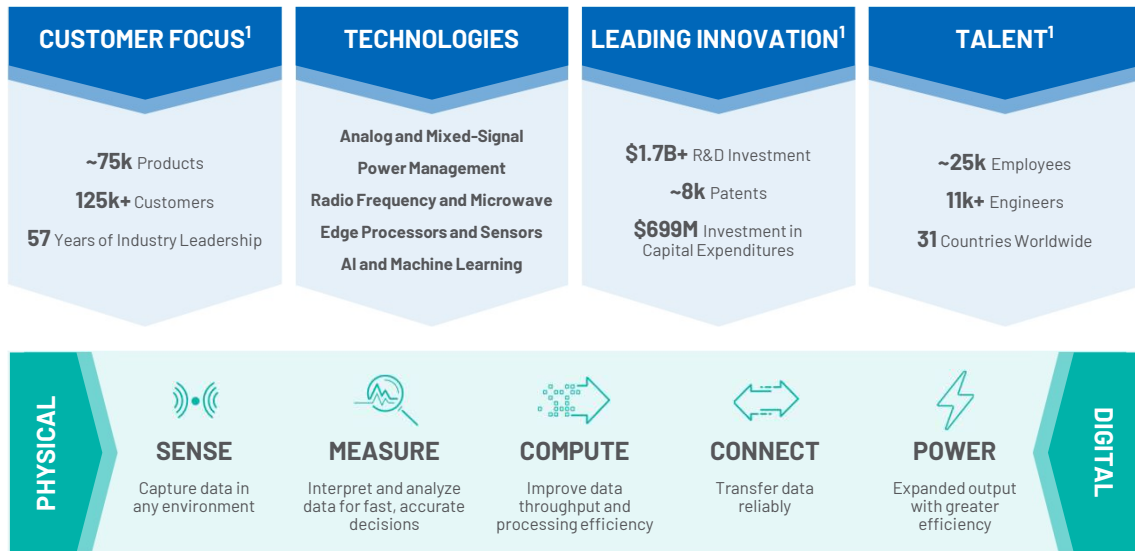
Analog Devices is the leading global high-performance analog technology company. The ADI University Program strives to make engineering education accessible through hundreds free Active Learning lab exercises, affordable test instrumentation, low-cost parts kits, and hardware modules designed to back up textbook theory with hands-on experience.

In this focus group, we will discuss Analog Devices' educational offerings, including:

- ▶ Free, open-source lab exercises on basic circuits, data conversion (A to D and D to A), amplifiers, filters, oscillators, transmission lines, radio, and more.
- ▶ Affordable, student-accessible test equipment that is also at home on a professional's lab bench
- ▶ A free, complete course on Software Defined Radio for Engineers
- ▶ A power electronics learning module that exposes the operation of buck and boost converters, without requiring expensive test equipment
- ▶ And an exciting new Artificial Intelligence Microcontroller with Neural Network Accelerator, bringing modern computer science to the physical world.

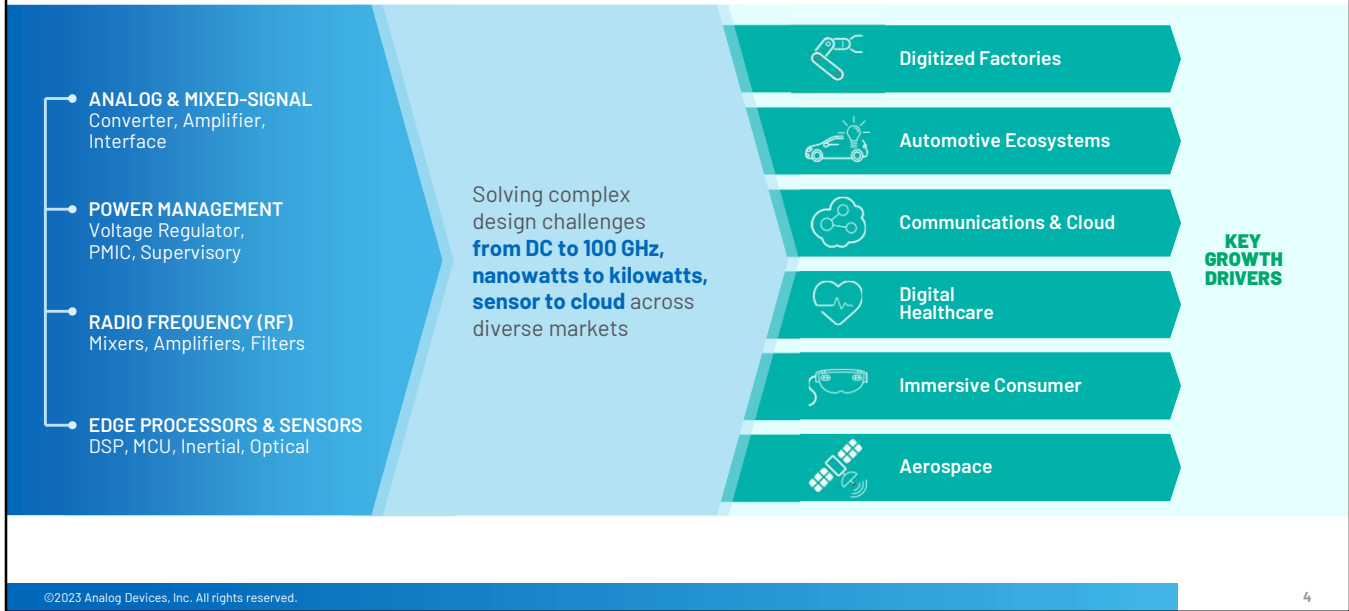
We will have hardware on hand for live demos, and will encourage active discussion during the session.

# Creating a More Connected, Safer, and Sustainable Future



1. Information as of end of fiscal year 2022

# Leading Technologies Across Applications

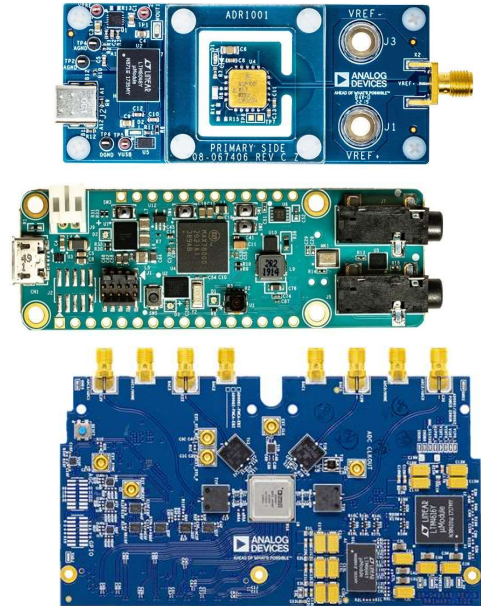


## Analog Devices: Just analog, right?

...Wrong!

- ▶ ADR1001 Ovenized voltage reference
  - Hairdryer and multimeter experiment
- ▶ MAX78000 A.I. uC w/ Convolutional neural network
  - Handwriting, facial recognition on a tiny microcontroller, at the "intelligent edge"
- ▶ Apollo Mixed Signal Front End
  - 20Gsps ADC, 28Gsps DAC
  - Software-defined Radio
  - Digital Signal Processing

And everywhere in between – devices, software drivers, applications, network connectivity, and that's just to start



## ADI University Program Goals

### ▶ ADI University Program goals

- **“Bridge the gap between academia and industry, drive innovation through collaborative research and education, and empower the next generation of engineers and scientists to solve the world's toughest challenges, creating a better future for all.”**
- But “keeping it real”, we’re selfish: Ensure a steady flow of analog engineers for our own selfish purposes, accepting that everyone benefits (even future competitors).

### ▶ **Presentation Goals:**

- Present a broad overview of ADI’s educational material
- Stimulate the imagination for additional labs, instruments, etc. Please ask questions, make suggestions for what more ADI can do.

## University Program Elements

- ▶ Content! (Labs, Lessons, Tutorials)
  - [wiki.analog.com/university](https://wiki.analog.com/university)
- ▶ Educational Modules (Test Instruments, "lesson on a board")
- ▶ Ltspice
- ▶ Senior design / thesis projects: Another use for eval boards and reference designs
- ▶ And of course, parts: Op-amps, regulators, ADCs, DACs, etc.



The screenshot shows the top navigation bar of the Analog Devices Wiki, including the logo and the text "Wiki". Below the navigation bar is a green notification banner stating: "This version (14-Jan-2023 16:23) was approved by Doug Mercer. The Previously approved version (28-Jul-2022 14:48) is available." The main content area is titled "Engineering University Program" and includes a "Mission" section with a paragraph about Analog Devices' commitment to education. Below the mission is a section titled "Teaching Materials & Lab Exercises" which contains a numbered list of 8 items, each with a sub-link for "Lab Exercises".

Resources and Tools ▾ Education Content ▾ Wiki Help ▾ Wiki Tools ▾

This version (14-Jan-2023 16:23) was approved by Doug Mercer. The Previously approved version (28-Jul-2022 14:48) is available.

### Engineering University Program

#### Mission

Analog Devices is as passionate about educating the next generation of young circuit design engineers as it is about pioneering the next technological breakthroughs. The University Program is a platform where

#### Teaching Materials & Lab Exercises

1. Introduction to Electrical Engineering - Teaching Material
  - Introduction to Electrical Engineering - Lab Exercises
2. Circuits I & II - Teaching Material
  - Circuits I & II - Lab Exercises
3. Electronics I & II - Teaching Material
  - Electronics I & II - Lab Exercises
4. Power Electronics - Teaching Material
  - Power Electronics - Lab Exercises
5. Digital Communications - Teaching Material
  - Digital Communications - Lab Exercises
6. Introduction to Digital Electronics- Teaching Material
  - Introduction to Digital Electronics - Lab Exercises
7. Signals and Systems - Teaching Material
  - Active Learning Module Signals and Systems Lab Activities
8. Electromagnetic Fields and Waves - Teaching Material
  - Active Learning Module Electromagnetic Fields and Waves Lab Activities

©2023 Analog Devices, Inc. All rights reserved. 5 July 2023 7

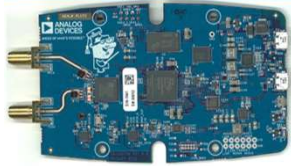
## Affordable (but good) instruments

- ▶ Courseware
- ▶ >100 Active Learning Module Lab Activities
  - Circuits I & II
  - Electronics I & II
  - Power
  - Communications
- ▶ Instructional Videos
- ▶ “Virtual Classroom” EngineerZone Community
  - Help and Guide Students in their Learning Experience
  - Includes:
    - Blog Posts from ADI Staff
    - Document Sharing
    - Discussion
    - Questions





# ADALM-PLUTO for Junior - Senior - Grad Level Communications Labs



\$230



- The easy to use ADALM-PLUTO active learning module (PlutoSDR) introduces the fundamentals of
  - Software-defined radio (SDR)
  - Radio frequency (RF)
  - Wireless communications
- Helps develop a foundation in real-world RF and communications

Pluto	
Target user	Junior - Senior
Design	Open
RX/TX	1/1
Frequency Range	325 MHz to 3.8 GHz
TRX	AD9363
FPGA	Zynq Z-7010
Connectivity	MATLAB, GNURadio, etc
Channel bandwidth	200kHz - 20 MHz
Cost	\$149

**Textbook + Labs**  
Software-Defined Radio For Engineers

- Kit Includes**
- 1 x ADI PlutoSDR active learning module
  - 2 x antenna (824-894 MHz/1710-2170 MHz)
  - 1 x 15 cm SMA cable
  - 1 x USB cable

# Reference Designs and Eval Boards: Academic project "enablers"



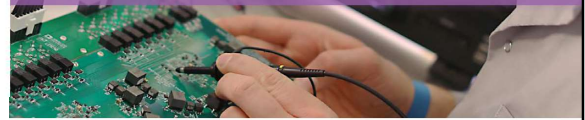
- ▶ Diverse applications:
  - Thermocouples
  - Gas detection
  - pH, Turbidity, Fluorescence
  - Strain gauge
- ▶ Typically Pmod or Arduino
- ▶ Typically include software examples
- ▶ Lets students focus on the important aspects of their projects



## SOLUTIONS FOR RAPID PROTOTYPING

Answering the Needs of Practicing Engineers

Analog Devices participates in many vibrant hardware and software ecosystems, from Arduino to Pmod. Rapidly make your prototype and test your concepts. ADI's reference designs support many different applications and technologies, and include everything from hardware to embedded firmware that makes it easier for engineers to get their product concepts up and running fast.



### Arduino Shields

Arduino shields are a popular hardware form factor that was first standardized by Arduino, and typically refer to being mechanically and electrically compatible with the Arduino Uno Rev 3 board. Many different FPGA, microprocessor, and DSP vendors provide plug and play connectivity into their development boards and environments using this form factor. Arduino shields provide analog and digital pins to configure devices and digitize signals coming from the real world. The digital communications protocols supported by Arduino shields are SPI, I2C, UART, PWM, and GPIO. All the below boards are compatible with either the EVAL-ADICP390 or EVAL-ADICP3029 and should work with any compatible Arduino form factor microcontroller platform. All software is open source and can be found on [github.com/analogdevicesinc/eval](https://github.com/analogdevicesinc/eval).

#### Water Quality Measurement System

- ▶ Measure from 1 to 4 sensor channels
- ▶ Selectable SPI, I2C, or UART communication
- ▶ 10-pin JTAG/SWD connector for easy programming
- ▶ Visit [analog.com/EVAL-CN0429-EBZ](https://analog.com/EVAL-CN0429-EBZ)



#### Dual Electrochemical Gas Detector

- ▶ Temperature compensation
- ▶ Work with industry-standard gas sensors
- ▶ Programmable for a variety of gases
- ▶ Visit [analog.com/EVAL-CN0398-AR02](https://analog.com/EVAL-CN0398-AR02)



#### Water Turbidity Measurement System

- ▶ 0 FTU to 1000 FTU measurement range
- ▶ ±0.5 FTU system accuracy (up to 1000 FTU)
- ▶ Integrated ambient light rejection
- ▶ Visit [analog.com/EVAL-CN0409-AR02](https://analog.com/EVAL-CN0409-AR02)



#### NDIR Thermopile-Based Gas Sensing Design

- ▶ Optimized for CO<sub>2</sub> gas
- ▶ Single supply
- ▶ Visit [analog.com/EVAL-CN0338-AR02](https://analog.com/EVAL-CN0338-AR02)



#### Total Dissolved Solids Measurement System

- ▶ Temperature compensation
- ▶ 1 µs to 1 s measurement range
- ▶ Standard DIN conductivity probe connector
- ▶ Visit [analog.com/EVAL-CN0411-AR02](https://analog.com/EVAL-CN0411-AR02)



#### Electrochemical Toxic Gas Detection

- ▶ Programmable for multiple other gases
- ▶ Resolution down to 1 ppm
- ▶ Low power, single-supply operation
- ▶ Visit [analog.com/EVAL-CN0357-AR02](https://analog.com/EVAL-CN0357-AR02)



## LTspice What's the point?

- ▶ Is it a "Sanity Check"? (No, see a therapist.)
- ▶ "That is how you verify the design" (No, you do that on the bench)

- ▶ So what is the point?

"The point of simulation is so you understand your circuit better... To develop intuition... There is no way to understate the value of cultivating intuition about how your circuit works."

<https://www.youtube.com/watch?v=x6TrbD7-lwU>



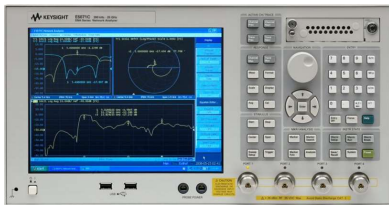
LT Spice with Mike Engelhardt



## Transmission Lines and Standing Waves with LTspice and low-cost materials

- ▶ Challenges making hands-on lab on wave propagation labs
  - TDRs are expensive
  - Fast scopes and network analyzers are expensive
  - You can use slow equipment... (if you have kilometers of cable!)

Keysight E5071C

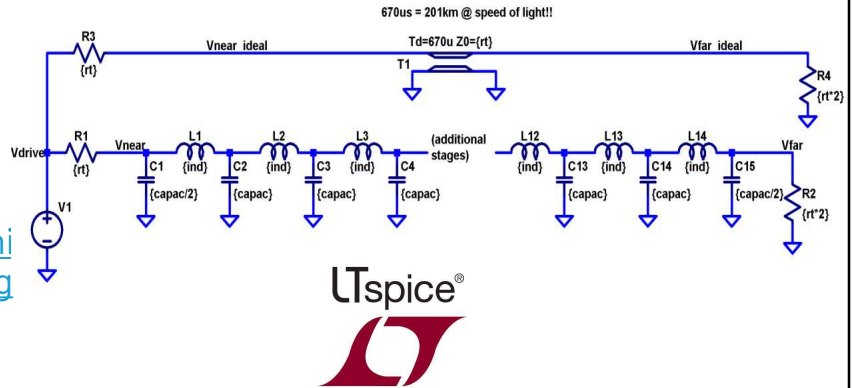


300m RG58  $\approx 1\mu\text{s}$   
(for \$130)



# Kitchen Table Solution: Pringles Can and LTspice

- ▶ Establish similitude between simulation and something you can touch
- ▶ Trust that the LTspice simulation will hold for faster situations who's measurement exceeds the student's test equipment budget
- ▶ Full Writeup: [https://wiki.analog.com/university/labs/tlines\\_standing\\_waves\\_adalm2000](https://wiki.analog.com/university/labs/tlines_standing_waves_adalm2000)



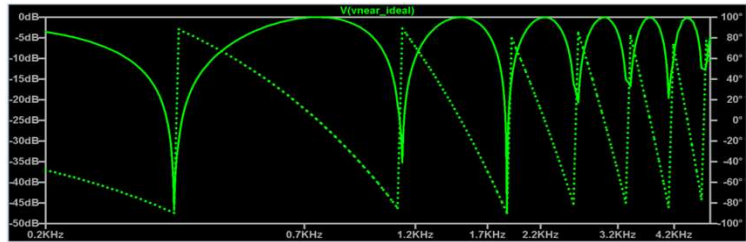
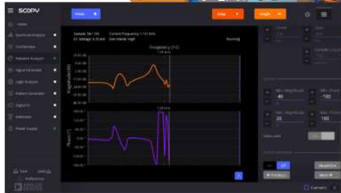
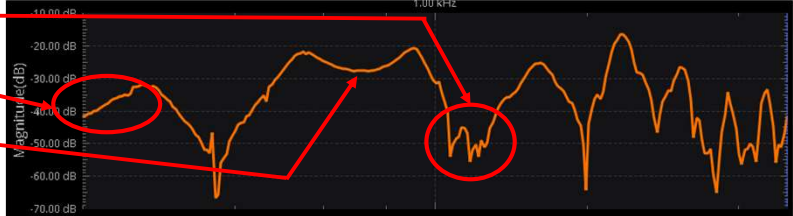
- ▶ URL: [https://wiki.analog.com/university/labs/tlines\\_standing\\_waves\\_adalm2000](https://wiki.analog.com/university/labs/tlines_standing_waves_adalm2000)

# Scopy Network Analyzer vs. LTspice

Resonance in metal cap (changes with thumb pressing)

Speaker response lower limit

Who Knows? (Any acoustics experts in the audience?)



LTspice simulation

# Same Model Applied to Switched Cap Power Supplies

▶ <https://wiki.analog.com/university/courses/electronics/switched-cap-power-supplies>

▶ Objectives:

- First exposure to the idea of storing energy
- First exposure to a switching regulator of any type.
- Introduction to LTspice

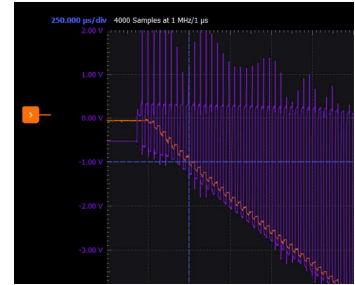
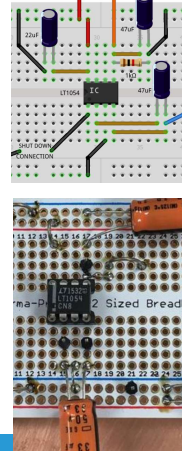
▶ Why LT1054, of all parts?

- Slow (25kHz), will work on solderless breadboard
- Versatile – demonstrates inverter, doubler, and a very silly buck

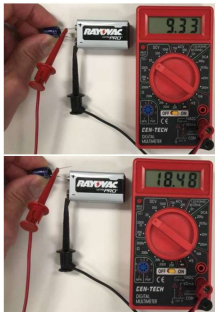
### Instrumentation and Measurements



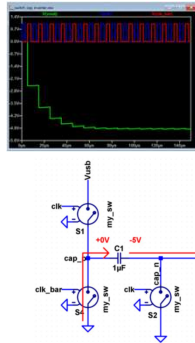
### Breadboarding and construction



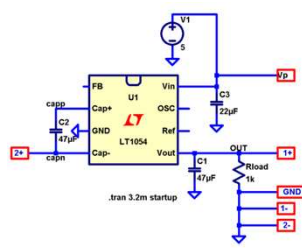
### Hands-On



### LTspice Theory



### LTspice Circuit Simulation



# ADALM2000: Instruments and Scopy GUI

Runs on Windows, Linux,  
Mac, Android

Oscilloscope

Spectrum Analyzer

Network Analyzer

Signal Generator

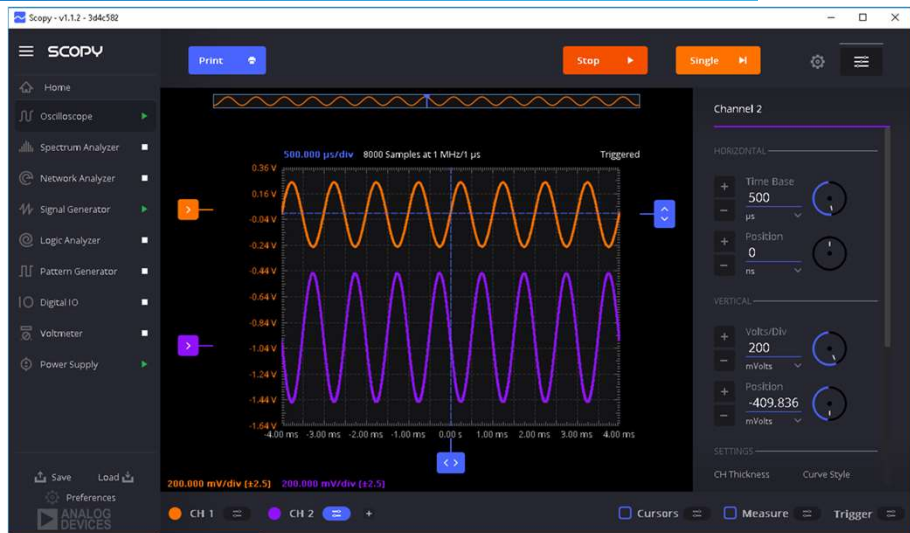
Logic Analyzer

Digital Pattern Generator

Digital I/O

Voltmeter

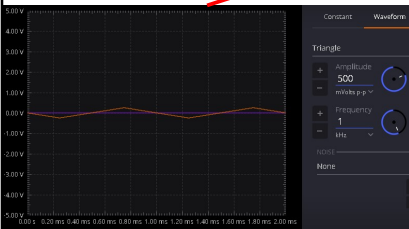
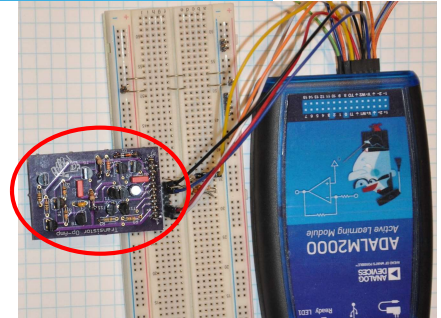
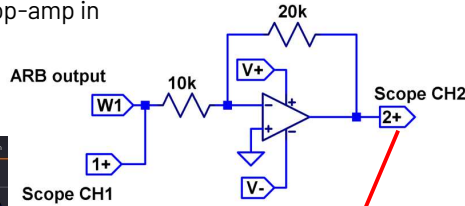
Power Supply (+/-  
5V@50mA)





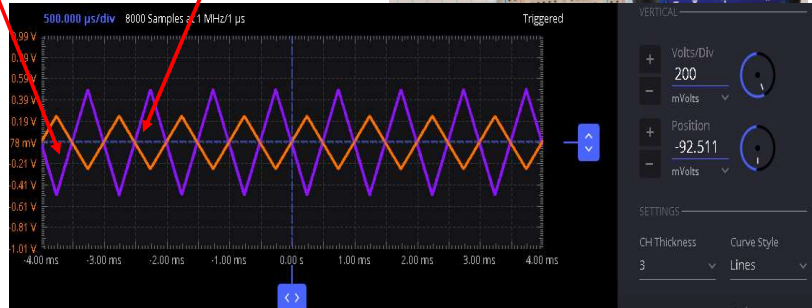
# ADALM2000 "Traditional Use"

Use scope, signal generator to test op-amp in gain of -2

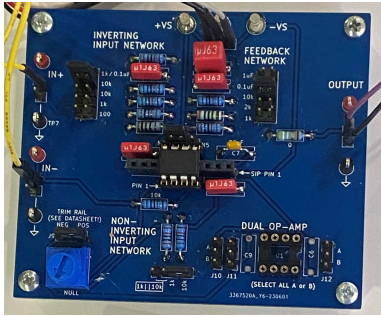


Switch over to Scopy's network analyzer to measure bandwidth, stability

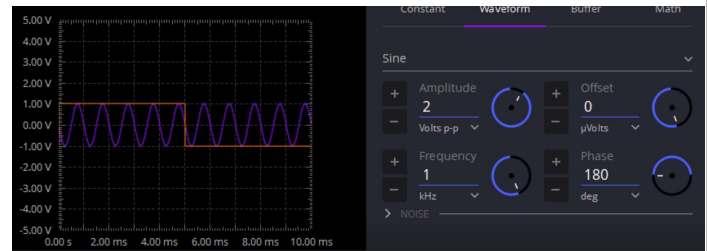
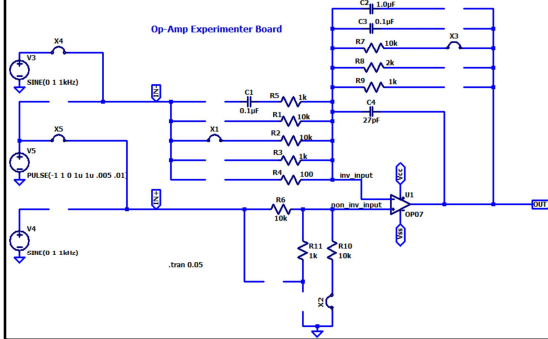
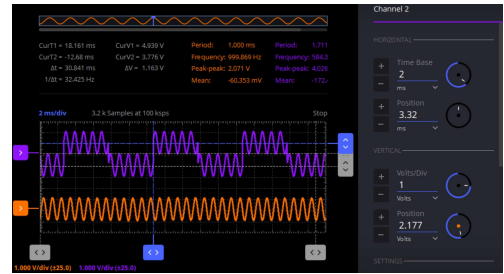
Discrete op-amp design at:  
[github.com/analogdevicesinc/education\\_tools](https://github.com/analogdevicesinc/education_tools)



# An Experiment in Breadboard Debug Mitigation



- ▶ Inverting, noninverting, difference, differentiator, integrator
- ▶ Design files on GitHub (see speaker notes below), cheaper to order yourself 😊



©2023 Analog Devices, Inc. All rights reserved.

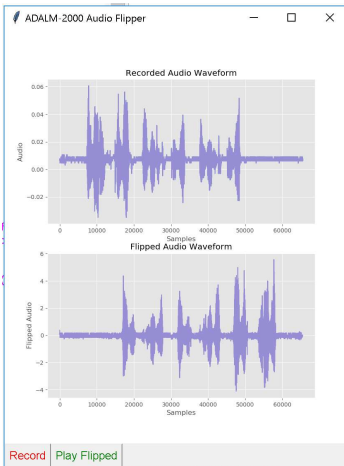
5 July 2023

18

- ▶ Op-Amp Experimenter board files: [https://github.com/mthoren-adi/education\\_tools/tree/op-amp-experimenter/experiment-boards/op\\_amp\\_experimenter](https://github.com/mthoren-adi/education_tools/tree/op-amp-experimenter/experiment-boards/op_amp_experimenter)
- ▶ Op-Amp Experimenter Ltspice sim: [https://github.com/mthoren-adi/education\\_tools/tree/op-amp-experimenter/m2k/ltspice/op\\_amp\\_experimenter](https://github.com/mthoren-adi/education_tools/tree/op-amp-experimenter/m2k/ltspice/op_amp_experimenter)
- ▶ Discrete op-amp exercise: <https://wiki.analog.com/university/courses/electronics/electronics-lab-13>
- ▶ Video: [https://www.youtube.com/watch?v=zzXmWyu\\_rVM](https://www.youtube.com/watch?v=zzXmWyu_rVM)

## Scopy getting in the way? Libm2k!

- ▶ ADALM2000 for Computer Science and DSP? You bet!
- ▶ Full-featured API, cross platform native C library
- ▶ Python and C# bindings (MATLAB possible)



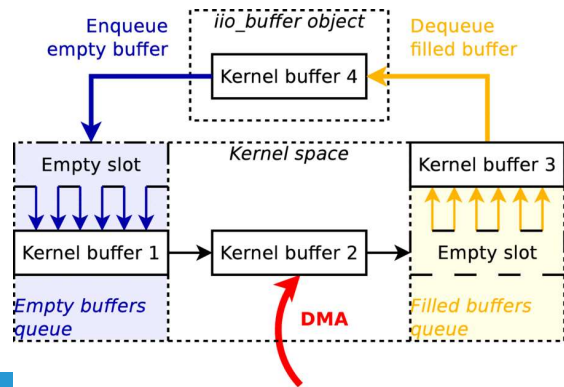
```

8 ctx=libm2k.m2kOpen()
9 ain=ctx.getAnalogIn()
10 aout=ctx.getAnalogOut()
11 ain.setSampleRate(1000000)
12 ain.setOversamplingRatio(133) # compensate for DAC sample rate
13 ain.setRange(1, -1, 1)
14 data = ain.getSamples(2**16) # grab a buffer of data
15 aout.setSampleRate(0, 7500) # Set output sample rate
16 audio = data[1]
17 dc = np.convolve(audio, (np.ones(64)/64.0), mode='same') # FIR filter
18 audio = audio - dc # DC baseline restore
19 audio[0:100] = [0]*100 # get rid of filter edge effect
20 audio = np.flip(audio) # Here's where the flip happens!!
21 audio = audio * 100.0 # add some gain
22 buffer = [audio, audio]
23 aout.push(buffer) # Play!

```

# ADALM2000 Under the Hood – Open and “Anti-NIH”, no proprietary anything

- ▶ Built on Xilinx Zynq SoC running embedded Linux (Think: Raspberry Pi/BeagleBone + custom peripherals)
- ▶ Same infrastructure used to support ADI customers
- ▶ Translation: This is **real, industrial computer science**, not a one-off science project.
- ▶ Source code provides fodder for lessons in:
  - FPGA programming
  - FIFOs/DMA
  - Computer Networking
  - Embedded systems in general
- ▶ All of this applies to ADALM-PLUTO, too.



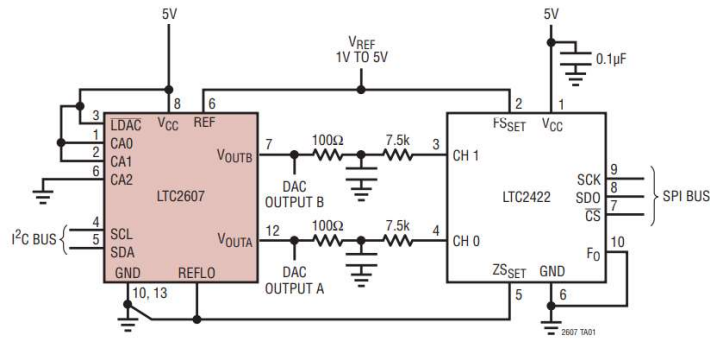
# ADALM2000 for Embedded Debug



“But I got the Arduino library from the INTERNET! Why isn't it working?”

(Have you looked at the actual bus traffic?)

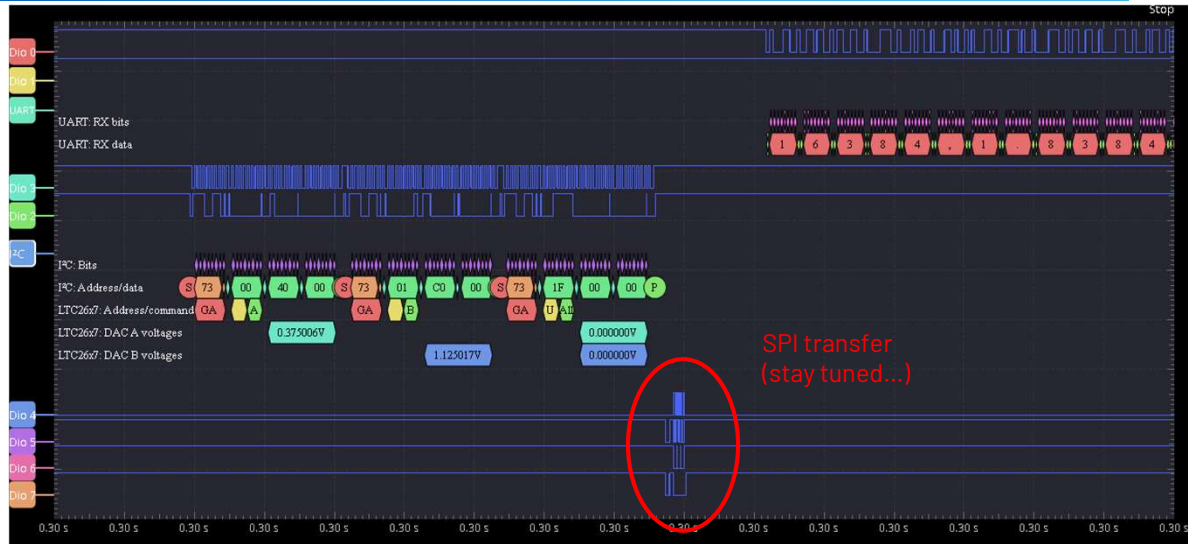
Contrived example: Arduino + LTC2607 I2C DAC, LTC2422 SPI ADC, UART



```

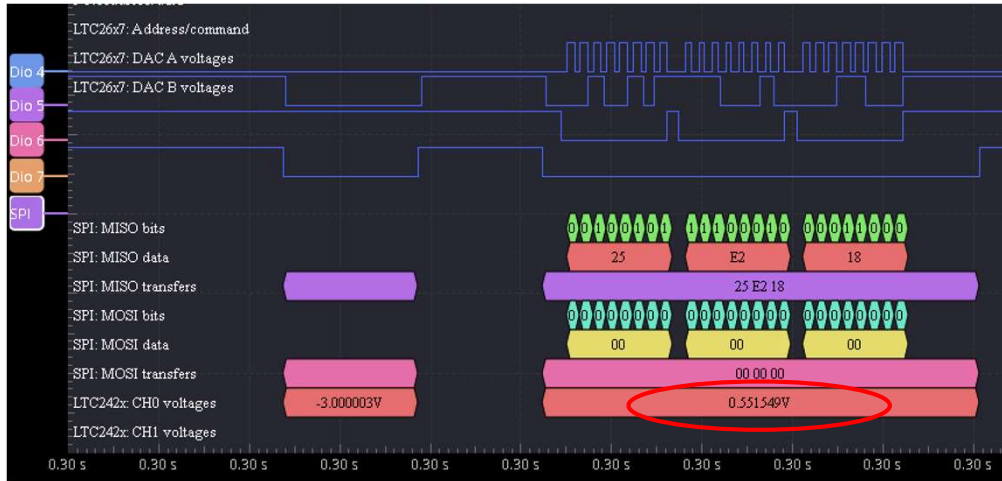
COM6
Send
16384,1.836659,16384,1.835343,Out of sync!!
16384,1.836688,16384,1.835358,Out of sync!!
16384,1.836693,16384,1.835300,Out of sync!!
16384,1.836669,16384,1.835386,Out of sync!!
16384,1.836721,16384,1.835372,Out of sync!!
16384,1.836659,16384,1.835362,Out of sync!!
16384,1.836702,16384,1.835348,Out of sync!!
16384,1.836697,16384,1.835396,Out of sync!!
16384,1.836683,16384,1.835324,Out of sync!!
16384,1.836712,16384,1.835329,Out of sync!!
16384,1.836669,16384,1.835310,Out of sync!!
16384,1.836731,16384,1.835348,Out of sync!!
16384,1.836697,
Autoscroll Carriage return 115200 baud Clear output
    
```

# Sigrok UART, I2C decoders, LTC2607 stacked decoder (upstreamed)



▶ DAC output displayed in human-readable volts

# Sigrok SPI decoder, LTC2422 stacked decoder (upstreamed)



► ADC input displayed in human-readable volts

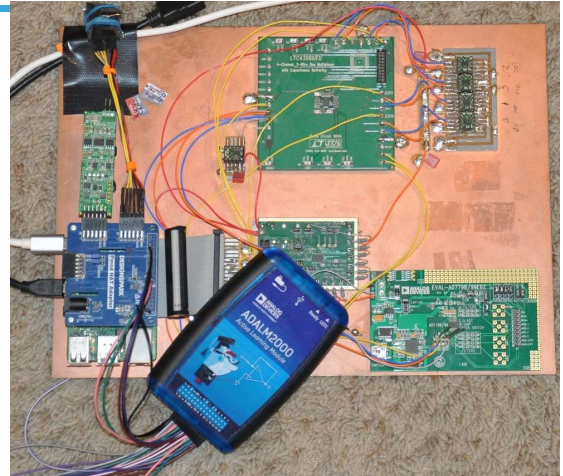


# Yes, it's usable "in real life" (Popular tool for field engineers)

- ▶ Uncovered SPI mode issue in AD5791 Linux device tree while testing with Raspberry Pi
- ▶ Zoomed right into the issue w/ M2K+Scopy

Confirmed that modes 1(spi-cpha) and 2 (spi-cpol) work, modes 0 (none) and 3 (spi-cpha, spi-cpol) don't. I suppose it's a somewhat arbitrary choice between mode 1 and 2, any objection to keeping as-is? (mode 1, spi-cpha)

Mode 1 Scopyshot:

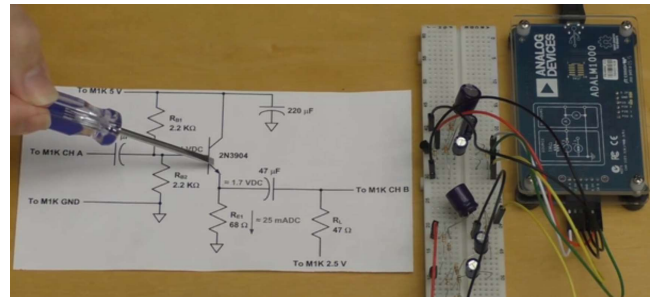
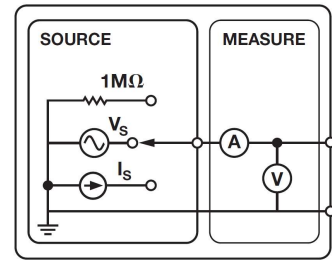


Sheet O' Copper Clad Linux Device Driver Test Bed



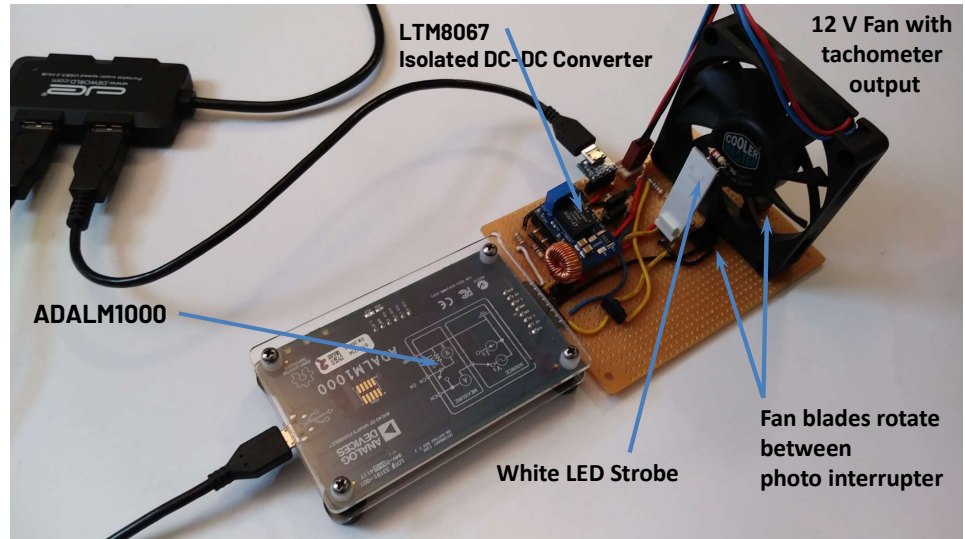
# ADALM1000: \$70 SMU for basic electronics

- ▶ Measure Voltage
- ▶ Source Voltage/Measure Current
- ▶ Source Current/Measure Voltage
- ▶ 100ksps Oscilloscope
- ▶ 100ksps Function Generator
- ▶ 16-bit accuracy
- ▶ C/C++/Python/MATLAB support



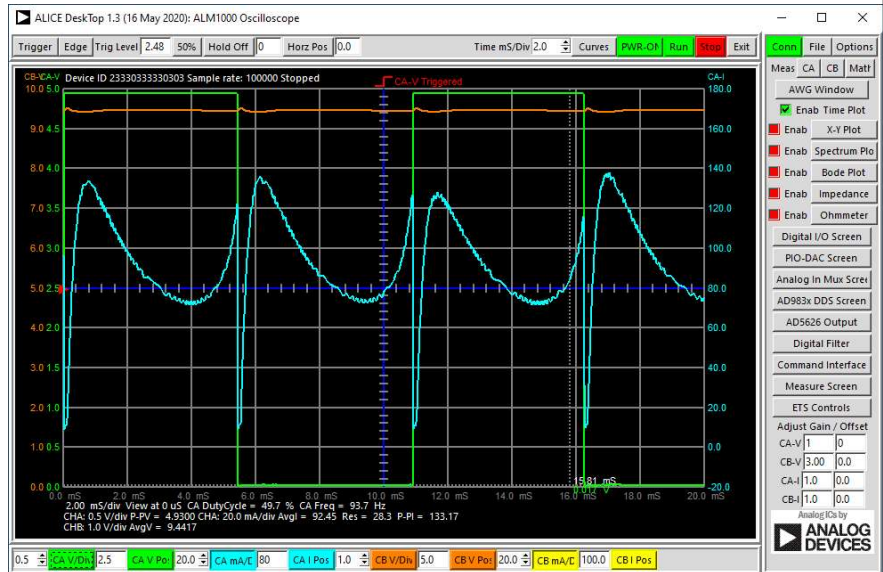
## Example: ADALM1000 Measuring Instrumented Brushless Fan Motor

DC-DC converter module input is powered from extra USB port on powered hub. DC-DC converter (adjusted to output 5 V) negative output connected to channel A output. DC-DC converter positive output powers 12 V fan. Channel B output drives LED strobe. Motor current waveform measured on channel A output, Fan Tach or blade photo interrupter (switchable) measured on AIN, Fan voltage (sum of CH A and DC-DC output) measured on BIN (3:1 resistor divider).



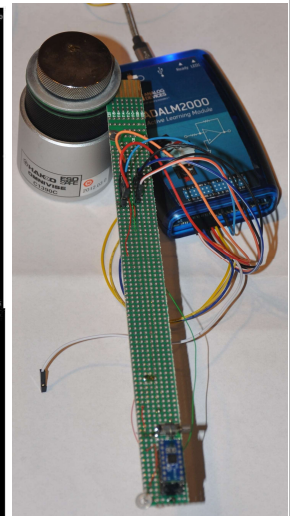
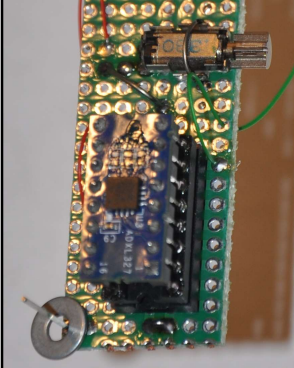
# Multiple channel trace ALICE Screen Shots

Time scale close up - Green trace is fan Tach output waveform (91Hz). Orange trace is DC voltage on Fan. Light Blue (cyan) trace is motor current, Fan Tach outputs 2 pulses per revolution. Motor current waveform shows 4 pulses of current per revolution. Narrow current spike aligned with Tach rising edge. Hall effect switch for brushless motor commutator also provides Tach output signal.



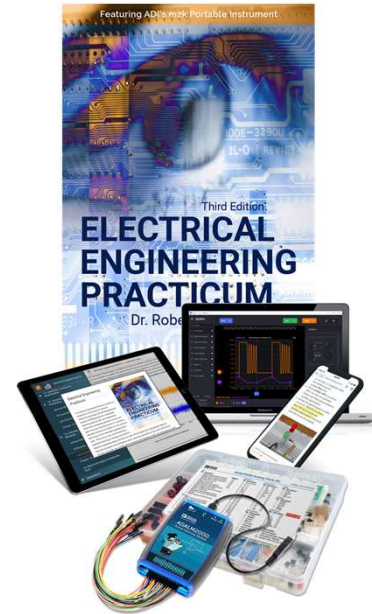
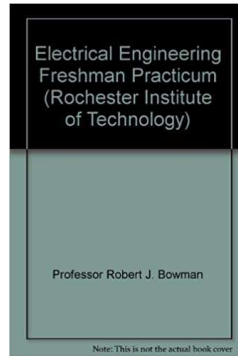
# M2K 4 MEs: Vibration Analysis

Measure Cell Phone vibrator with ADXL327  
Detect "imminent failure" from rattling washer



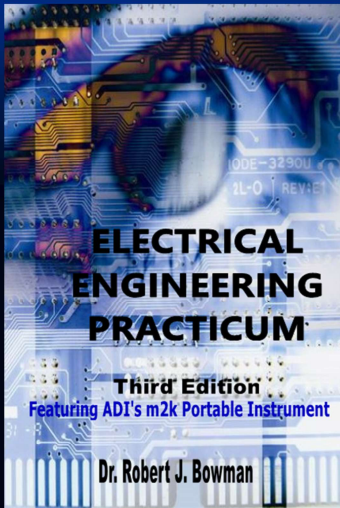
## In need of a fully developed freshman lab curriculum?

- ▶ M2k-based edition  
Developed by Dr. Robert Bowman, RIT
- ▶ First edition, 2009
- ▶ Evangelizes student-owned equipment, structured, experiential learning augmented with unstructured exploring



- ▶ <https://www.trunity.com/trubook-electrical-engineering-practicum-by-robert-j-bowman-3rd-ed.html>

# Practicum E-Text



Dr. Robert J. Bowman

Week 1 Introduction to Electrical Engineering and the Personal Test Lab **Test Instruments**

Week 2 Power Supplies and Electrical Power

Week 3 Signal Generators and Waveforms

Week 4 Resistors and Ohm's Law

Week 5 Diodes and Rectification **Circuit Elements**

Week 6 Capacitors and Time Constants

Week 7 Inductors and Resonance

Week 8 Thermal Sensors and Temperature

Week 9 Accelerometers and Tilt Sensing

Week 10 Microphones and Sound Sensing **Transducers**

Week 11 Radio Frequencies and Amplitude Modulation

Week 12 Radio Frequencies and Amplitude Demodulation **Electronic Systems**

Week 13 Amplifiers and Sound Amplification

Week 14 Infrared (IR) Emission and IR Transmitter

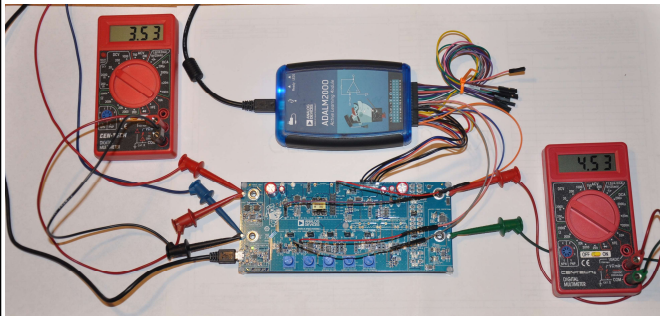
Week 15 Infrared (IR) Detection and IR Receiver

**Electrical Engineering Practicum**



## ADALM-SR1: A Semester's worth of labs in one board

- ▶ Breadboards build debugging skill (understatement of the day)
- ▶ ADALM-SR1: companion board for a family of power electronics labs
- ▶ "Medium-Cost, semester-worth-of-lab boards" might be the way of the future...
- ▶ [https://wiki.analog.com/university/tools/lab\\_hw/adalm-sr1](https://wiki.analog.com/university/tools/lab_hw/adalm-sr1)



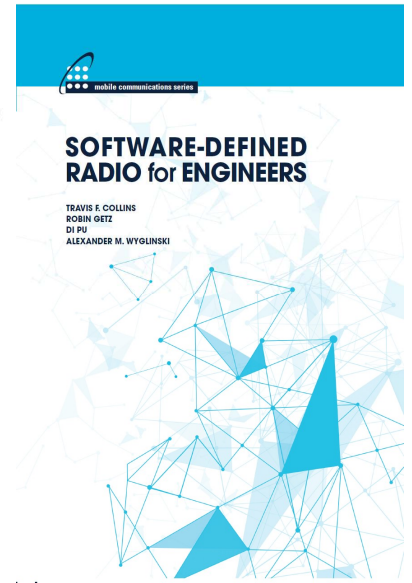
Self-contained – no extra  
equipment required)

## Low-Cost SDR module and FREE SDR Textbook

- ▶ Practical textbook for teaching communications theory based on PlutoSDR complete with:
  - Labs:
    - PlutoSDR
    - Carrier and timing synchronization
    - Frame recovery
    - Channel estimation/equalization
    - OFDM
  - Lecture Slides
  - Chapter questions
  - MATLAB code
- ▶ Published by Artech House and made available through Analog Devices
- ▶ Designed for junior, senior, or early graduates



\$230





# PlutoSDR is a general-purpose SDR tool

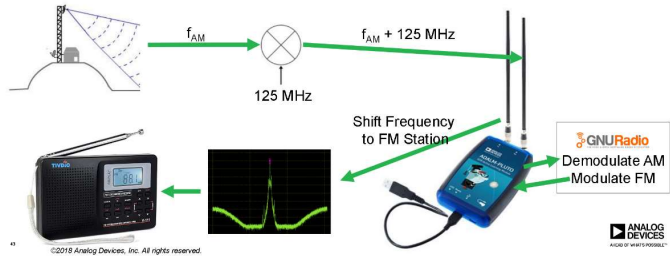
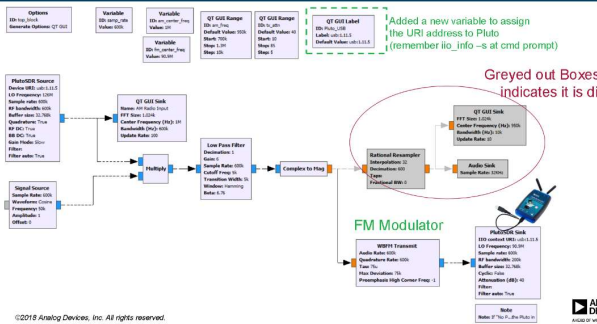
## AM on an FM Radio using GNURadio: An excellent intro to radio concepts

<https://github.com/jonkraft/>

### Pluto Lab 1: Play an AM Station on an FM Radio

- For this lab, we'll convert an AM station to FM and send it to an FM station that we choose
- AM is around 1000kHz. That's too low for Pluto, so we use a mixer to shift up by 125MHz

#### Lab1: GNU Radio Flow Chart

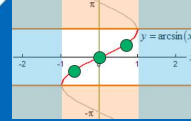


# Phased Array Learning Platform

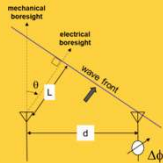
Digitizer



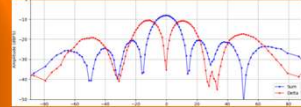
Antenna Impairments



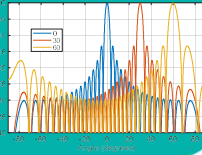
Steering Angle



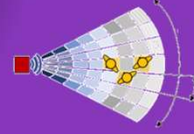
Monopulse Tracking



Antenna Patterns

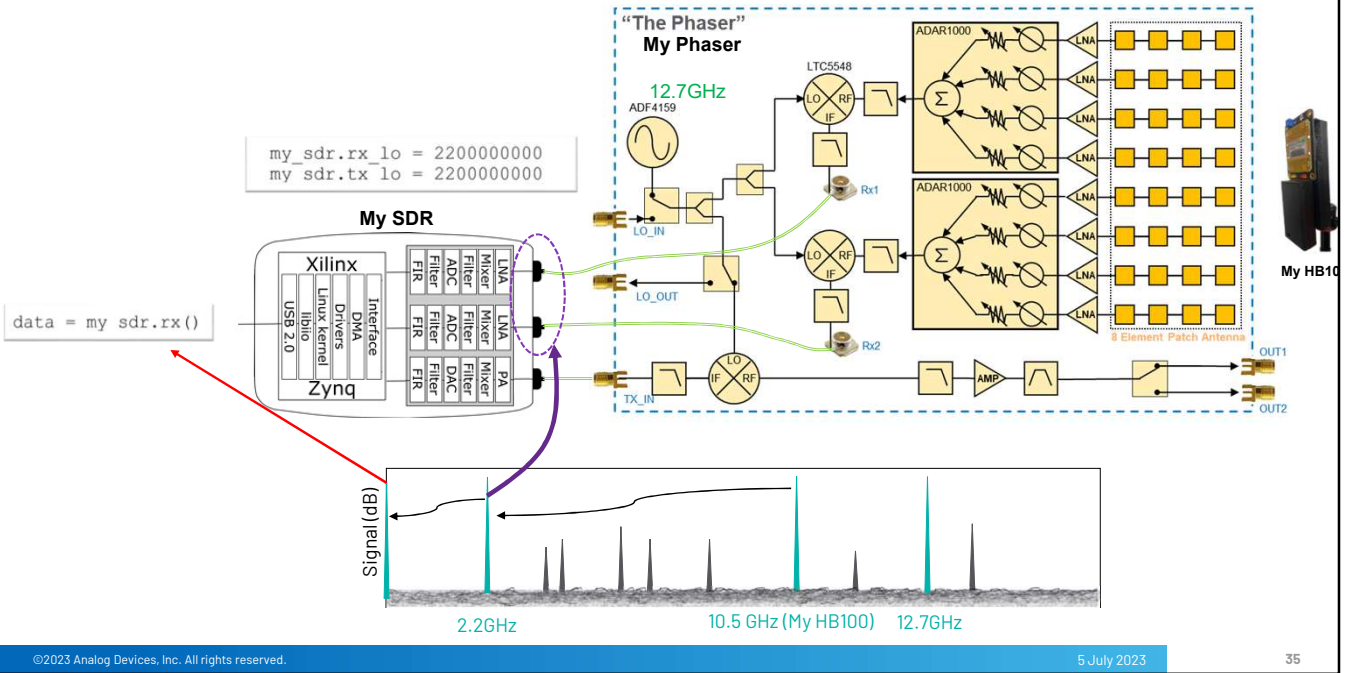


Radar



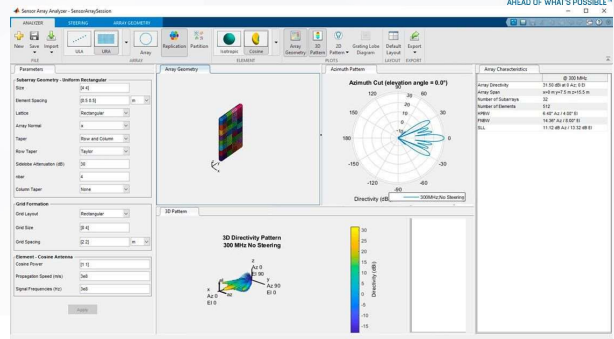
► [Wiki.analog.com/phaser](http://Wiki.analog.com/phaser)

# Microwaves to RF to DSP in Python or MATLAB

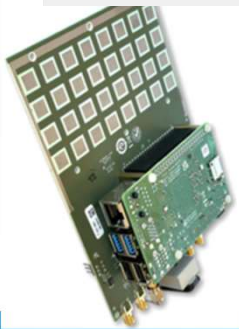


# Curriculum Goals

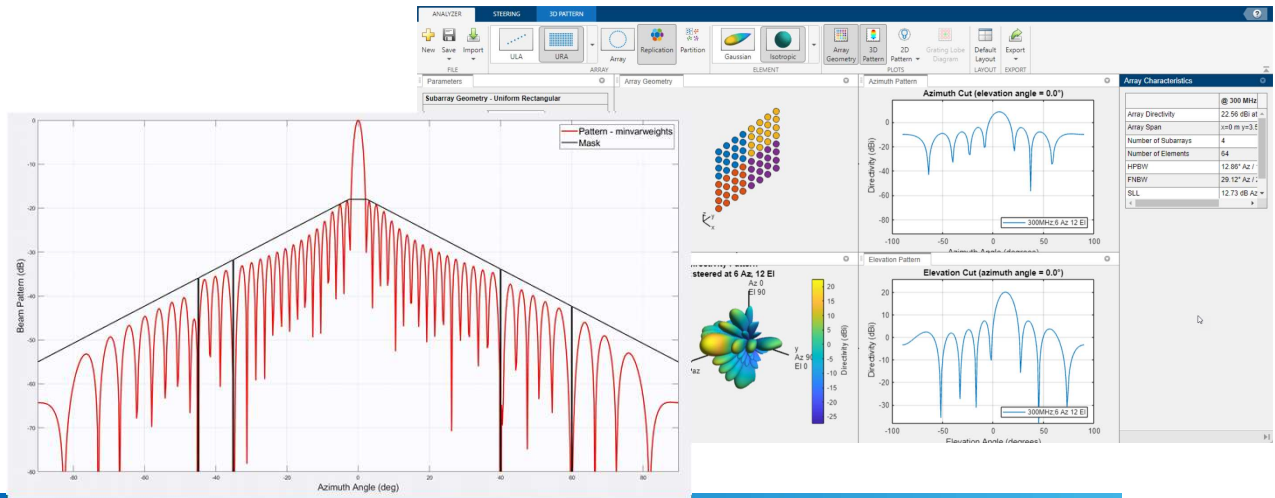
1. Gain an understanding of phased array and beamforming concepts
2. Learn how simulation models can be used to predict array and beam behavior for system design and test
3. Validate simulation models using prototype hardware
4. Learn about practical applications for phased array systems



The screenshot displays the ANSYS HFSS simulation environment. The top panel shows the 'Array Geometry' with a 3D model of a rectangular array. The middle panel shows '2D Directivity Patterns' for a '300 MHz No Steering' configuration, with a color scale for Directivity (dB) ranging from -15 to 30. The bottom panel shows '3D Directivity Patterns' and two plots: 'Radiation Coefficient' vs. 'Azimuth Angle (degrees)' and 'Normalized Gain' vs. 'Azimuth Angle (degrees)'. The radiation coefficient plot shows a main lobe centered at 0 degrees, with side lobes. The normalized gain plot shows a similar pattern. The ANSYS logo and 'AHEAD OF WHAT'S POSSIBLE™' tagline are visible in the top right corner of the software interface.



# MATLAB Goal: Establish a Common Design Language and Development Framework



©2023 Analog Devices, Inc. All rights reserved.



- ▶ This is the light cover slide option.

## What can you buy and do with >\$65?



- ▶ Buy a Whiskey Kit
- ▶ Get Drunk?
- ▶ Maybe impress friends with B/Spoke setup



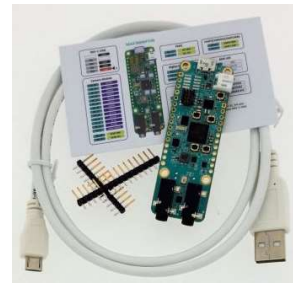
- ▶ Buy a Lemon tree
- ▶ Make lemonade!
- ▶ (much more altruistic)



- ▶ Buy an Uber from Evergreen Museum to here...
- ▶ Speak at ASEE events

## Or... Learn how to use AI to impact the world!

- ▶ A low-cost learning kit focusing on combining embedded and digital worlds
  - Collaboration with Elektor
- ▶ Includes MAX78000 development board & required cables
  - Can be purchased separately for **\$28.36**
- ▶ Includes detailed instruction-based workbook
- ▶ Ideal for learning IoT, Edge applications
- ▶ Reasonably Priced at \$64 for the package
  - Board by itself: \$29
  - E-book: \$29.81





## Key Benefits

- ▶ Great entry point for working with HW + SW + AI using ADI's award winning micro controller
- ▶ The package is great for all learning levels
- ▶ Core HW can be expanded and used to develop real-world solutions
  - UT Austin: Student developed Recycler object detection application to sort glass, aluminum, & plastic
  - Univ. ETH Zurich: Using it to monitor parking spaces for the city as opposed to using costly sensor installs
- ▶ ADI as an educational partner
  - We are investing in education through our Active Learning Program
  - Great complimentary offering of components and measurement equipment

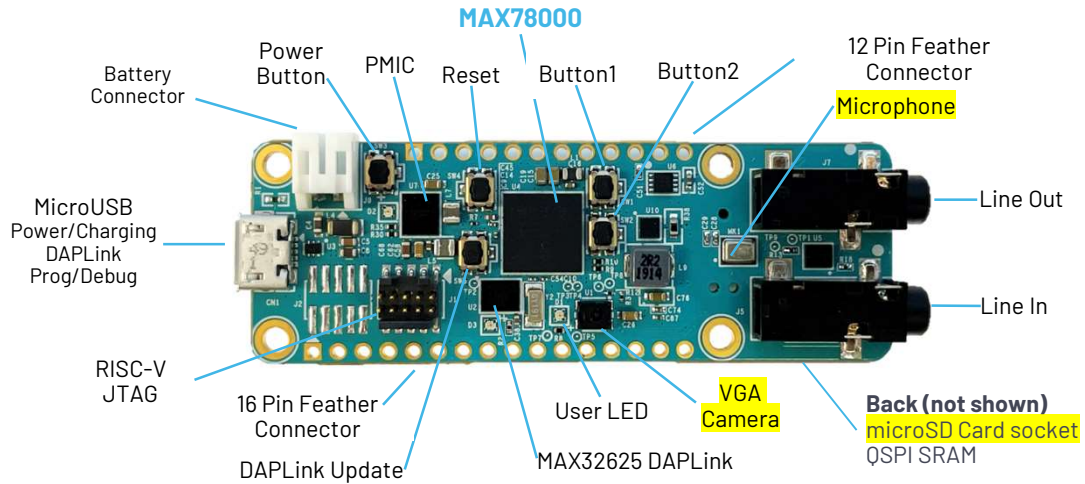
- ▶ Delete all instruction slides prior to use and any of the example slides that are not needed.

## Elektor HOW2 Package – Workbook Contents

- ▶ Tons of great step by step learning modules each concluding with an exercise(s).
  - Creating a C program
  - Hardware projects, "Blink a light", Voice guided LED blinking
  - Modules to learn about ADCs, UART, I2C, SPI, Timers, Pulse Train Engine, TRNG, OWM, I2S, Camera, TFT Display, & more!
  - Using the AI Micro: Play with a Convolutional Neural Network (CNN) on embedded HW and learn how to:
    - Training new keywords (audio, visual)
    - Modeling and synthesis
- ▶ Provides
  - Coding examples and references
  - Core learning concepts
  - flows perfectly for instruction or self-learning




## Featured board: MAX78000FTHR



# MAX78000 AI Micro - System-on-Chip

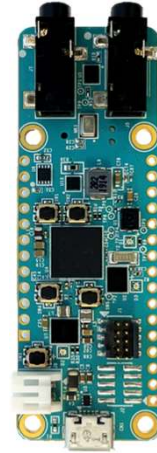
► Get to work with ARM M4/RISC-V Co-Processors

<p><b>Memory</b></p> <ul style="list-style-type: none"> <li>Flash 512 KB</li> <li>SRAM 128 KB</li> </ul>	<p><b>Ultra Low Power Micro</b></p> <ul style="list-style-type: none"> <li>ARM Cortex-M4F 100 MHz</li> <li>Cache</li> <li>RISC-V</li> </ul>	<p><b>CNN Accelerator</b></p> <p>64 processors</p>  <p>442 KB Kernel Memory 384 KB SRAM Cache 512 KB Data Memory 32 Model Layer Registers</p>
<p><b>Timers</b></p> <ul style="list-style-type: none"> <li>3 × Timer</li> <li>Watchdog</li> </ul>	<p><b>External Interfaces</b></p> <ul style="list-style-type: none"> <li>2 × Quad SPI</li> <li>UART, 2 × I2C, ADC, I2S</li> <li>Parallel Camera</li> </ul>	
<p><b>Security</b></p> <ul style="list-style-type: none"> <li>AES</li> <li>TRNG</li> </ul>	<p><b>Power</b></p> <ul style="list-style-type: none"> <li>SIMO</li> </ul>	

## MAX78000FTHR- Pre-loaded to go!

When connected to a PC:

- ▶ Recognized as composite USB device
- ▶ DAPLINK, WebUSB, CMSIS, Mass storage device, & Serial Port
- ▶ Pre-loaded with audio keyword spotting demo which students can manipulate in later exercises
  - Recognizes 0-9, Go, Stop, Left, Right, Up, Down, On, Off
  - Once "Go" starts program, LED will blink number spoken
- ▶ Free SDK (instructions in workbook)
  - GCC Toolchains for ARM and RISC-V processors
  - Eclipse ICD (C/C++)
  - MinGW





# Direct from the workbook – Pinout Descriptions

How2: Get Started with the MAX78000FTHR Board

**SWS:** DAPLink adapter button. Keep this button pressed while applying power to the board to put the MAX32625 DAPLink adapter onboard to MAINTENANCE mode for DAPLink firmware updates.

The functions of the LEDs are:

**D1:** Connected to the MAX78000 GPIO ports. This LED can be controlled by user firmware.

**Port 2\_0 : Red**  
**Port 2\_1 : Green**  
**Port 2\_2 : Blue**

**D2:** Connected to MAX20303 I<sup>2</sup>C LEDs outputs. These LEDs can be controlled through I<sup>2</sup>C commands. They also can be configured as charge status indicators by issuing I<sup>2</sup>C commands.

**D3:** DAPLink adapter MAX32625 status LED. Controlled by the DAPLink adapter and cannot be used as a user LED.

### 1.4 • GPIO Pinout

There are two headers at either side of the board where GPIO signals are terminated. As shown in Figure 1.3, **Header 34** has 12 pins and **Header 38** has 16 pins. The board includes the following GPIO port pins:

**PORT0:** P0\_5, P0\_6, P0\_7, P0\_8, P0\_9, P0\_11, P0\_16, P0\_17, P0\_19  
**PORT1:** P1\_0, P1\_1  
**PORT2:** P2\_3, P2\_4, P2\_6, P2\_7  
**PORT3:** P3\_1

Tables 1.1 and 1.2 show the 34 and 38 pin names and their descriptions. Notice some pins are shared. For example, P1\_0 and p1\_1 are shared with the UART RX and TX pins respectively. Similarly, P2\_3 and P2\_4 are shared with analog inputs AIN3 and AIN4 respectively.

Chapter 1 • The MAX78000FTHR Development Board

PN	NAME	DESCRIPTION
1	SVS	SVS Switched Connector to the Battery. This is the primary system power supply and automatically switches between the battery voltage and the USB supply when available.
2	PWR	Turns off the PMIC if shorted to Ground for 15 seconds. Hard power-down button.
3	VBUS	USB VBUS Signal. This can be used as a 5V supply when connected to USB. This pin can also be used as an input to power the board, but this should only be done when not using the USB connector since there is no circuitry to prevent current from flowing back into the USB connector.
4	P1_8	GPIO
5	MPC3	GPIO controlled by PMIC through the PC interface. Open drain or push-pull programmable.
6	P0_9	GPIO or QSPI/ SDIO signal. Shared with SD card and on-board QSPI SRAM.
7	P0_8	GPIO or QSPI/ SDIO signal. Shared with SD card and on-board QSPI SRAM.
8	P0_11	LPUART or QSPI/ save select signal.
9	P0_19	GPIO
10	P3_1	GPIO or Wake-up signal. This pin is 5.1V only.
11	P0_16	GPIO or I <sup>2</sup> C SCL signal. An on-board level shifter allows selecting 1.8V or 3.3V operation through RT0 or RT0 resistors. Do not populate both.
12	P0_17	GPIO or I <sup>2</sup> C SDA signal. An on-board level shifter allows selecting 1.8V or 3.3V operation through RT0 or RT0 resistors. Do not populate both.

Table 1.1 Header 34 pins

PN	NAME	DESCRIPTION
1	RST	Master Reset Signal
2	3V3	3.3V Output. Typically used to provide 3.3V to peripherals connected to the expansion headers.
3	1V8	1.8V Output. Typically used to provide 1.8V to peripherals connected to the expansion headers.
4	GND	Ground
5	P2_3	GPIO or Analog Input (AIN3 channel)
6	P2_4	GPIO or Analog Input (AIN4 channel)
7	P1_1	GPIO or UART2 Tx signal
8	P1_0	GPIO or UART2 Rx signal
9	MPC1	GPIO controlled by PMIC through PC interface. Open drain or push-pull programmable
10	MPC2	GPIO controlled by PMIC through PC interface. Open drain or push-pull programmable
11	P0_7	GPIO or QSPI/ clock signal. Shared with SD card and on-board QSPI SRAM
12	P0_5	GPIO or QSPI/ MOS signal. Shared with SD card and on-board QSPI SRAM
13	P0_6	GPIO or QSPI/ MISO signal. Shared with SD card and on-board QSPI SRAM
14	P2_6	GPIO or LPUART Rx signal
15	P2_7	GPIO or LPUART Tx signal
16	GND	Ground

Table 1.2 Header 38 pins

• 16

• 17

# Direct from the workbook – oodles of guided projects!

How2: Get Started with the MAX7800FTHR Board

- 2.2.19 • Security (AES) ..... 29
- 2.2.20 • True Random Number Generator (TRNG) Non-Deterministic Random Bit Generator (NDRBG) ..... 29
- 2.2.21 • CRC ..... 29
- 2.2.22 • Bootloader ..... 30
- 2.2.23 • Device Resets ..... 30
- 2.2.24 • Interrupts and Exceptions ..... 30

Chapter 3 • Beginning with the MAX7800FTHR Development Board ..... 31

- 3.1 • Overview ..... 31
- 3.2 • Installing the Eclipse MaximsDK Software ..... 31
- 3.3 • Using the Eclipse MaximsDK – example MAX7800FTHR program ..... 32
- 3.4 • Project 1 – Creating a C program – display message ..... 41

Chapter 4 • Simple MAX7800FTHR Hardware Projects ..... 46

- 4.1 • Overview ..... 46
- 4.2 • Project 1 – External flashing LED (+1.8V output port voltage) ..... 46
- 4.3 • Project 2 – Alternately flashing LEDs (+1.8V output port voltage) ..... 50
- 4.4 • Project 3 – Alternately flashing LEDs (+3.3V output port voltage) ..... 53
- 4.5 • Project 4 – Rotating LEDs – same port pins ..... 55
- 4.6 • Project 5 – Rotating LEDs – different port pins ..... 58
- 4.7 • Project 6 – Binary up counter with LEDs ..... 60
- 4.8 • Project 7 – Random flashing LEDs ..... 63
- 4.9 • Project 8 – Push button and LED – using an on-board button ..... 65
- 4.10 • Project 9 – Two pushbuttons and two LEDs – using external buttons ..... 67
- 4.11 • Project 10 – Using an external button – external interrupts ..... 71
- 4.12 • Using LCDs ..... 73
- 4.13 • Project 11 – LCD seconds counter ..... 76

Chapter 5 • Analog-To-Digital converters (ADC) ..... 89

- 5.1 • Overview ..... 89
- 5.2 • Project 1 – Voltmeter with LCD ..... 89
- 5.3 • Project 2 – Temperature measurement ..... 94
- 5.4 • Project 3 – ON/OFF temperature controller ..... 97
- 5.5 • Project 4 – ADC with completion interrupt – displaying the temperature ..... 101

Chapter 6 • Serial Communication – UART ..... 104

- 6.1 • Overview ..... 104
- 6.2 • MAX7800FTHR UART serial ports ..... 106
- 6.3 • Project 1 – Sending the temperature readings to a terminal with relative time stamping ..... 108
- 6.4 • Project 2 – Calculator project using a terminal ..... 111
- 6.5 • Project 3 – MAX7800FTHR and Arduino Uno serial communication ..... 114
- 6.6 • Project 4 – UART interrupts ..... 118

• 10

• Table of Contents

Chapter 7 • I2C Bus Interface ..... 121

- 7.1 • Overview ..... 121
- 7.2 • The I2C Bus ..... 121
- 7.3 • I2C pins of the MAX78000 microcontroller ..... 122
- 7.4 • Project 1 – I2C port expander ..... 123
- 7.5 • Project 2 – TMP102 temperature sensor ..... 128

Chapter 8 • SPI Bus Interface ..... 136

- 8.1 • Overview ..... 136
- 8.2 • MAX78000 microcontroller SPI ports ..... 137
- 8.3 • Project 1 – SPI send/receive ..... 137

Chapter 9 • Timers ..... 142

- 9.1 • Overview ..... 142
- 9.2 • Timer operation ..... 143
- 9.3 • 32-bit single/cascade and dual (6-bit) modes ..... 143
- 9.4 • Project 1 – Time delay – using a one-shot timer (monostable) ..... 143
- 9.5 • Project 2 – Continuously running timer (astable) ..... 149
- 9.6 • Project 3 – Refreshing a 2-digit 7-segment display – seconds counter ..... 153
- 9.7 • Project 4 – Refreshing a 4-digit 7-segment display – seconds counter ..... 165
- 9.8 • Pulse width modulation (PWM) ..... 171
- 9.8.1 • MAX78000 PWM ..... 173
- 9.9 • Project 5 – Pulse width modulation (PWM) – generating 10 kHz square wave ..... 175
- 9.10 • Project 6 – Pulse width modulation (PWM) – changing the brightness of an LED ..... 177
- 9.11 • Project 7 – Pulse width modulation (PWM) – brushed DC motor speed control ..... 180

Chapter 10 • Pulse Train Engine (PT) ..... 184

- 10.1 • Overview ..... 184
- 10.2 • Project 1 – Generate a pulse train with a specified sequence ..... 184
- 10.3 • Project 2 – Generate a pulse train with a specified frequency ..... 186

Chapter 11 • True Random Number Generator Engine (TRNG) ..... 189

- 11.1 • Overview ..... 189
- 11.2 • Project 1 – Generate random numbers ..... 189

Chapter 12 • 1-Wire Master (OWM) ..... 192

- 12.1 • Overview ..... 192
- 12.2 • MAX78000 microcontroller 1-Wire pins ..... 192
- 12.3 • Project 1 – DS1820 1-Wire digital thermometer ..... 192

Chapter 13 • I2S Bus Interface ..... 200

- 13.1 • Overview ..... 200
- 13.2 • MAX78000 microcontroller I2S support ..... 201
- 13.3 • MAX78000 microcontroller I2S pins ..... 201
- 13.4 • Project 1 – I2S Bus – receiving microphone data ..... 202

• 11



## Several pre-trained models to manipulate and learn on!

```
// Common Demo Name
#define DEMO_STRING_SIZE          15
#define FACEID_DEMO_NAME         "FaceID"
#define UNET_DEMO_NAME           "UNet"
#define WILDLIFE_DEMO_NAME       "WildLife"
#define CATSDOGS_DEMO_NAME       "CatsDogs"
#define IMAGECAPTURE_DEMO_NAME   "Image Capture"
```

- ▶ Capable of video, audio, and sensory AI applications
- ▶ Downloadable content and app notes on GitHub
- ▶ Only limitation is creativity and time..... well, and the model size 😊

## Links and other info

- ▶ Links – Take a picture!
- ▶ MAX AI Github: [GitHub - MaximIntegratedAI/refdes](#)
- ▶ Visit Analog.com and search “MAX78000FTHR”
- ▶ **Elektor Kit:** <https://www.elektor.com/get-started-with-the-max78000fthr-bundle>
- ▶ Email: [Nathan.frey@analog.com](mailto:Nathan.frey@analog.com) Happy to provide more info!

- ▶ Delete all instruction slides prior to use and any of the example slides that are not needed.

Thank You!

Questions?

Email:

[Samantha.Fontaine@analog.com](mailto:Samantha.Fontaine@analog.com)

[mark.thoren@analog.com](mailto:mark.thoren@analog.com)

[Nathan.frey@analog.com](mailto:Nathan.frey@analog.com)