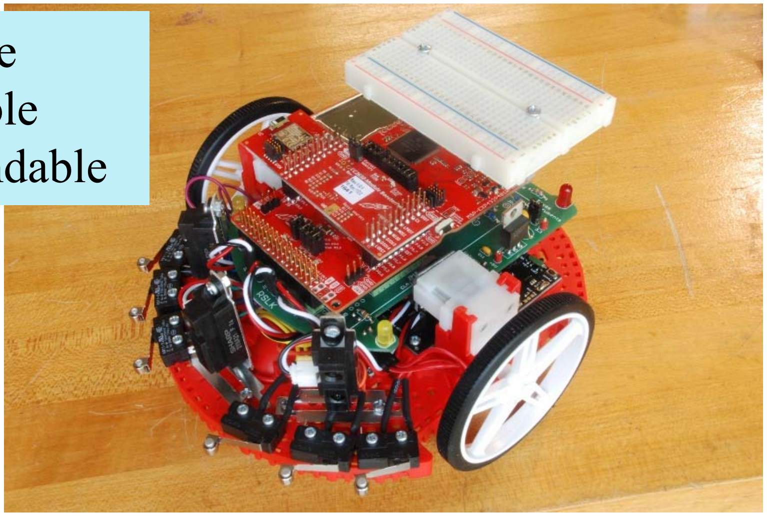
Robot Systems Learning Kit

Learn. Build. Compete.

- Mark Easley, measley@ti.com
- Jonathan Valvano, valvano@mail.utexas.edu
- Daniel Valvano
- Ayesha Mayhugh
- Franklin Cooper Jr.
- Jason Rubadue

- Simple
- Flexible
- Expandable



Outline: Lunch and Learn Session

A. Embedded Systems

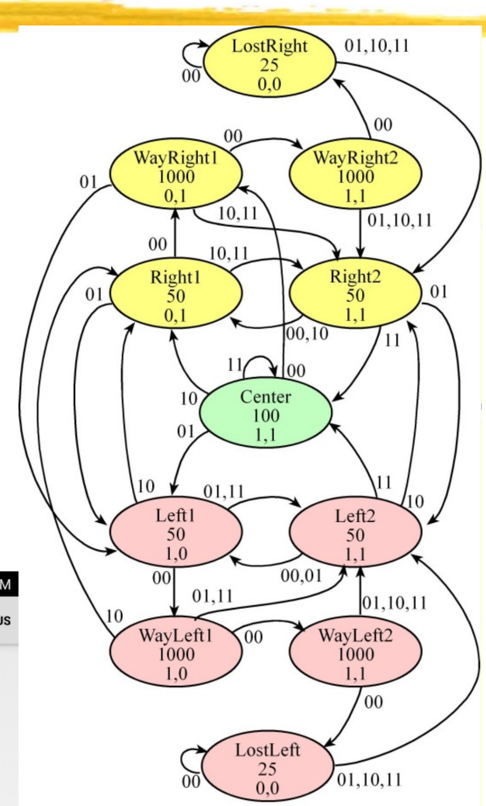
- Books, labs, competitions, MOOCs

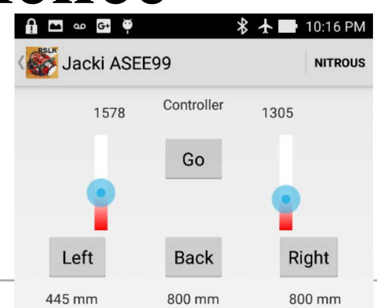
B. Robot Systems Learning Kit

- Curriculum, hardware, software, tools, CCS
- Finite State Machine, Motor, logic analyzer

C. RSLK Hands on Experience

- Closed loop controller
- Bluetooth Low Energy





Download and install Android app **http://users.ece.utexas.edu/~valvano/android**

A. Courses, Books, and Labs

Introduction to Embedded Systems

Freshman, MOOC

System Level Design

Junior, PCB, IoT

Real-time Operating Systems

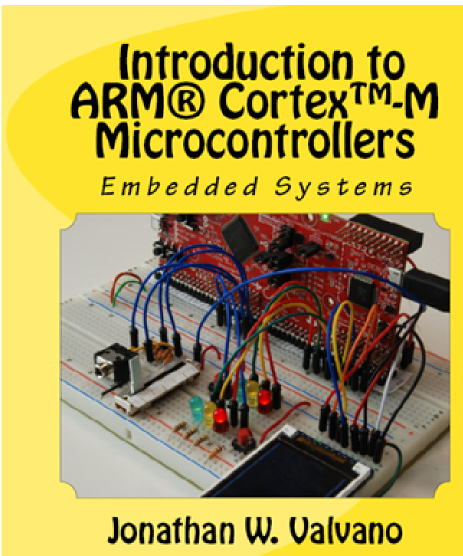
Senior/grad ,CAN, Robot

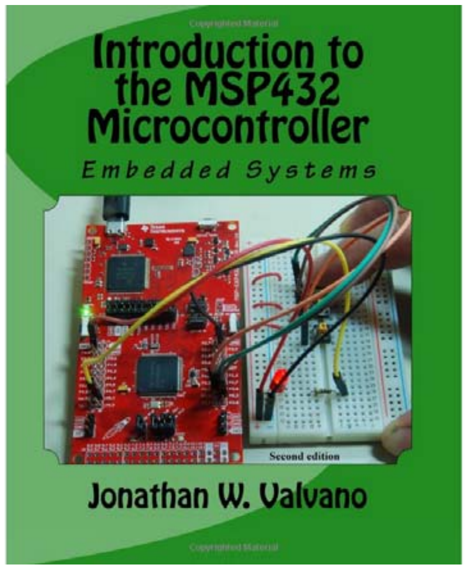
- Cortex-M4
- serial, SPI, ADC,
- timer, PWM, DMA
- interrupt controller
- JTAG debugger
- floating point

EK-TM4C123GXL, 43 I/O pins, 32k RAM, 256k RAM, 80 MHz, USB, CAN $13
MSP-EXP432E401Y, 90 I/O pins, 256k RAM, 1M ROM, 120 MHz, Ethernet $20
MSP-EXP432P401R, 67 I/O pins, 64k RAM, 256k ROM, 48 MHz, low power $13

A. Introduction to Embedded Systems

Volume 1 (freshmen EE or BME) 11,000 sold, 507 pages, $38



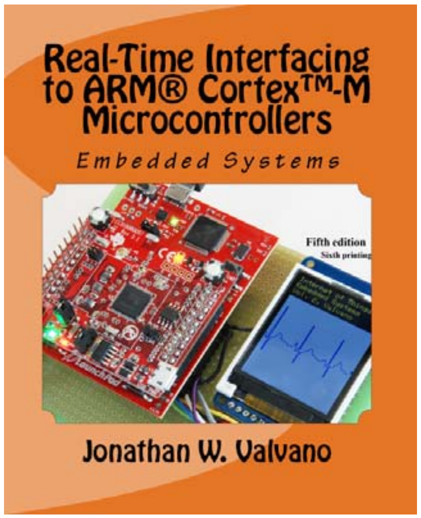


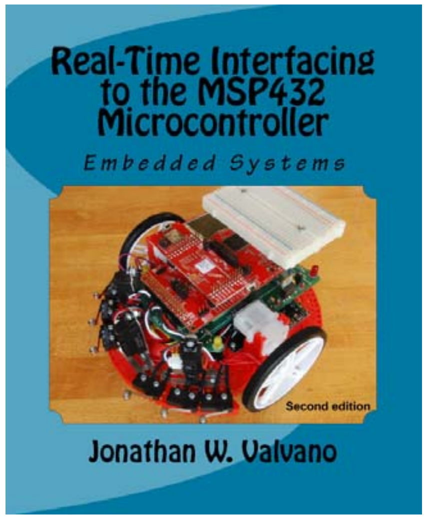
- Switches and LEDs
- Design and Debugging
- Finite State Machine
- Interrupts, data flow
- DAC output, ADC input
- LCD graphics interface
- Fixed-point
- UART and distributed
- Capstone system design
  - video game, RSLK basic kit

- Assembly or C programming
- Design and Debugging
  - Simulation, logic analyzer, oscilloscope

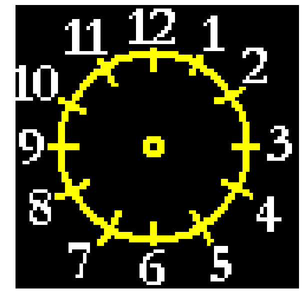
A. Interfacing and Systems

Volume 2 (junior EE) 5,000 sold, 600 pages, $40





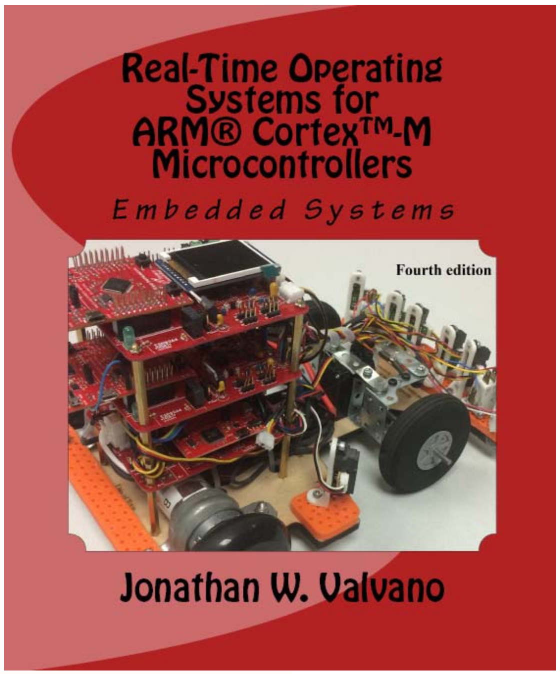
- Graphics device driver
- Alarm clock
- Stepper or DC motor
- Music player
- Temperature acquisition
- Wifi and IoT
- Capstone design
  - open ended, PCB, Power
  - RSLK advanced kit



- Hardware/software debugging
- Design and debugging
- Analog, power, computer

A. Real-Time Operating Systems

Volume 3 (senior/grad EE) 3,600 sold, 485 pages, $40



- Memory manager, device driver
- Thread switching RTOS
- Blocking semaphores
- Priority scheduler
- Digital and analog filters, FFT
- Digital control systems
- File system
- CAN or Ethernet network
- Autonomous robot racing

- TM4C
- MSP432

MOOC running on edX

A. Competition

Volume 1 (freshmen EE or BME)

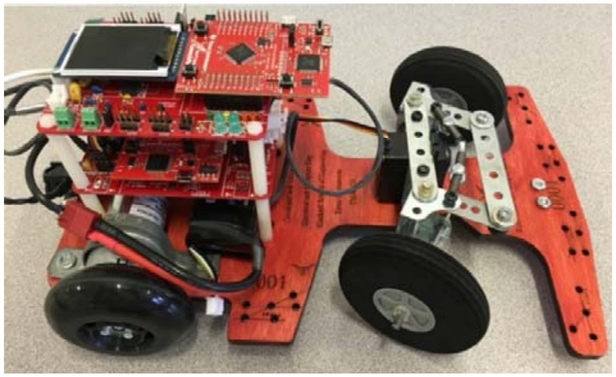
- Handheld game, peer review

Volume 2 (junior EE)

- Design cycle, testing, systems

Volume 3 (senior/grad EE)

- Autonomous Robot Racing



http://youtu.be/QxDQUUDStOw
http://youtu.be/z6_jlM2Y5qI
http://youtu.be/K9FD50qpGwg
http://youtu.be/bZ1fXtN1T08
http://youtu.be/GKctvlvprAQ

A. Embedded Systems – Shape the World

- What is and isn't a MOOC?
  - Spring 14, 15, 16, 17, 18
  - over 120,000 enrolled
  - over 3 million video hits
  - over 7% got certificates
  - 2/3 who started, finished
  - 95% approval rate

Physical kit increased completion rates

- Introduction, I/O, C programming
- Interrupt-driven interfacing, wifi
- Real-time operating systems, BLE

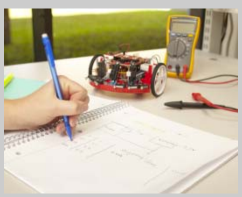
- Lab kit + Teaching videos
- LaunchPad simulator, graders, voltmeter, scope
- IoT: wifi and BLE in MOOC

B. TI Robotics System Learning Kit

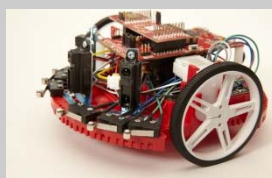
www.ti.com/rslk

Learn, Build & Compete

- **Comprehensive**, **modular,** curriculum to teach embedded systems and applications
- Accompanied by a robotics hardware kit that provides an active learning experience
- Courseware will guide you to **design, build, code** and finally **test** your own robotic system
- Collaborate to compete and develop **problem solving skills, systems knowledge** and **creativity**



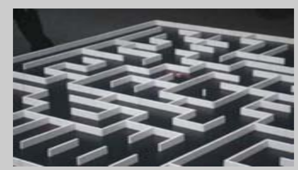
Curriculum



Hardware

Code Composer™ Studio

Software



Projects

TI Training
training.ti.com

TI Resources

B. *RSLK:* Module topics

Basic | Advanced

- 1 Code Composer Studio IDE (Basic)
- 2 Voltage, Current, and Power (Basic)
- 3 ARM Architecture (Assembly Program) (Basic)
- 4: Software Design using MSP432 (Basic)
- 5: Battery and Voltage Regulation (Basic)
- 6 General Purpose Input Output Ports (Basic)
- 7 Finite State Machines (Basic)
- 8 LEDs and Switches (Basic)
- 9 SysTick Timer (Basic)
- 10 Debugging Real-time Systems (Basic)
- 11 LCD Interface (Basic)
- 12 DC Motors (Basic)
- 13 PWM and Periodic interrupts (Basic)
- 14 I/O Triggered Interrupts (Basic)
- 15 Data Acquisition (Advanced)
- 16 Tachometer (Advanced)
- 17 Control Systems (Advanced)
- 18 Serial Communications (Advanced)
- 19 Bluetooth Low Energy - IOT (Advanced)
- 20 Wi-Fi (Advanced)

B. Example module: introduction

Introduction: Battery and Voltage Regulation

Educational Objectives:

MEASURE Voltage, current, and energy for a battery
UNDERSTAND Voltage regulation for the robot
INTERFACE The circuits needed to power the robot from batteries

Prerequisites (Module 2)

- Voltage, current, energy, power (Module 2)
- Resistance, capacitance (Module 2)

Recommended reading materials for students:

- Volume 1 Section 1.1
  Embedded Systems: Introduction to the MSP432 Microcontroller, ISBN: 978-1512185676, Jonathan Valvano, copyright © 2017

  or

- Volume 2 Section 9.2
  Embedded Systems: Real-Time Interfacing to the MSP432 Microcontroller, ISBN: 978-1514676585, Jonathan Valvano, copyright © 2017

Every embedded system needs power to operate. The source of power could be
- 120 VAC 60 Hz, with an AC to DC converter
- DC power, like +5V on USB or +12V in an automobile
- Battery
- Energy harvesting like solar or EM field pickup

When debugging the LaunchPad, you use +5V from the PC via the USB cable. However, to run the robot autonomously, it will need battery power. The battery voltage is not constant; it decreases with age and use. Therefore, you will use a **regulator** to provide a constant voltage to power most of the electronics for the robot. In this module, we will introduce two types of regulators: linear and switching. There are many considerations when choosing a regulator, and we will discuss some of these considerations.

You will power the robot motors directly off the battery voltage. The Romi Chassis (Pololu part #3502) can hold 6 AA batteries. If you use NiMH batteries (1.2V each), this will create a +7.2V source for the robot. The motors do not need a constant voltage to operate, and running directly off the batteries is the most efficient use of energy. As you might imagine, the motors use most of the power required by the robot.

The robot will take the battery +7.2V input and create a +5V regulated power source. In particular, you will use the Motor Driver and Power Distribution Board for Romi Chassis (Pololu part #3543). We will explain the battery and voltage regulation in this module. You will connect the +5V regulated power source to the LaunchPad, and the LaunchPad will create a +3.3V power source using its own regulator. The LaunchPad powers the MSP432 with this +3.3V. The MSP432 itself has regulators inside the chip. For example, V_{CORE} is the internal voltage at which the processor operates, and it is typically +1.2V. You will power the motors directly off the battery, some of the external devices with +5V and others with +3.3V.

The **energy** (E in joules) stored in a battery can be calculated from voltage (V in volts), current (I in amps), and time (t in seconds). Energy has neither polarity nor direction. The energy rating for the battery is given in amp-hr, because the assumption is the voltage is constant. The NiMH batteries listed in the lab bill of materials (BOM) are rated at 1900 mA-hr. This means the battery can supply 1 amp for 1.9 hours. Six of these batteries, placed in series, can supply 7.2V at 1 amp for 1.9 hours.

In the lab associated with this module, see Figure 1, you will study the batteries, measuring their energy storage. Next, you will build/interface the circuits needed to power the LaunchPad off batteries.

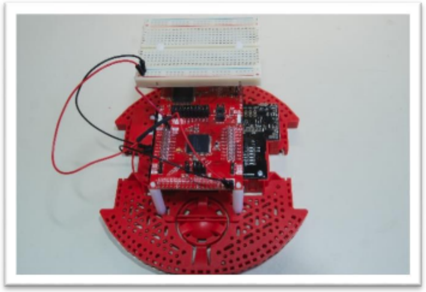


Figure 1. Romi Chassis, Motor Driver and Power Distribution Board, LaunchPad, and small protoboard.

- **Introduction**
- Lectures
- Videos
- Labs
- Activities
- Quizzes

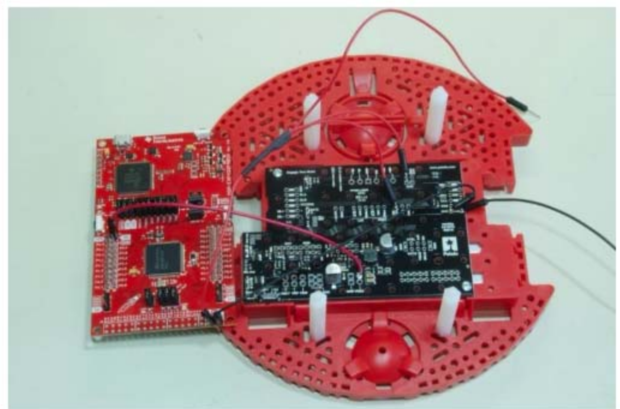
Learning objectives
Resources

B. Example module: lecture slides

Battery and Voltage Regulation

You will learn in this module

- Power sources - Batteries
  - Voltage, V (volts)
  - Current, I (amps)
  - Energy, E (joules)
- Voltage regulation (Constant Voltage)
  - Purpose
  - Types
  - Circuits
- Performance measurements (Lab)
  - Monitoring Battery Voltage, Current, Storage
  - Voltage regulation (DC voltmeter)
  - Noise (AC voltmeter, oscilloscope)



Motor Driver and Power Distribution board Powering TI's Launchpad Development board

- Introduction
- **Lectures**
- **Videos**
- Labs
- Activities
- Quizzes

Teaching slides
Lecture videos

B. Example module: lab manual

Lab 5: Battery and Voltage Regulation

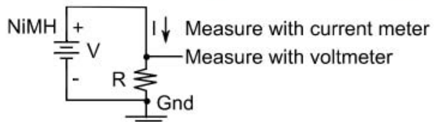


Figure 2. Battery circuit.

The second step is to configure the **MDPDB** for use on the robot. Figure 3 shows the locations of the power-related pins on the motor driver board.

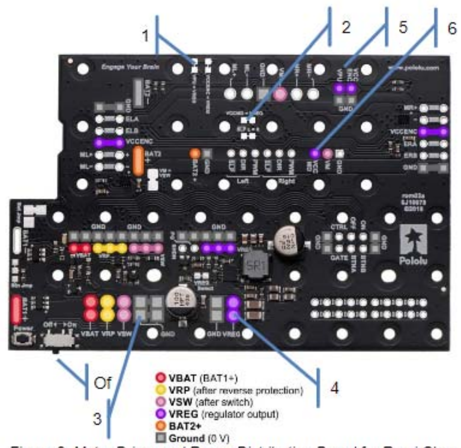


Figure 3. Motor Driver and Power Distribution Board for Romi Chassis.

Follow these steps to configure the motor driver board for the robot:

1) Cut the VPU—VREG jumper on **the MDPDB**
2) Cut the VCCMD—VREG jumper on the **MDPDB**
3) Solder a ground wire from the **MDPDB** to ground on the LaunchPad
   We suggest you review the entire MDPDB User's Guide
   Make sure not to use holes needed later for the motors
4) Connect VREG (+5V) from the **MDPDB** to +5V on the LaunchPad
   You will need a way to connect/disconnect +5V.
   We suggest you solder one end of a wire to the VREG signal on the **MDPDB** and use a female header to connect it to +5V on the LaunchPad.
5) Connect VPU from the **MDPDB** to 3.3V on the LaunchPad
6) Connect VCCMD from the **MDPDB** to 3.3V on the LaunchPad
7) Solder a wire with a male header to the +3.3V power. Solder a second wire with a male header to ground. These two will be used to bring power to the solderless breadboard (shown in Figure 4).

Figure 4 shows the partially completed power system.

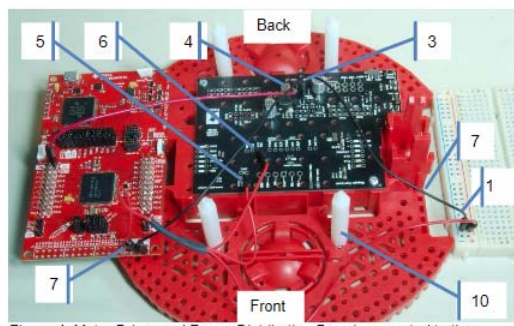


Figure 4. Motor Driver and Power Distribution Board connected to the LaunchPad. Grounds connected. VREG (MDPDB) to +5V (LaunchPad). +3.3V (LaunchPad) to VPU (MDPDB) and VCCMD (MDPDB).

- Introduction
- Lectures
- Videos
- **Labs**
- Activities
- Quizzes

Lab procedures
Lab videos

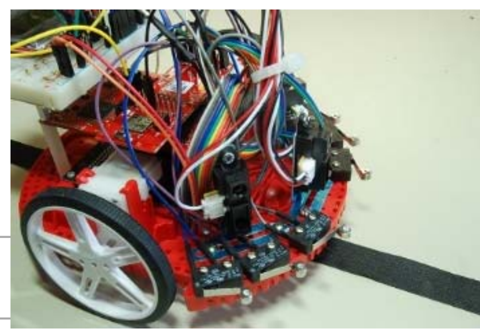
B. Module 1 Getting Started

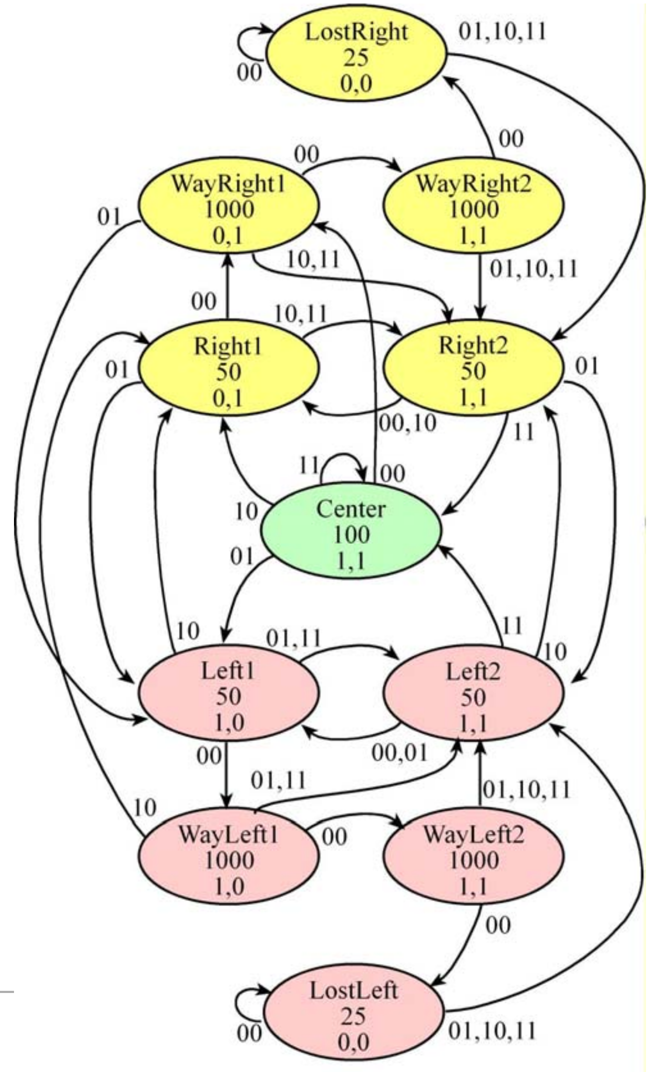
- TI-RSLK Module 1 - Lab video 1.2 - Getting started with CCS 02:14
  - https://training.ti.com/ti-rslk-module-1-lab-video-12-getting-started-ccs?cu=1135347
- TI-RSLK Module 1 - Lab video 1.3 - Running the TExaS logic analyzer 03:26
  - https://training.ti.com/ti-rslk-module-1-lab-video-13-running-texas-logic-analyzer?cu=1135347

Note that TExaS display also includes an Oscilloscope output

B. Module 3 Architecture, Module 7 Finite State Machine

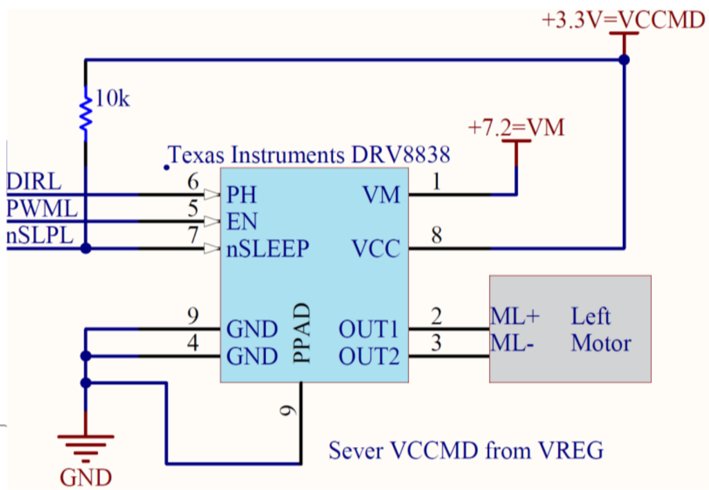
- Last two minutes of ARM architecture (start at 18:00)
  - Exposes important concepts (critical section)
  - https://training.ti.com/ti-rslk-module-3-lecture-video-part-i-arm-cortex-m-architecture?cu=1135347
- 7.2 FSM lab video (start at 1:44)
  - https://training.ti.com/ti-rslk-module-7-lab-video-7-2-running-solution-code-designing-better-fsm?cu=1135347

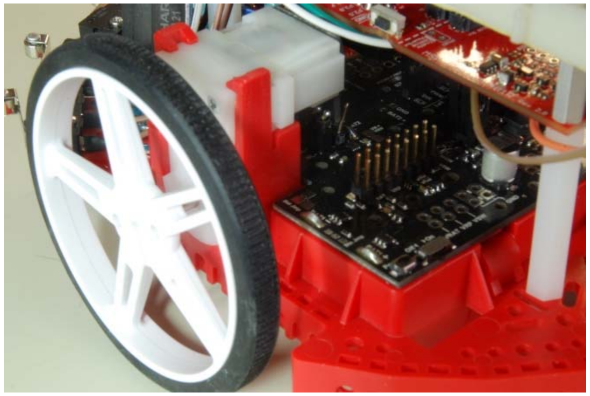




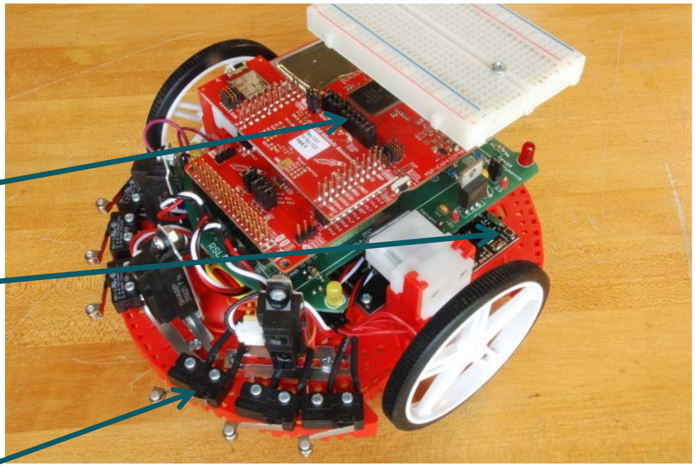
B. Module 12. DC motors

- Lab 12 motor lab measurements 2:39
  - https://training.ti.com/ti-rslk-module-12-lab-video-12-1-demonstrate-motor-fundamentals?cu=1135347
- Lab 12 present patterns 1:46
  - https://training.ti.com/ti-rslk-module-12-lab-video-12-2-demonstrate-robot-moving-preset-pattern?cu=1135347





C. Tachometer Hands On Activity



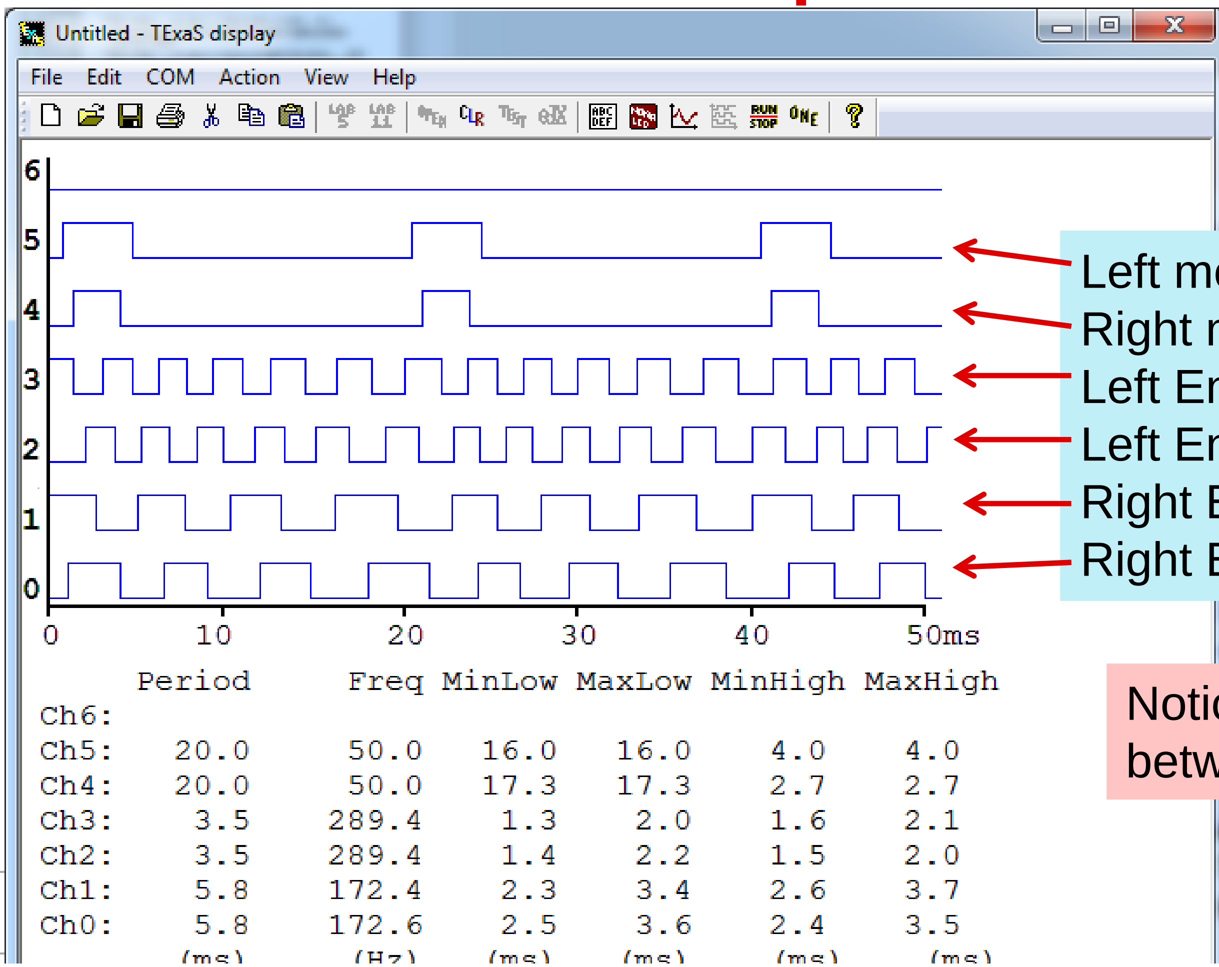
- Place the robot up on blocks (so it doesn't fly off table)
- Make sure 7-bit jumper is in place on LaunchPad
- Connect USB from LaunchPad to PC
- Turn on robot power
- Start TExaSdisplay
  - Execute **Open Next Port** until is connects to LaunchPad
  - Click on logic analyzer toolbar (View->Logic Analyzer)
- Run Lab 17 (press switch 3)
  - Lab 17 is autonomous racing robot solution, proportional controller
  - It attempts to follow constant distance from the closest wall
- Observe PWM outputs and tachometer inputs on logic analyzer

C. Module 17 Proportional Control

```
void SysTick_Handler(void){ // runs at 100 Hz
  if(LeftDistance < RightDistance){
    Error = LeftDistance - SetPoint;
  }else {
    Error = SetPoint - RightDistance;
  }
  UR = PWMnominal+Kp*Error; // proportional control
  UL = PWMnominal-Kp*Error; // proportional control
  if(UR < (PWMnominal-SWING)) UR = PWMnominal-SWING; // -1000 to +1000
  if(UR > (PWMnominal+SWING)) UR = PWMnominal+SWING;
  if(UL < (PWMnominal-SWING)) UL = PWMnominal-SWING;
  if(UL > (PWMnominal+SWING)) UL = PWMnominal+SWING;
  Motor_Forward(UL,UR);
  ControllerFlag = 1;
}
```

Which switch you press
0 – PWMnominal = 2500
1 – PWMnominal = 3000
2 – PWMnominal = 3500
3 – PWMnominal = 4000
4 – PWMnominal = 4500
5 – PWMnominal = 5000

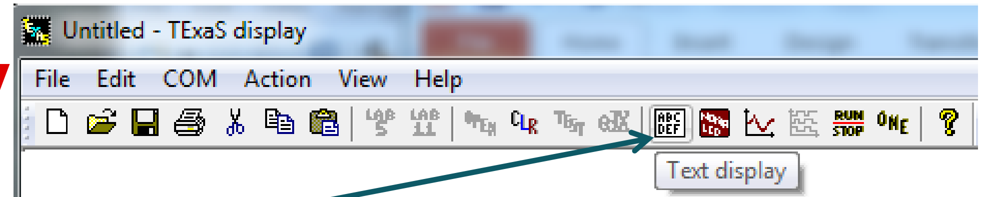
C. Module 17 PWM output and Tachometer input

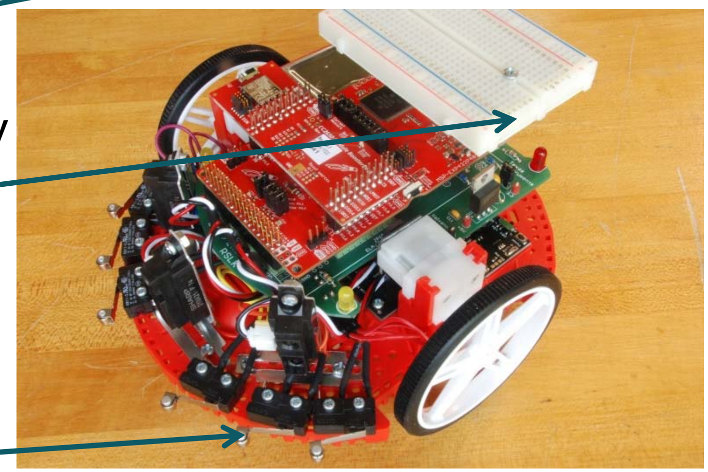


Left motor PWM, P2.7
Right motor PWM, P2.6
Left Encoder A, P8.2
Left Encoder B, P9.2
Right Encoder B, P10.5
Right Encoder A, P10.4

Notice the relationship between power and speed

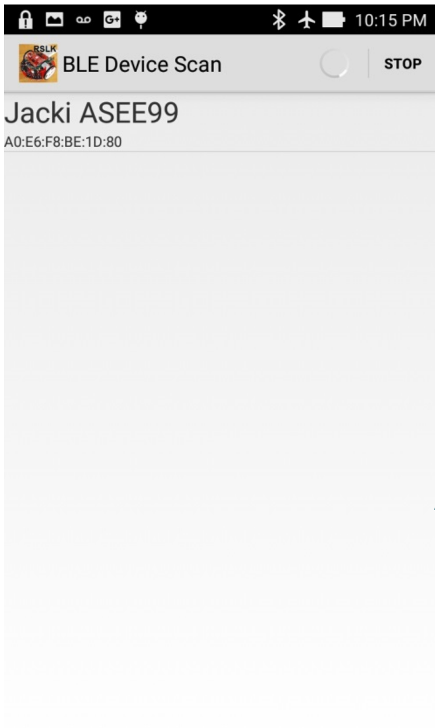
C. BLE Hands On Activity

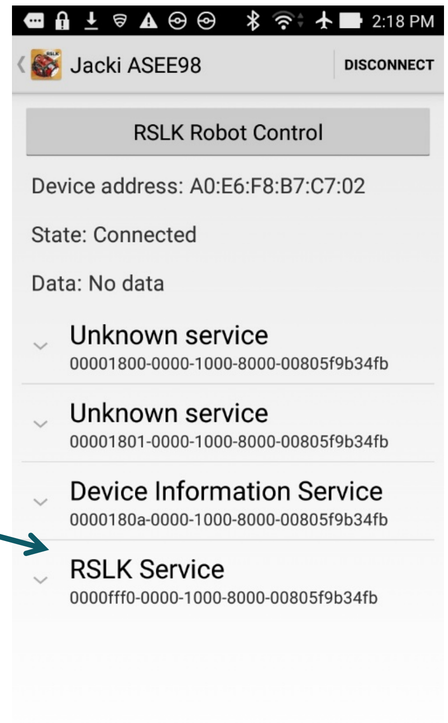


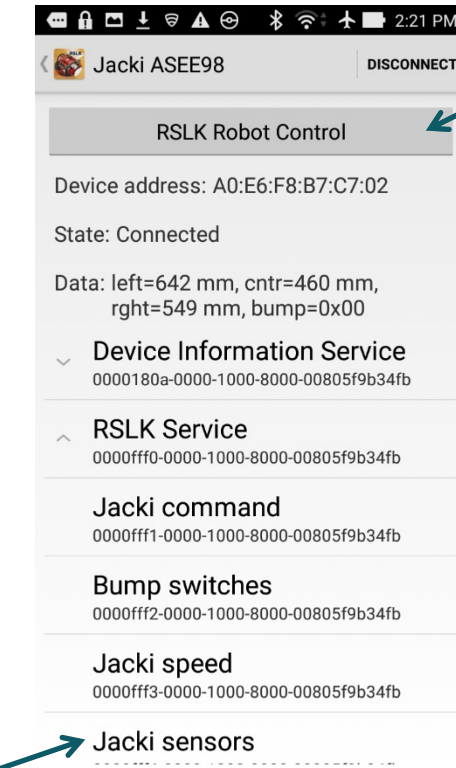


- Change TExaSdisplay mode to text
  – If you power cycle the robot, restart TExaSdisplay
- Reset the MSP432 (observe robot number)
- Download and install RSLK Android app
  – http://users.ece.utexas.edu/~valvano/android
- Start RSLK Android app
- Run Lab 19 (press switch 4)
  – Connect the RSLK to your robot
  – There are 4 services (RSLK Service is the interesting one)
  – The RSLK service has 4 characteristics (click on Jacki sensors)

RSLK Service has four characteristics

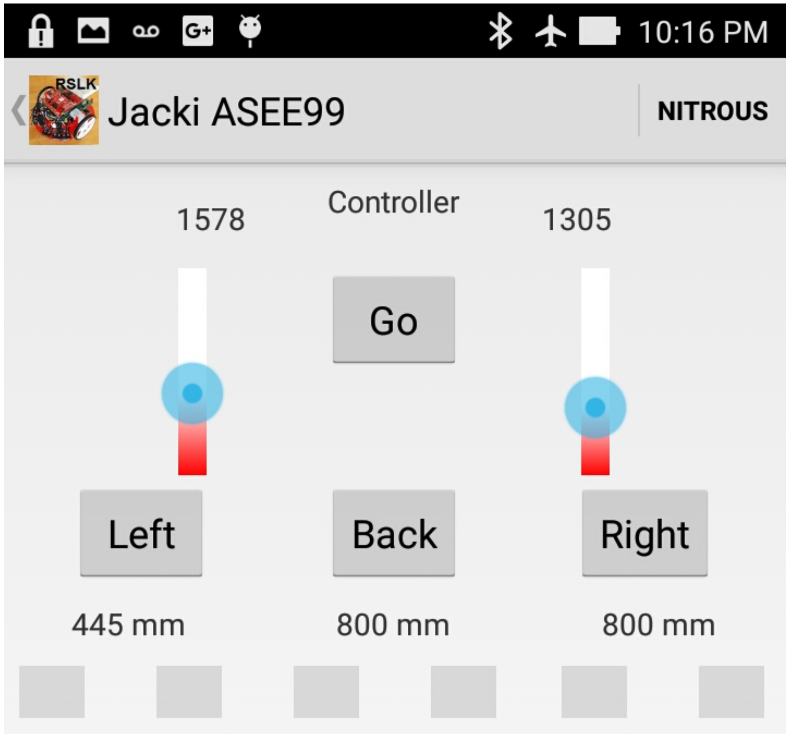






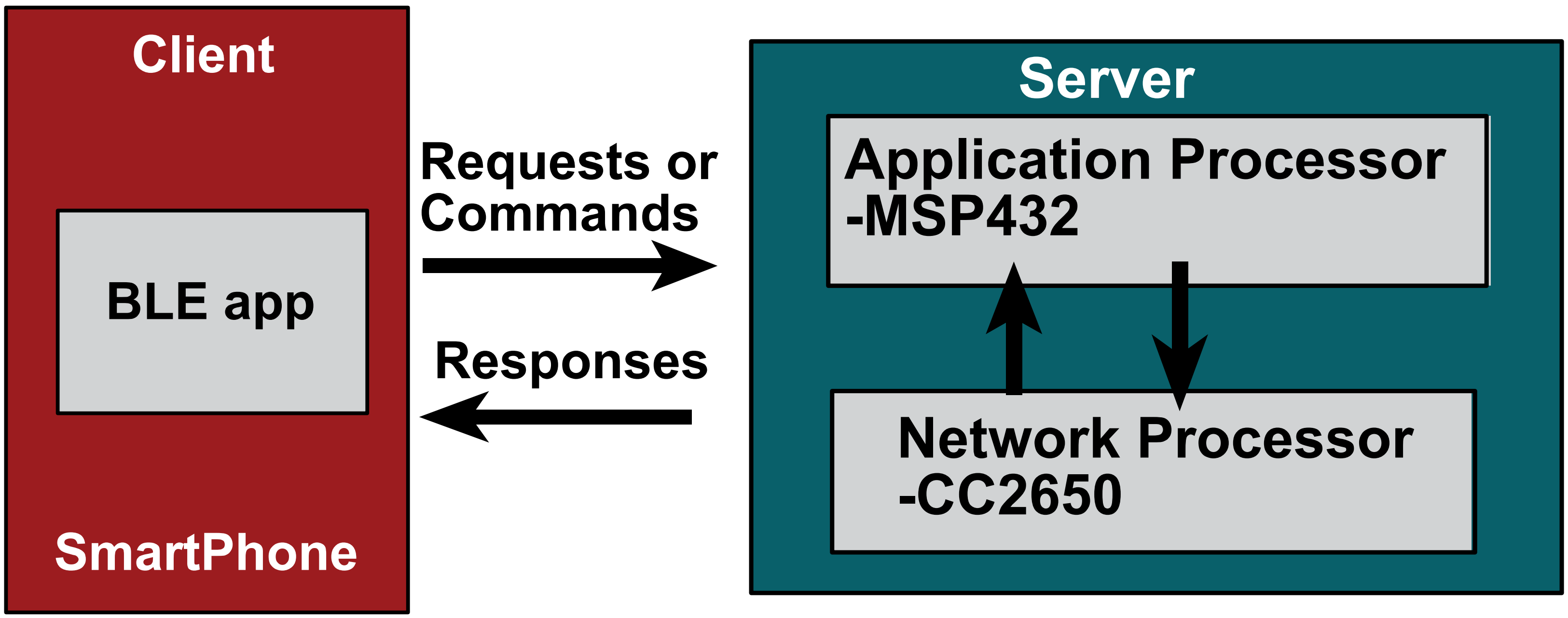
Click on characteristic to read

RSLK Robot controller

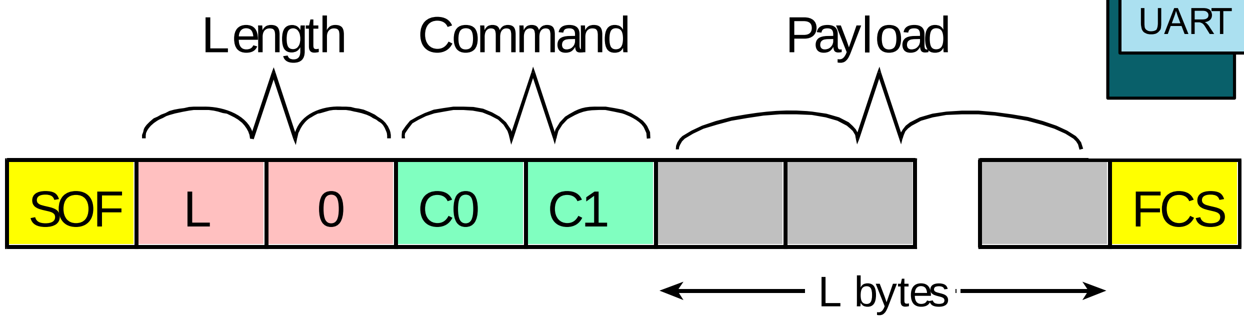


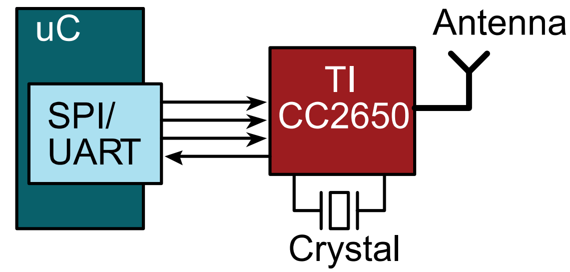
- Push the buttons
- Move the sliders
- Observe sensor data
  - On touch to slider
- Observe the BLE packets

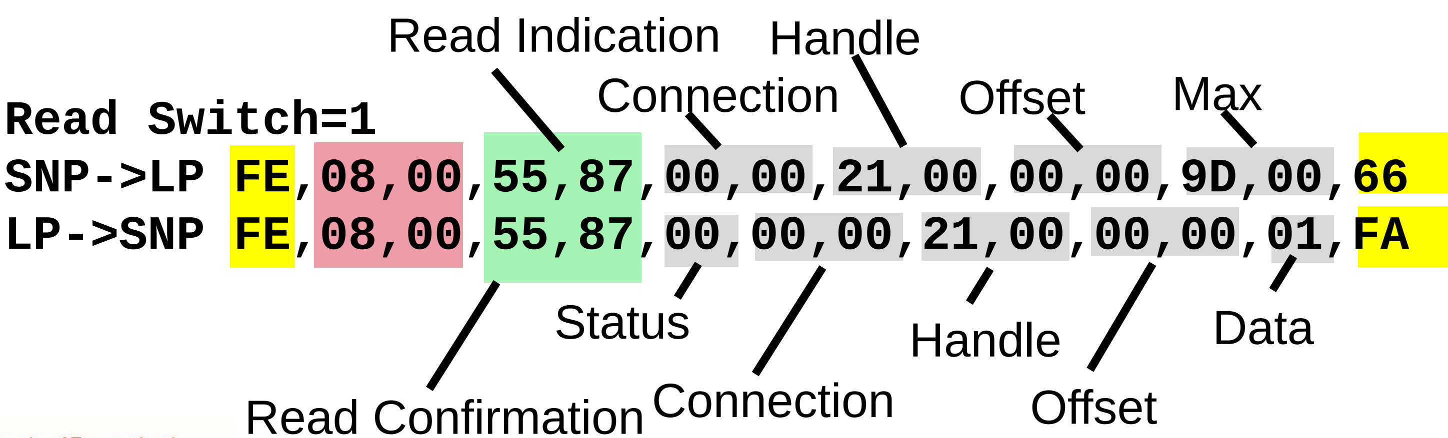
Client-server



Simple Network Processor





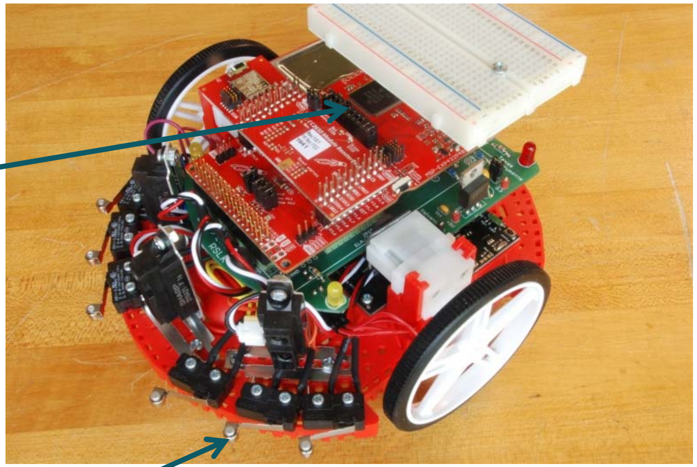


BLE made simple

```
void BLE_Init(uint8_t robotNum){volatile int r;
  EnableInterrupts();
  r = AP_Init();
  AP_AddService(0xFFF0); // RSLK GATT service
  //-------------------------
  JackiCommand = 0;  // read/write parameter
  AP_AddCharacteristic(0xFFF1,1,&JackiCommand,0x03,0x0A,"JackiCommand",&ReadCommand,&WriteCommand);
  //-------------------------
  JackiBumpSensor = Bump_Read(); // read only parameter (get from bump sensors)
  AP_AddCharacteristic(0xFFF2,1,&JackiBumpSensor,0x01,0x02,"JackiBumpSensor",&ReadJackiBumpSensor,0);
  //-------------------------
  JackiSpeed[0] = JackiSpeed[1] = 500;   // read/write parameter
  AP_AddCharacteristic(0xFFF3,4,JackiSpeed,0x03,0x0A,"JackiSpeed",&ReadJackiSpeed,&WriteJackiSpeed);
  //-------------------------
  // four 16-bit read only parameters (get from IR distance sensors)
  AP_AddCharacteristic(0xFFF6,8,SensorData19,0x01,0x02,"JackSensors",&ReadDistance,0);
  //-------------------------
  AP_RegisterService();
  AP_StartAdvertisementJacki19(robotNum);
}
```

Self-discovery

- Turn off power
- Remove 7-pin shorting jumper
- Disconnect USB
- Turn on robot
- Run one of the labs
  - Sw0 Lab 7 FSM line follow (use fluffy black tape)
  - Sw1 Lab 12 Simple motor open loop control
  - Sw2 Lab 16 Tachometer, incremental controller to set speed
  - Sw3 Lab 17 Proportional control, autonomous driving using distance sensor
  - **Sw4 Lab 19 BLE**



- Sign up for robot racing
- 4:30-6pm in TI pavilion
- 3 minute races
- prizes

Takeaway: Bottom up

- **Bottom up: From simple to complex**
  - Transistors → Gates → Computer → Systems
  - Assembly → C → Java/C++ → LabVIEW
- **Abstraction**
  - Understand → Put it in a box → Use the box
- **Systems**
  - Take two systems→ Connect → New system+

RSLK Race @ ASEE Robotics Pavilion Monday

- **Register your team to race for a prize**
  - 4:30pm-5pm Practice at RSLK Racing School
  - 5pm-6pm Track Races
- **Workshop teams get priority, races open to public after completion of Racing School**

Race Rules

Semifinal heats 4 robots at a time

3 minute time limit or first to 4 laps

Top 8 race in finals, Top team wins prize

1st Place Prizes for teams of max 4 people

For more information

Jonathan Valvano valvano@mail.utexas.edu

http://users.ece.utexas.edu/~valvano/

EE319K Introduction

EE445L Interfacing and systems

EE445M Real-time operating systems

The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering

https://www.edx.org/course/embedded-systems-shape-world-utaustinx-ut-6-10x

https://www.edx.org/course/embedded-systems-shape-world-multi-utaustinx-ut-6-20x

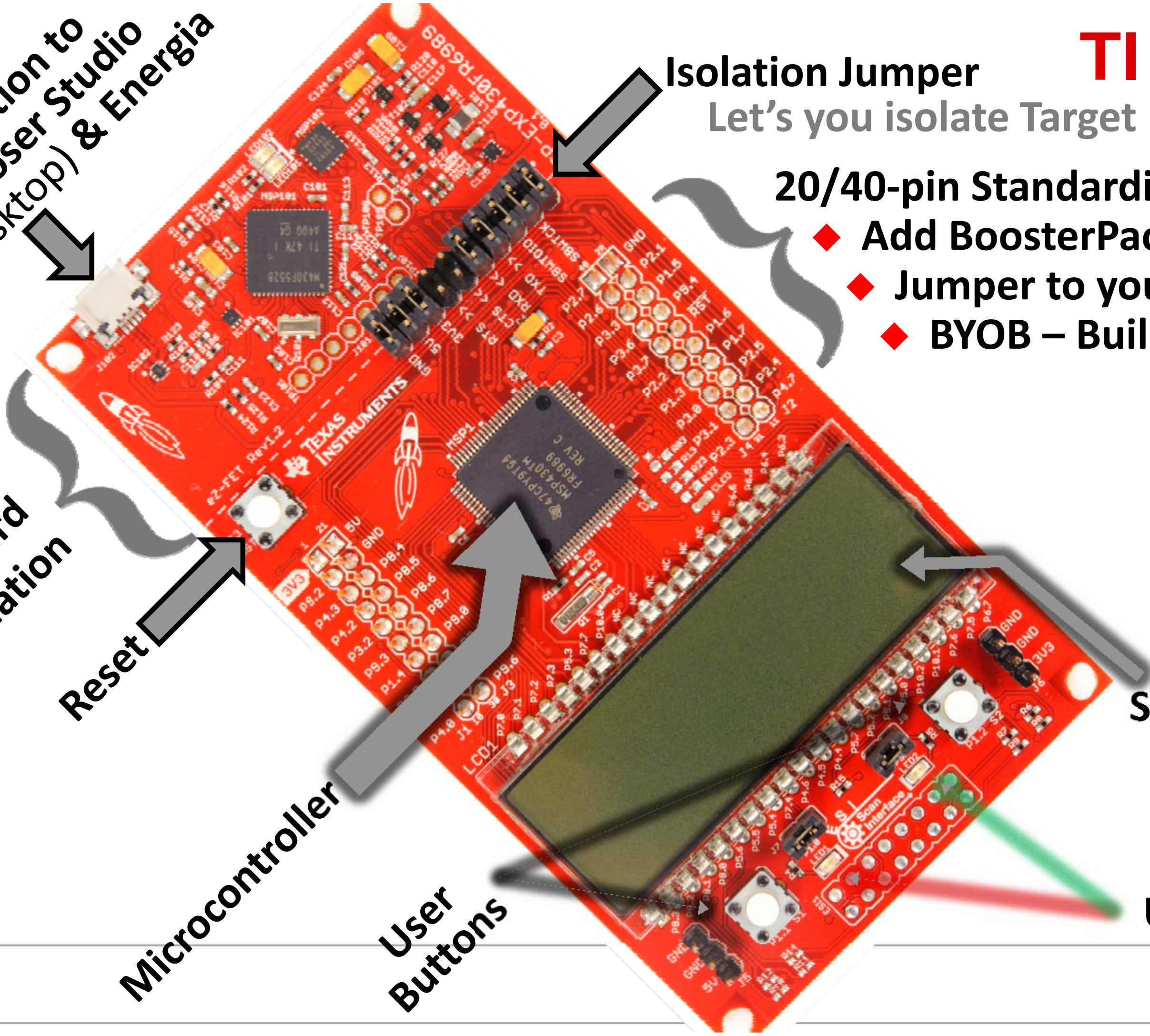
https://www.edx.org/course/real-time-bluetooth-networks-shape-the-world

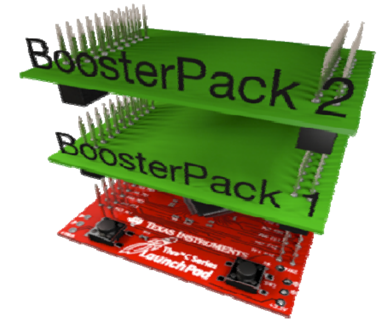
Mark Easley, Jason Rubadue, Franklin Cooper

www.ti.com/rslk univ@ti.com

Extra slides

TI LaunchPad™



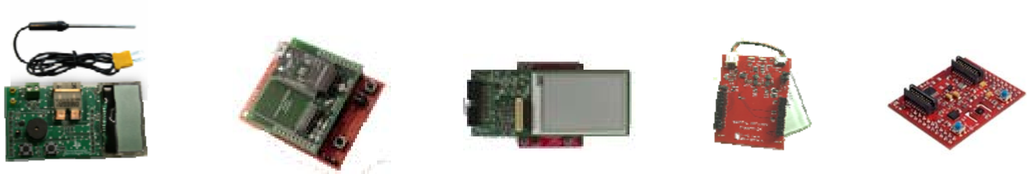


LaunchPad is TI's Common Denominator

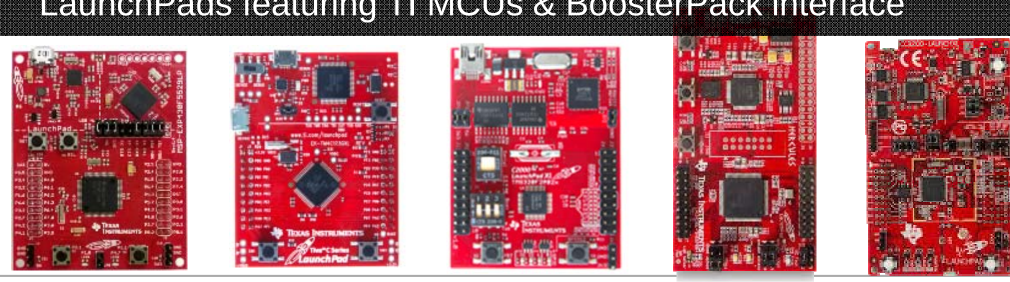
Modular hardware enables developers to explore new ideas quickly

TI Wireless: Sub-1GHz, NFC/RFID, Wi-Fi, ZigBee, BLE, Bluetooth

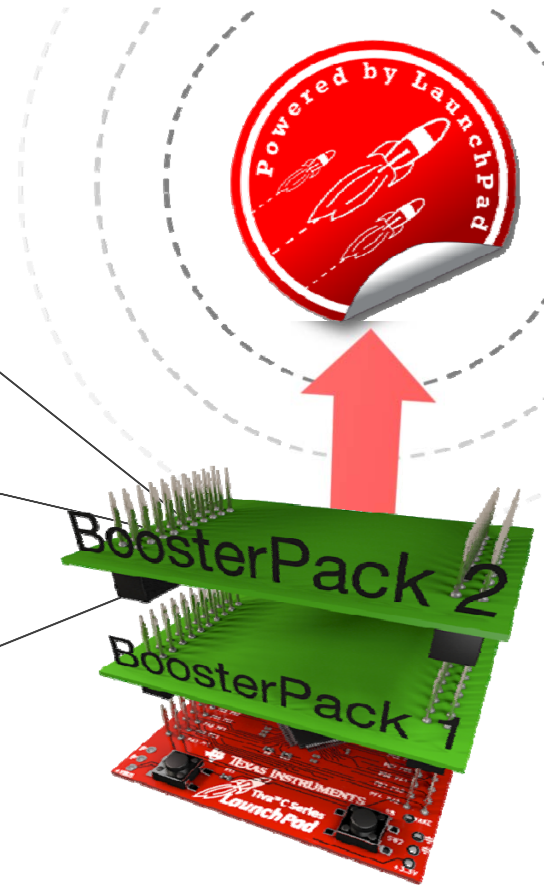
Analog, sensors, displays & more from TI, 3rd parties & Maker community



LaunchPads featuring TI MCUs & BoosterPack interface

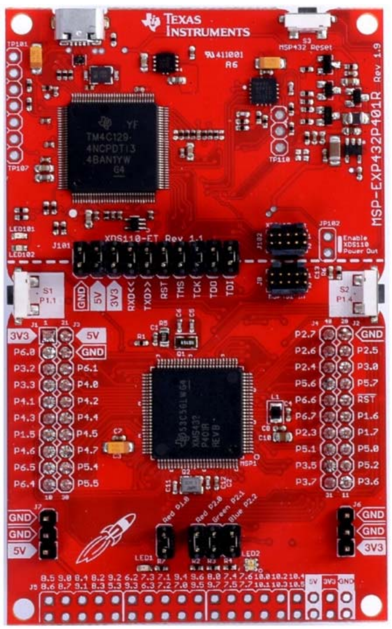


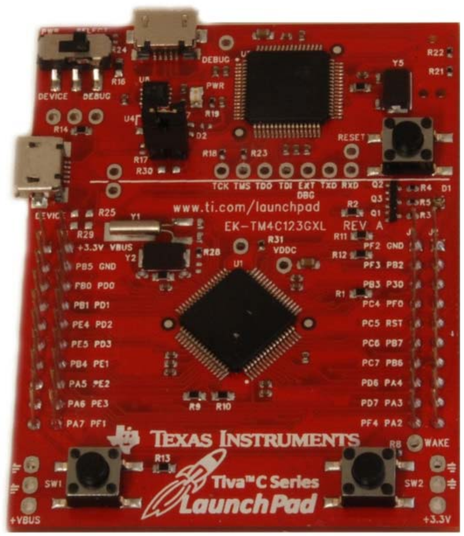
MSP430	TM4C	C2000	Hercules	CC3200
(Ultra-Low Power)	(ARM Cortex M4F)	(Real-time Control)	(Safety)	(MCU + WiFi)

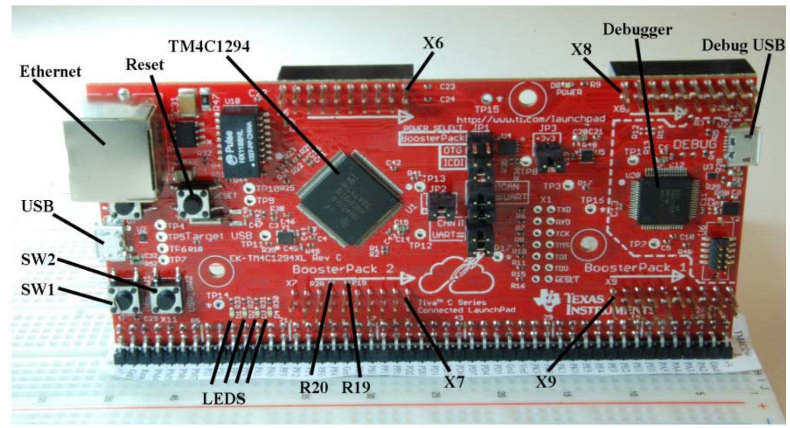


TI LaunchPad

- **Students should have their own board**





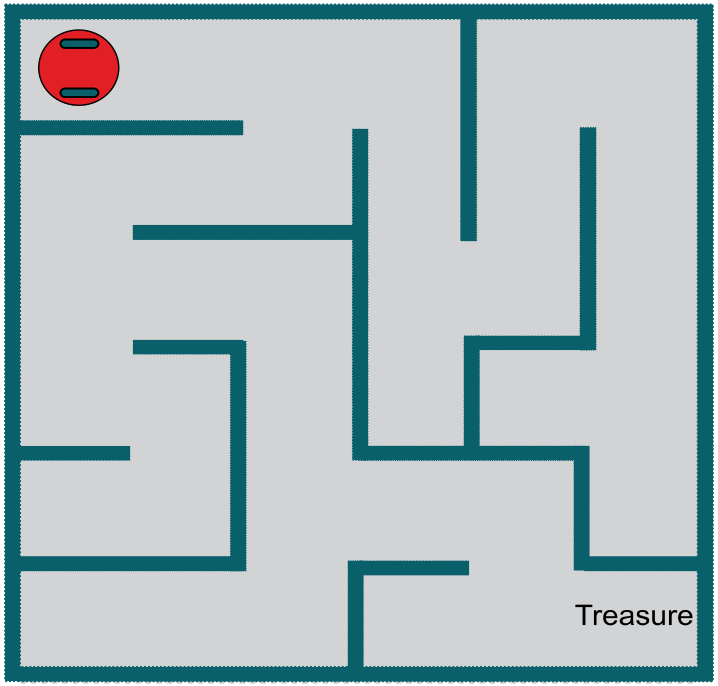


MSP432 LaunchPad $13
TM4C LaunchPad $13
Connected LP $20

Support for teaching on university.ti.com

Robotic System Learning Kit

- Systems approach
- Aggregate components
- Explorer robot
- Introduction
  - Volume 1
  - No Calculus
- Intermediate
  - Volume 2
  - Circuits
  - Signal processing

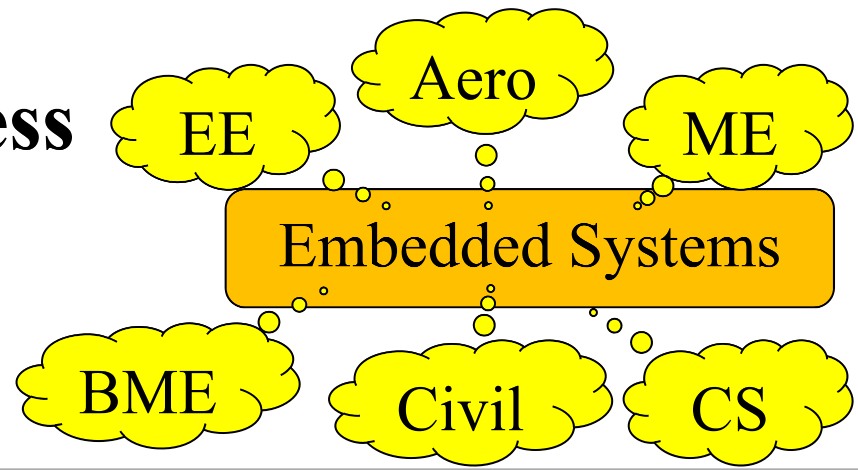


Applications

- **Compiler, Simulator, Debugger**
  - TI: Code Composer Studio
  - Keil: uVision
  - TExaS *(simulation, grading, LA, scope)*
- **Circuit design and PCB layout**
  - PCB Artist
  - Eagle (Autodesk)
  - Circuit Maker (Altium)
- **Design tools**
  - TI: WEBENCH

Educational Objectives

- **Outcomes, Measureables**
  - Career opportunities
  - Economic growth
  - Student feedback
- **Educational effectiveness**
  - Improved performance
  - Reduced resources
- **Educational team**



Takeaway: See one, do one, teach one

Students learn by doing

- **Equipment must work**
- **Assignments must be clear**
- **Assignments must be relevant**
- **Tasks support learning objectives**
- **Professors must do labs**

Students learn by teaching

A. Support for teaching

Website (download and edit)

- Examples for TM4C123, TM4C1294, MSP432
- PowerPoint slides
- Lab manual, data sheets
- **http://users.ece.utexas.edu/~valvano/**

http://ow.ly/ophC301Aa0p

3 MOOCs: Intro, Advanced (wifi), RTOS-BLE

LaunchPad tester

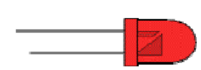
Adopt a book →
Free parts for LaunchPad tester

http://users.ece.utexas.edu/~valvano/arm/tester/

Takeaway: Empower Students

Students need to learn outside of lab

- **Students should have their own DVM**
- **Show labs to friends and parents**
- **Encourage them to work beyond lab**
  - Find sources of free parts
  - Give simple stuff away
- **Mentor their careers**
  - Job versus grad school
  - Online presence

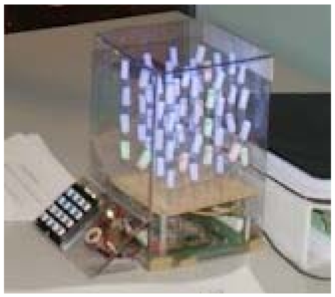


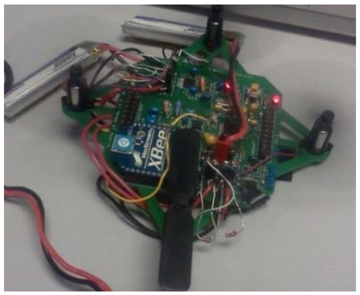
Successes: Competitions

Students need to appreciate relevance

- **Appropriate use of teams**
- **Build things that are fun to play with**
  - Show off to friends, family, interviewers
- **Competitions**
  - Fun, intense
- **Open-ended**
  - Creativity, life-long learning, springboard

Competitions





https://youtu.be/0ZOI5AGtdf0

http://youtu.be/bZ1fXtN1T08
http://youtu.be/GKctvlvprAQ

Conclusions

- Bottom-up
- Lab-centered
- Empower students
- Motivate students
- Be flexible
- Be a team builder
- Make a plan and do it

→

Understanding
Design
Innovation