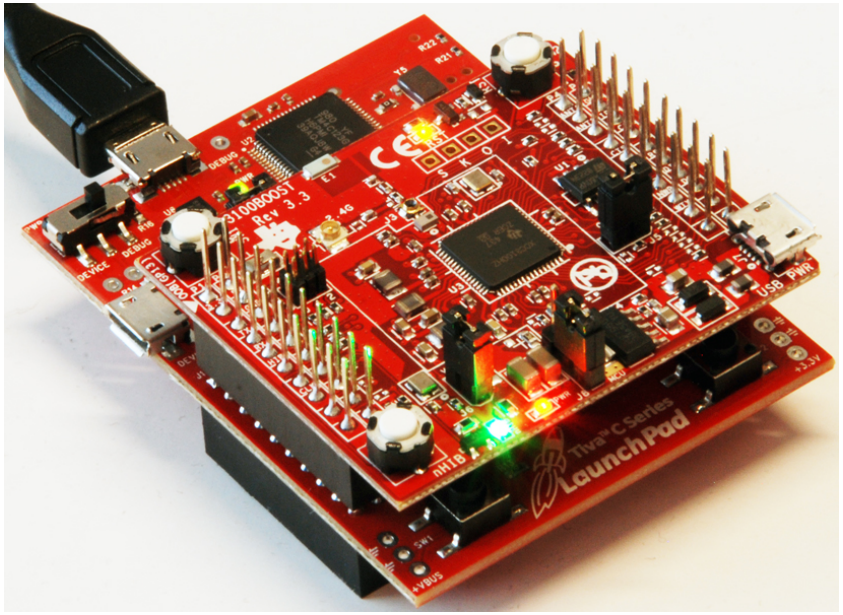
Embedded Systems Laboratory

- Using ARM Cortex M4
- From the Basics to Applications
- Internet of Things



Why M4?

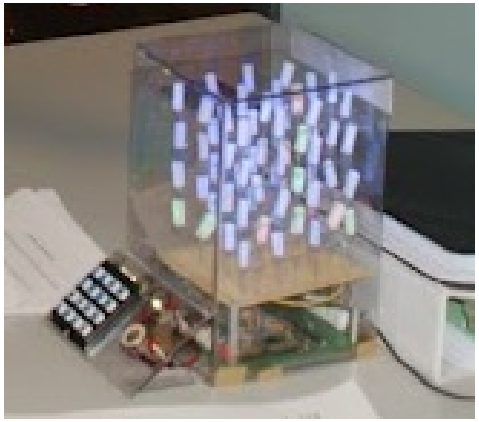
- Market share
- Complexity
- Parallelism
- Verification

Jonathan Valvano

THE UNIVERSITY OF TEXAS AT AUSTIN
UT ECE
ELECTRICAL & COMPUTER ENGINEERING

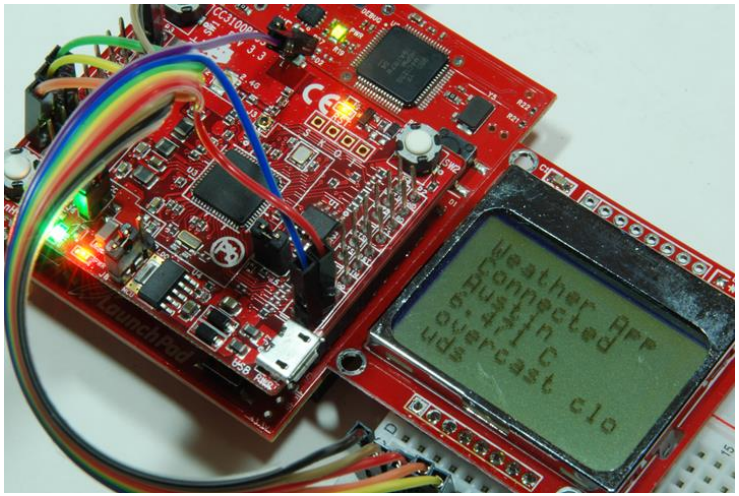
Outline

1. **Objectives** *What do students need forever?*
2. **Approach → 5 Takeaways**
3. **Boards, Books and Labs**
4. **Successes → Competitions**
5. **Conclusions**
6. **IoT demonstration**



Engineers make two things:

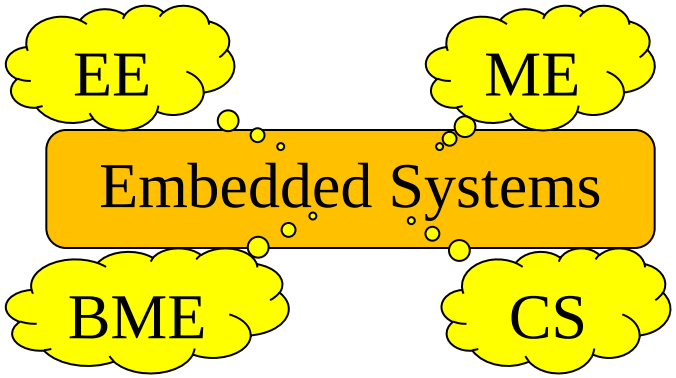
- Systems
- Interfaces between systems



Jonathan Valvano

1. Educational Objectives

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



2A. Takeaway: Bottom up (what?)

- 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+

2B. Takeaway: Lab-centered

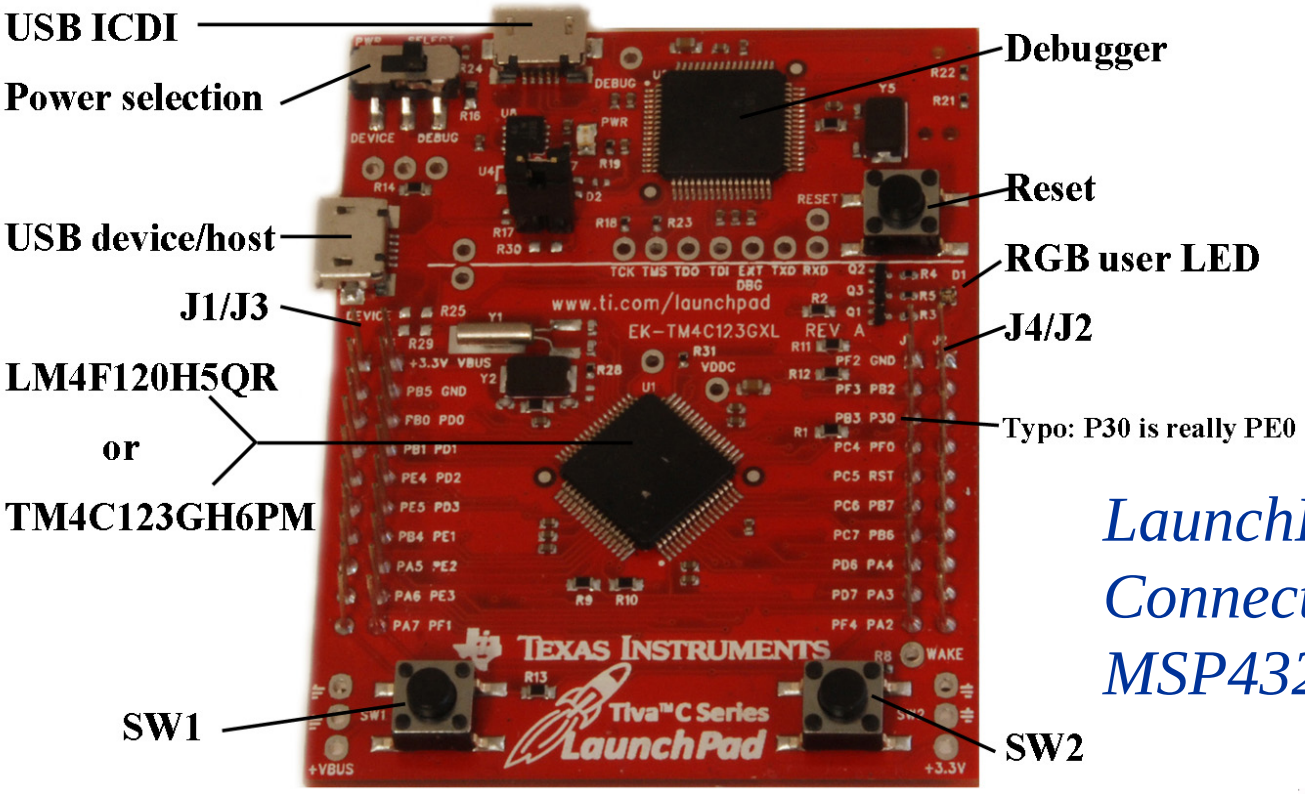
Students learn by doing

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

Students learn by teaching

2C. Takeaway: Empower Students

- **Students should have their own board**



LaunchPad $13
Connected LP $20
MSP432 $13

2C 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

2D. Takeaway: Structure vs Flexibility

Pedagogy: students learn at different speeds and in different ways

- Some need structure
  - Demonstrate working labs
- Some thrive on open ended design
  - Let students negotiate deliverables
- Allow for extra credit
- Create an open-ended design lab

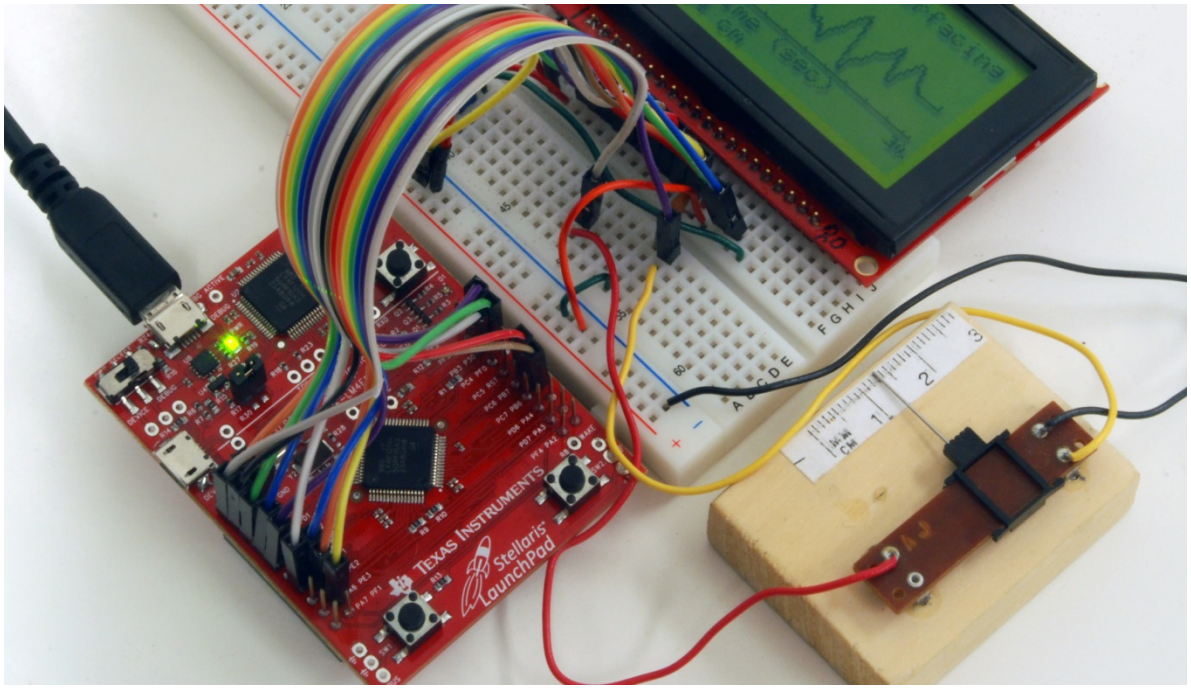
2E. Takeaway: Team-approach

It takes a village to educate

- **Empower the TAs**
  - **Invite them into the decision circle**
- **Empower the staff**
  - **Invite them into the decision circle**
- **Make excuses to show off projects**
  - **Chairman, Dean, Newspaper**
  - **Promote your students**

3. Boards, Books and Labs

Tiva LaunchPad TM4C123



- 43 I/O pins $13
- 32k RAM
- 256k EEPROM
- 80 MHz Cortex-M4
- serial, SPI, ADC, CAN
- timer, PWM, DMA
- interrupt controller
- JTAG debugger
- serial through USB
- floating point

EK-TM4C1294XL, 90 I/O pins, 256k RAM, 1M ROM, 120 MHz, Ethernet $20

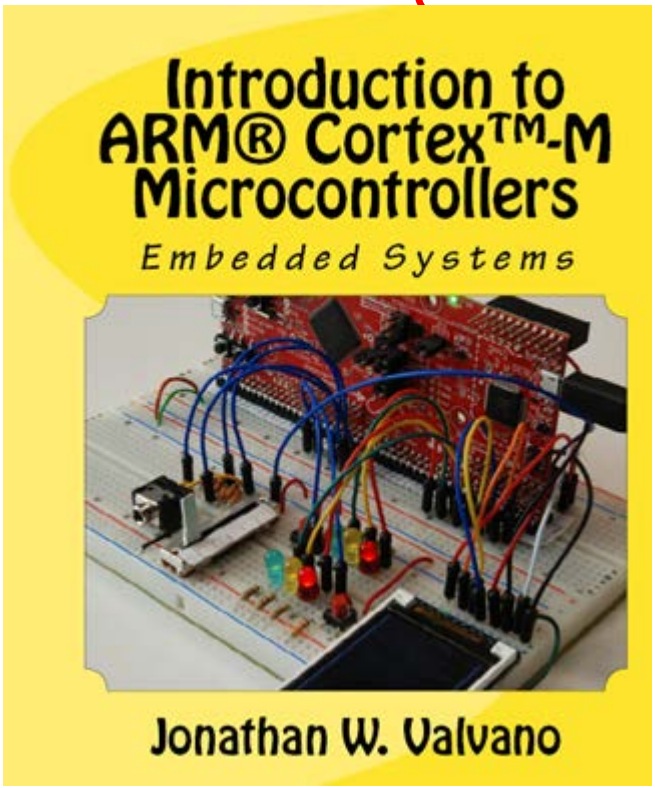
MSP-EXP432P401R, 67 I/O pins, 64k RAM, 256k ROM, 48 MHz, low power $13

3. Applications

- Compiler, Simulator, Debugger
  - Texas Instruments Code Composer Studio
  - Keil uVision
  - TExaS *(simulation, grading, scope)*
- Circuit design and PCB layout
  - PCB Artist
  - Eagle (100 by 80 mm, 2 layers)
- Design tools
  - Texas Instruments Filter Pro

3. Introduction to Embedded Systems

Volume 1 (freshmen EE or BME)



(bottom-up)

- Assembly or C programming
- Switch and LED interfacing
- Design and Debugging
  - Simulation, logic analyzer, scope
- Finite State Machine
- Local variables and stacks
- DAC output and interrupts
- LCD graphics interface, fixed-point
- ADC input, systems design
- UART and distributed systems
- Capstone design (video game)

Jonathan Valvano 5500 sold 505 pages, $41

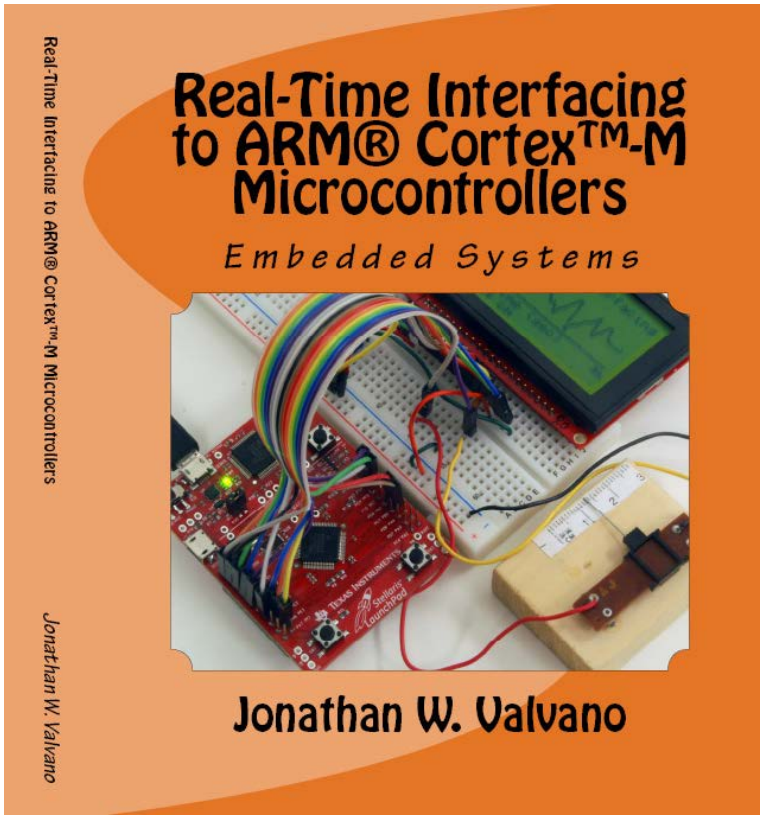
Embedded Systems – Shape the World

- What is and isn't a MOOC?
  - Spring 2014 and Spring 2015
  - over 70,000 enrolled
  - over 11,000 did a lab requiring a kit
  - over 5,300 got certificates (7.5%)
  - 2/3 who started, finished
  - 91% approval rate

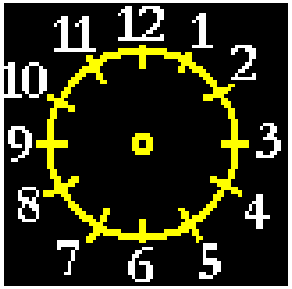
- Lab kit **Physical kit increased completion rates**
- Teaching videos
- LaunchPad simulator, graders, voltmeter, scope
- Today's IoT demo was a MOOC lab

3. Interfacing and Systems

Volume 2 (junior EE)



- Graphics device driver
- Hardware/software debugging
- Design and debugging
- Alarm clock
- Stepper motor
- Music player
- Temperature data acquisition
- Ethernet and wireless networks
- PCB layout, power
- Capstone design (open ended)



Kindle version

600 pages, $43

2700 sold

Jonathan Valvano

3. Real-Time Operating Systems

Volume 3 (senior/grad EE)

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

Jonathan Valvano

447 pages, $36

2000 sold

3. Support for teaching

Web site (download and edit)

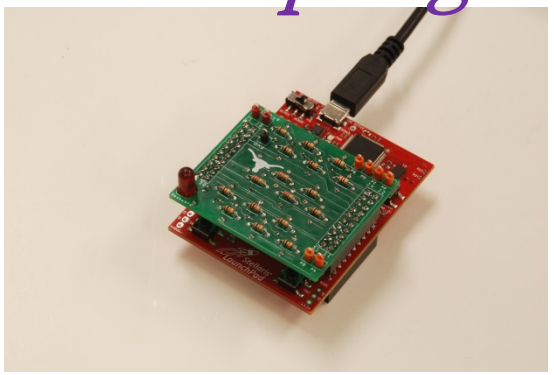
- Examples for TM4C123, TM4C1294, MSP432
- PowerPoint slides
- Lab manual, data sheets
- **http://users.ece.utexas.edu/~valvano/**

EdX Course rerun Spring 2016

Launchpad tester

Adopt a book →
Free parts for Launch

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



4. 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

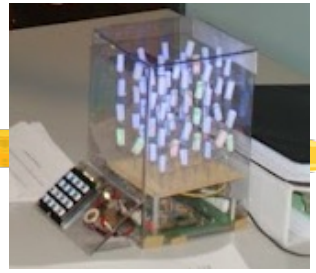
4. Competition

Volume 1 (freshmen EE or BME)

- Handheld game
- Peer review
- Teams of 2

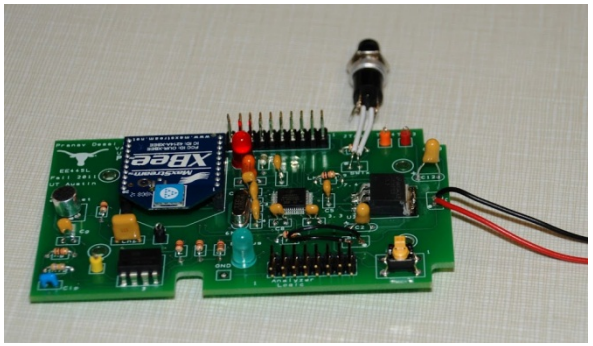
http://youtu.be/QxDQUUDStOw
http://youtu.be/z6_jlM2Y5qI

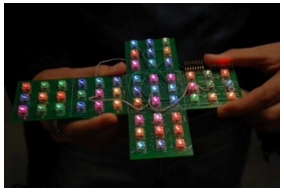
4. Competition



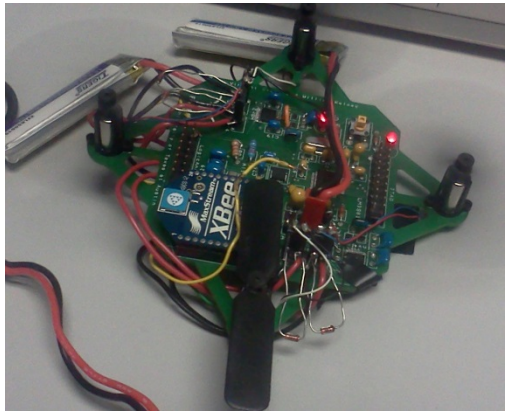
Volume 2 (junior EE)

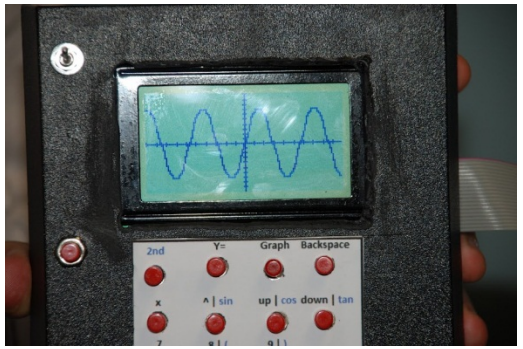
- Requirements document
- Design cycle
- Design for test
- Systems Engineering
- Verification





http://www.youtube.com/watch?v=K9FD50qpGwg



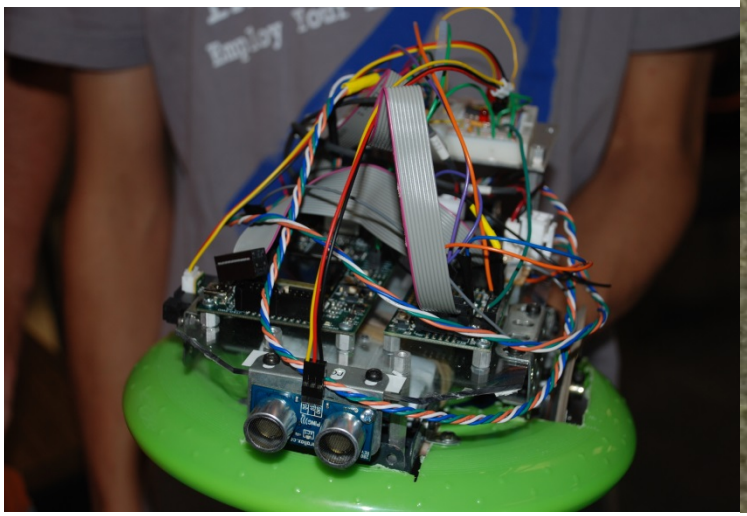


Jonathan Valvano

4. Competitions

Volume 3 (senior/grad EE)

- Autonomous Robot Racing
- Teams of four

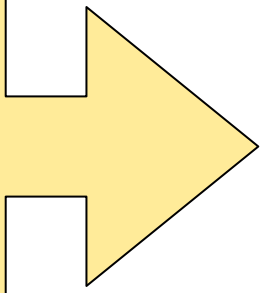


Jonathan Valvano

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

5. Conclusions

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



Understanding
Design
Innovation

5. Interesting web sites

Valvano Example code

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

http://tinyurl.com/nuq4zpx (CCS projects)

TI Example code

http://www.ti.com/tool/sw-ek-tm4c123gxl

http://www.ti.com/tool/ek-tm4c123gxl

Free samples

http://www.ladyada.net/library/procure/samples.html

Compilers

http://www.ti.com/tool/ccstudio

http://www.keil.com/arm/mdk.asp

5. For more information

Jonathan Valvano

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

EE319K Introduction

EE445L Interfacing and systems

EE445M Real-time operating systems

valvano@mail.utexas.edu

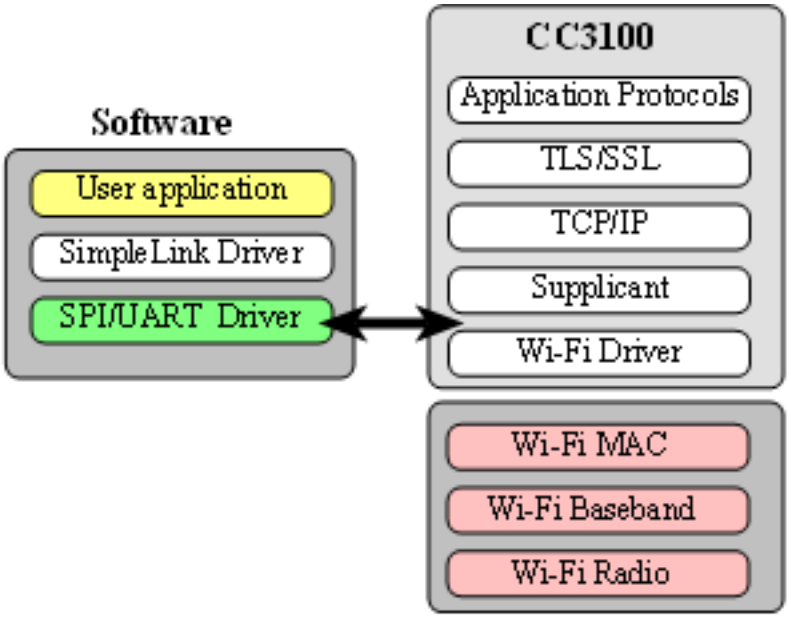
https://www.edx.org/course/utaustinx/utaustinx-ut-6-02x-embedded-systems-4806
http://users.ece.utexas.edu/~valvano/edX/
http://users.ece.utexas.edu/~valvano/Volume1/E-Book/VideoLinks.htm

Texas Instruments

univ@ti.com

6. IoT demonstration

- Code Composer Studio
- TExaSdisplay/PuTTY
- TM4C123 LaunchPad
- CC3100 Booster Pack

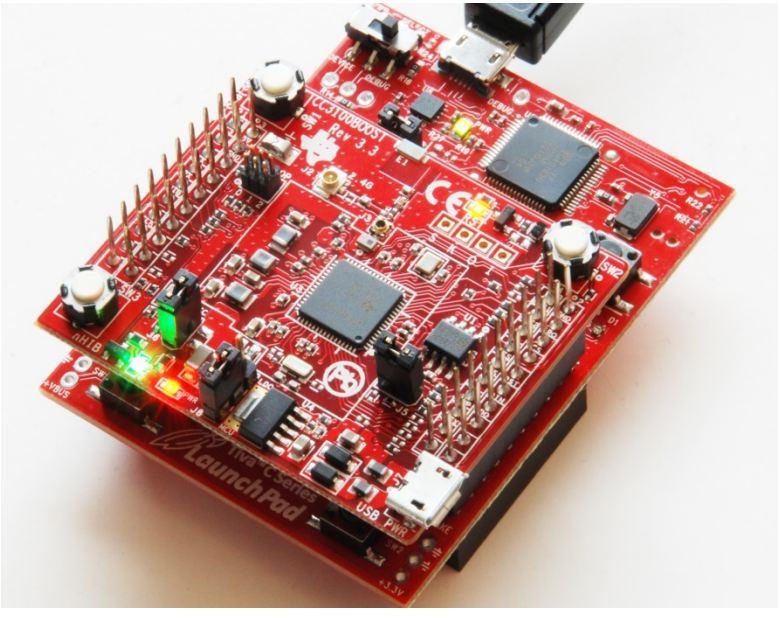


ValvanoWareTM4C123

- **CC3100GetWeather_4C123**
- **CC3100DataLog_4C123**

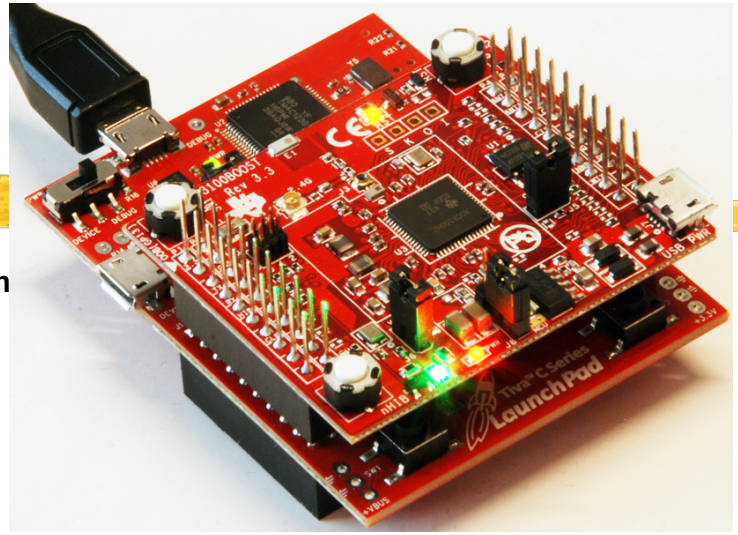
6. IoT demonstration overview

- Install drivers
  - Device Manager
- Configure Code Composer Studio
  - Import examples into workspace
- Get Weather
  - AP, sockets, TCP
- Data Logger
  - Server, Python



Jonathan Valvano

6. CC3100

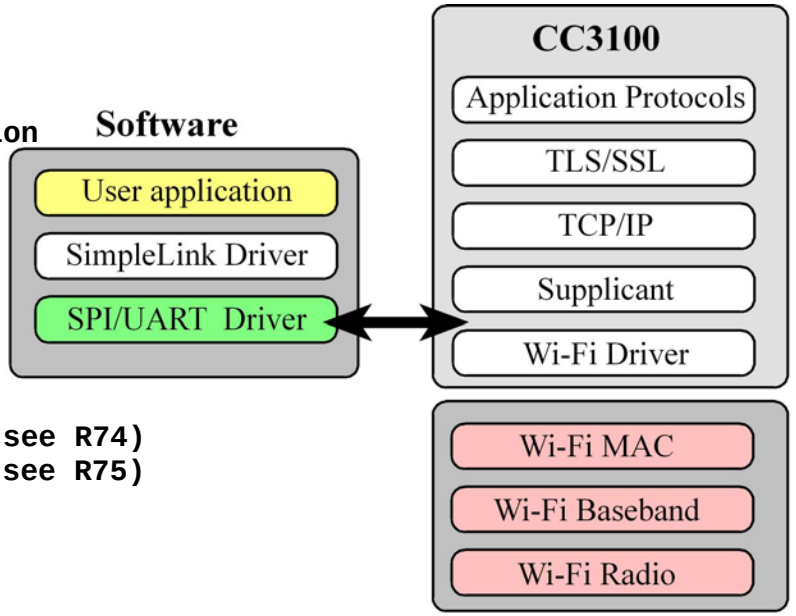


Pin	Signal	Direction
P1.1	3.3 VCC	IN
P1.2	PB5 UNUSED	NA
P1.3	PB0 UART1_TX	OUT
P1.4	PB1 UART1_RX	IN
P1.5	PE4 nHIB	IN
P1.6	PE5 UNUSED	NA
P1.7	PB4 SSI2_CLK	IN
P1.8	PA5 UNUSED	NA
P1.9	PA6 UNUSED	NA
P1.10	PA7 UNUSED	NA

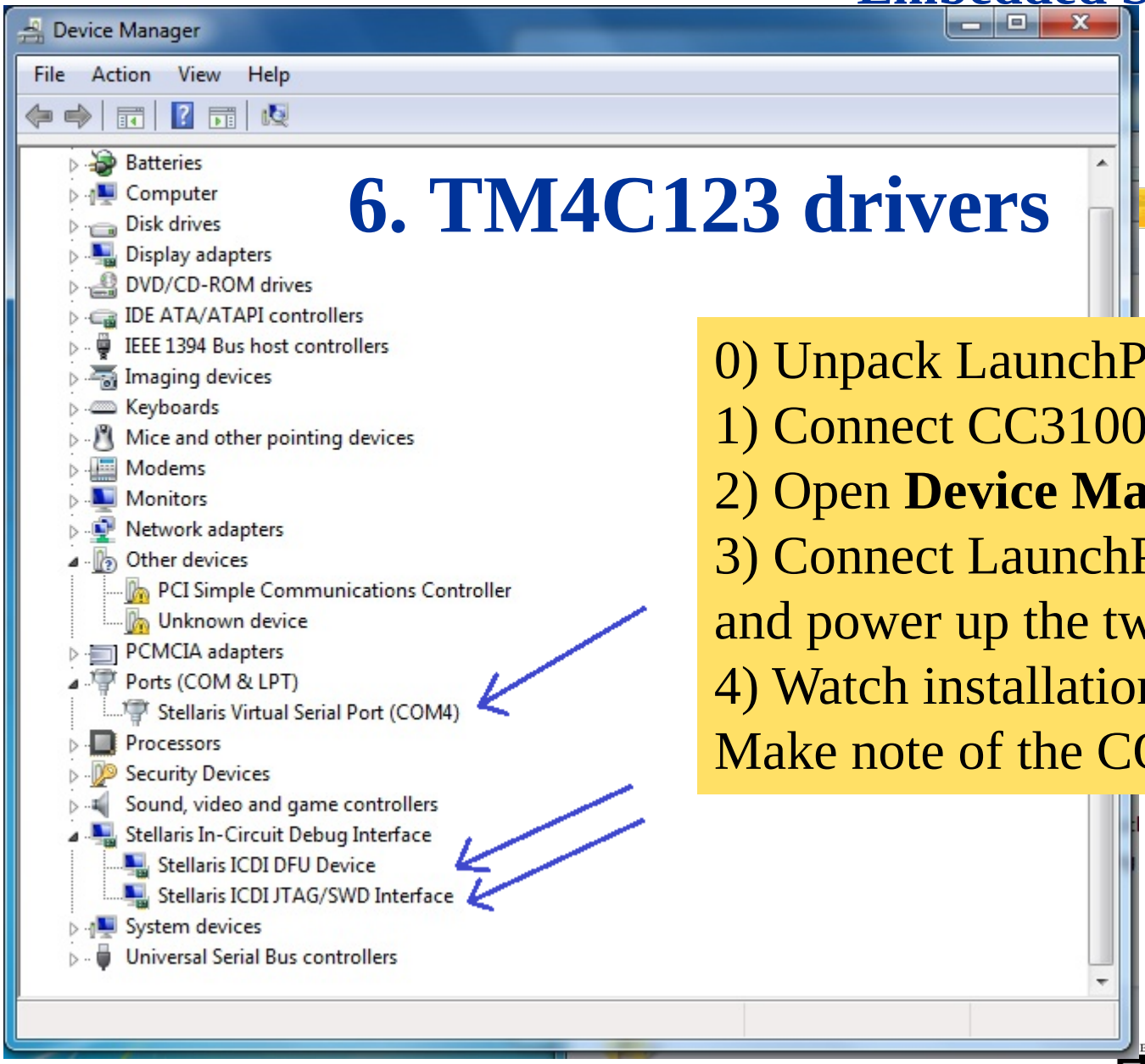
Pin	Signal	Direction
P2.1	Gnd GND	IN
P2.2	PB2 IRQ	OUT
P2.3	PE0 SSI2_CS	IN
P2.4	PF0 UNUSED	NA
P2.5	Reset nRESET	IN
P2.6	PB7 SSI2_MOSI	IN
P2.7	PB6 SSI2_MISO	OUT
P2.8	PA4 UNUSED	NA
P2.9	PA3 UNUSED	NA
P2.10	PA2 UNUSED	NA

Pin	Signal	Direction
P3.1	+5 +5 V	IN
P3.2	Gnd GND	IN
P3.3	PD0 UNUSED	NA
P3.4	PD1 UNUSED	NA
P3.5	PD2 UNUSED	NA
P3.6	PD3 UNUSED	NA
P3.7	PE1 UNUSED	NA
P3.8	PE2 UNUSED	NA
P3.9	PE3 UNUSED	NA
P3.10	PF1 UNUSED	NA

Pin	Signal	Direction
P4.1	PF2 UNUSED	OUT
P4.2	PF3 UNUSED	OUT
P4.3	PB3 UNUSED	NA
P4.4	PC4 UART1_CTS	IN
P4.5	PC5 UART1_RTS	OUT
P4.6	PC6 UNUSED	NA
P4.7	PC7 NWP_LOG_TX	OUT
P4.8	PD6 WLAN_LOG_TX	OUT
P4.9	PD7 UNUSED	IN (see R74)
P4.10	PF4 UNUSED	OUT(see R75)



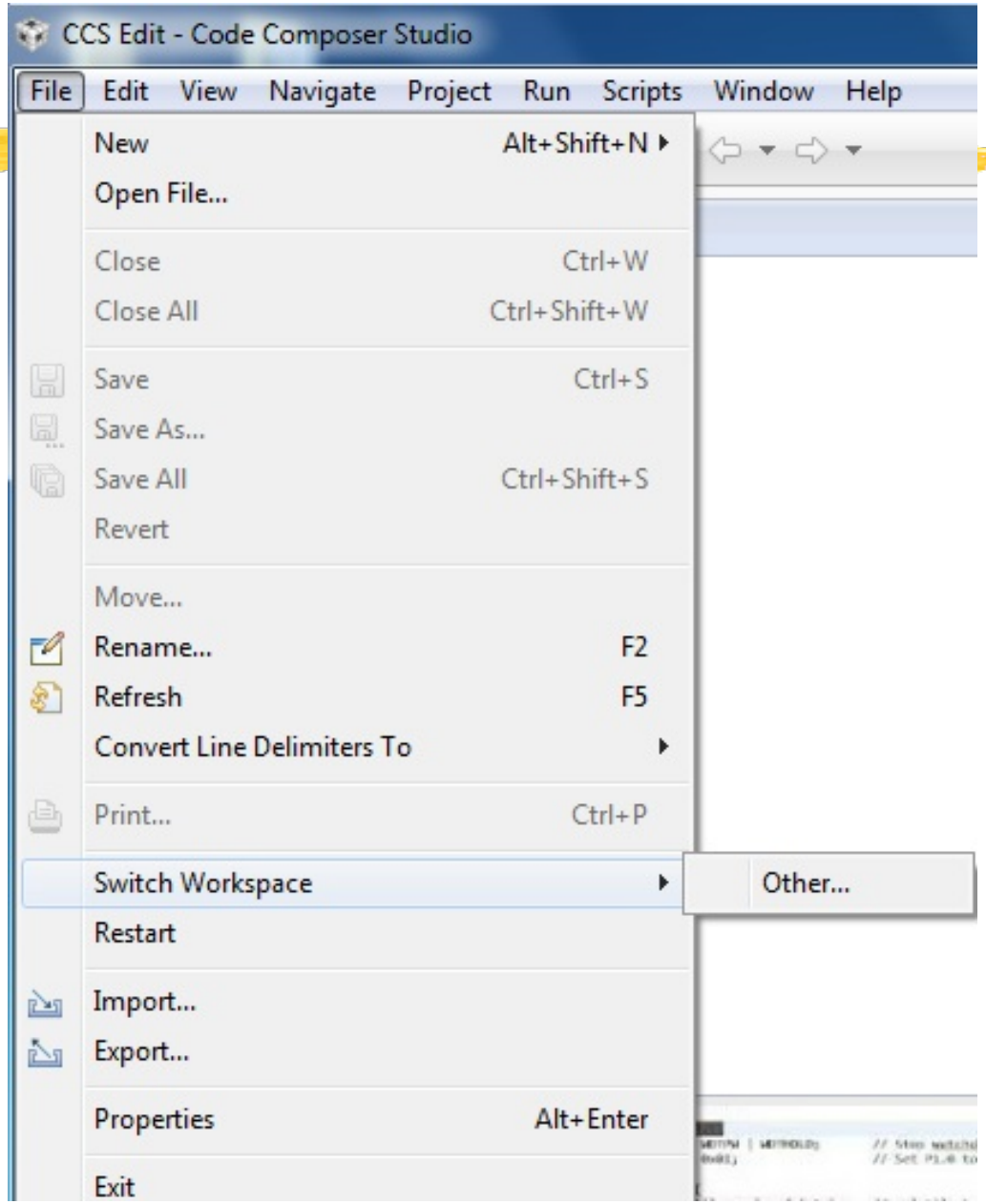
6. TM4C123 drivers



0) Unpack LaunchPad and CC3100
1) Connect CC3100 to LaunchPad
2) Open **Device Manager**
3) Connect LaunchPad USB to PC and power up the two boards
4) Watch installation
Make note of the COM port

6. Configure CCS

File->Switch Workspace
choose **Other…**

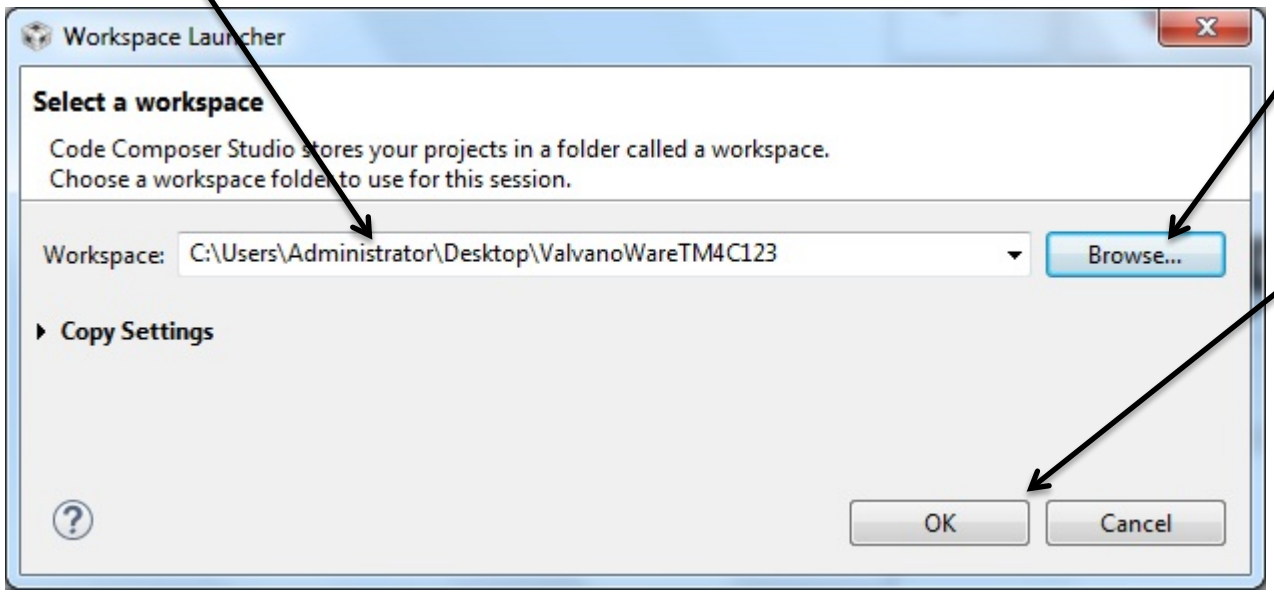


Jonathan Valvano

6. Configure CCS

1) Browse

2) Select
C:\Users\Administrator\Desktop\ValvanoWareTM4C123



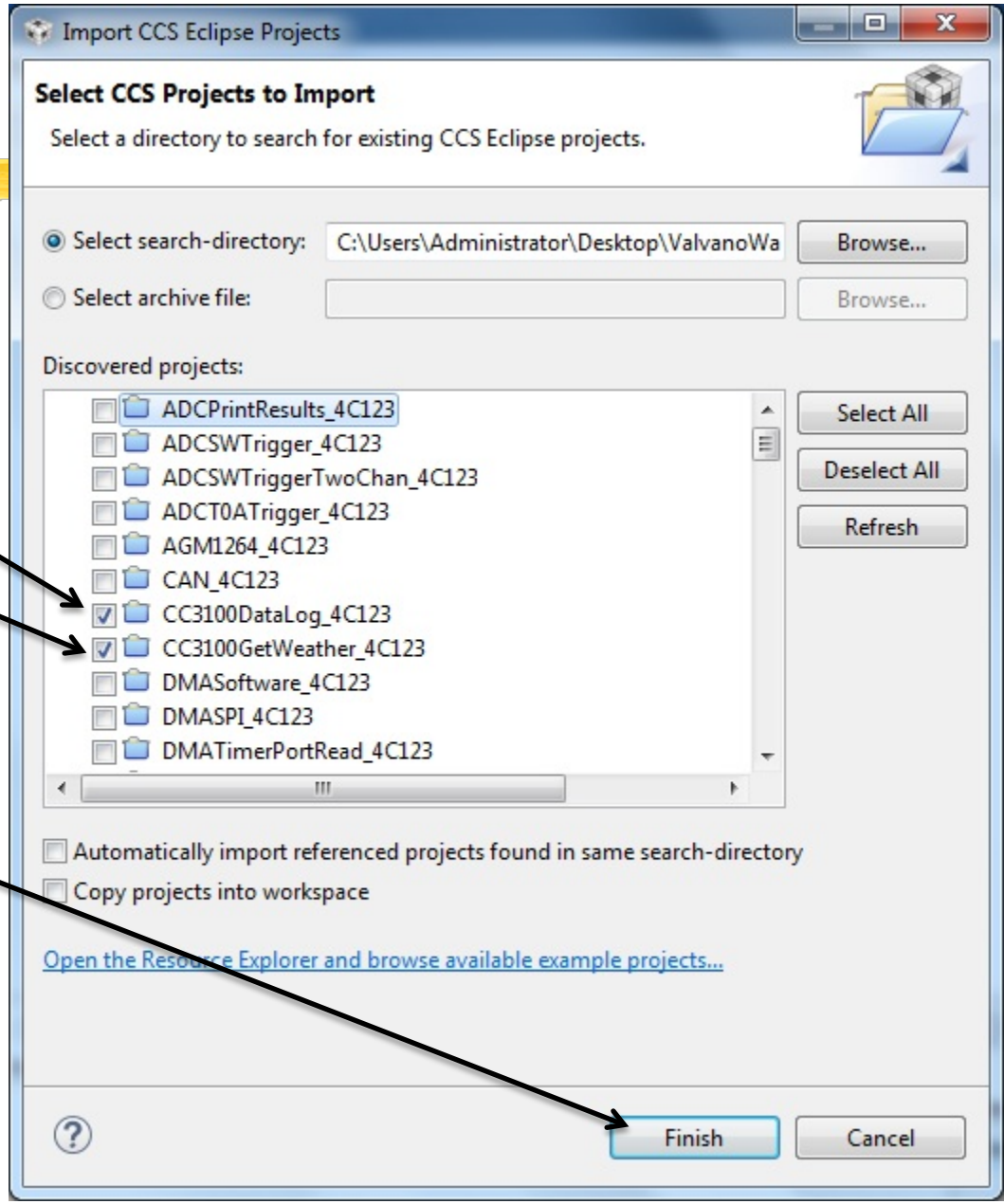
3) Ok

6. Configure CCS

1) File->Import

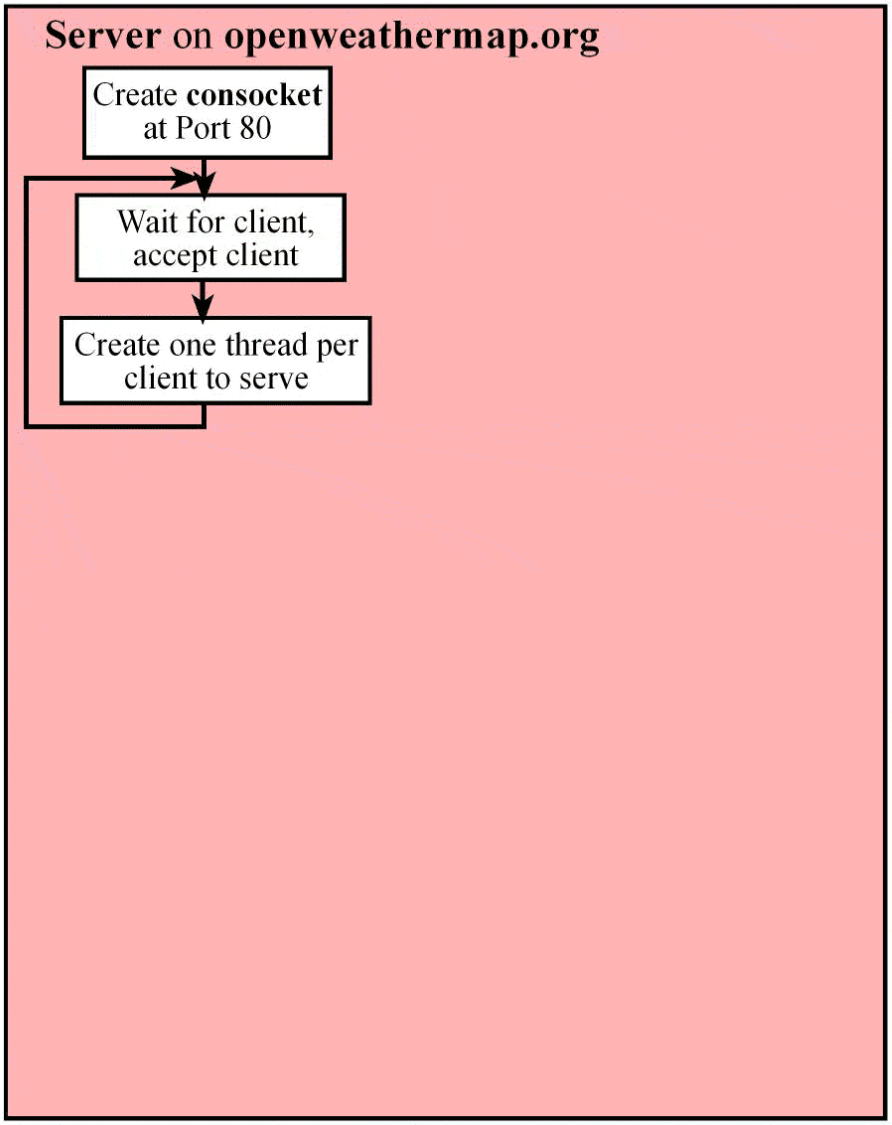
2) Click
CC3100DataLog_4C123
CC3100GetWeather_4C123

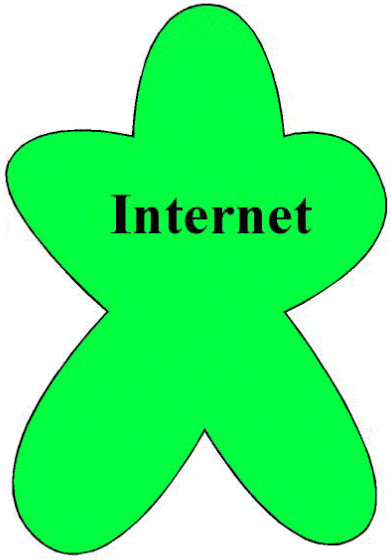
3) Click Finish



Jonathan Valvano

6. Get Weather



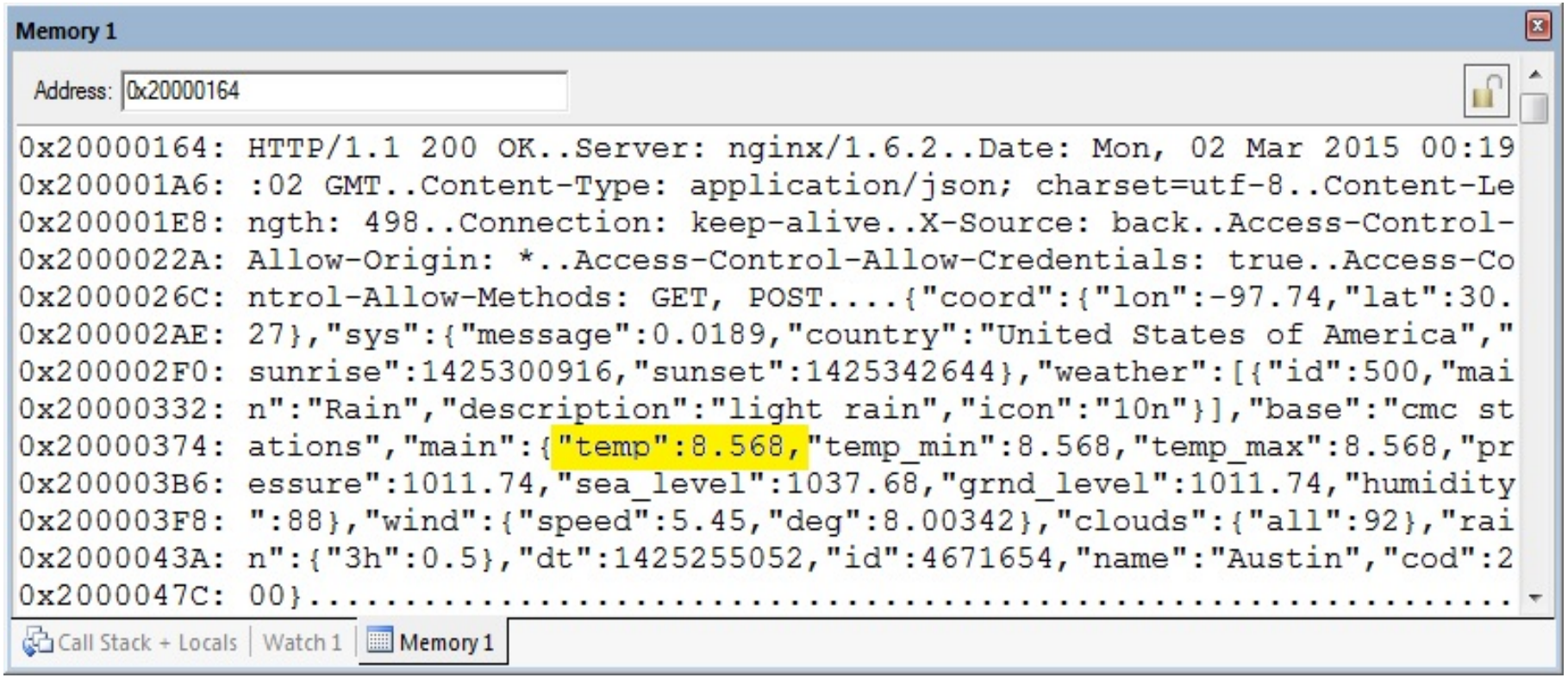


6. Get Weather (follow handout)

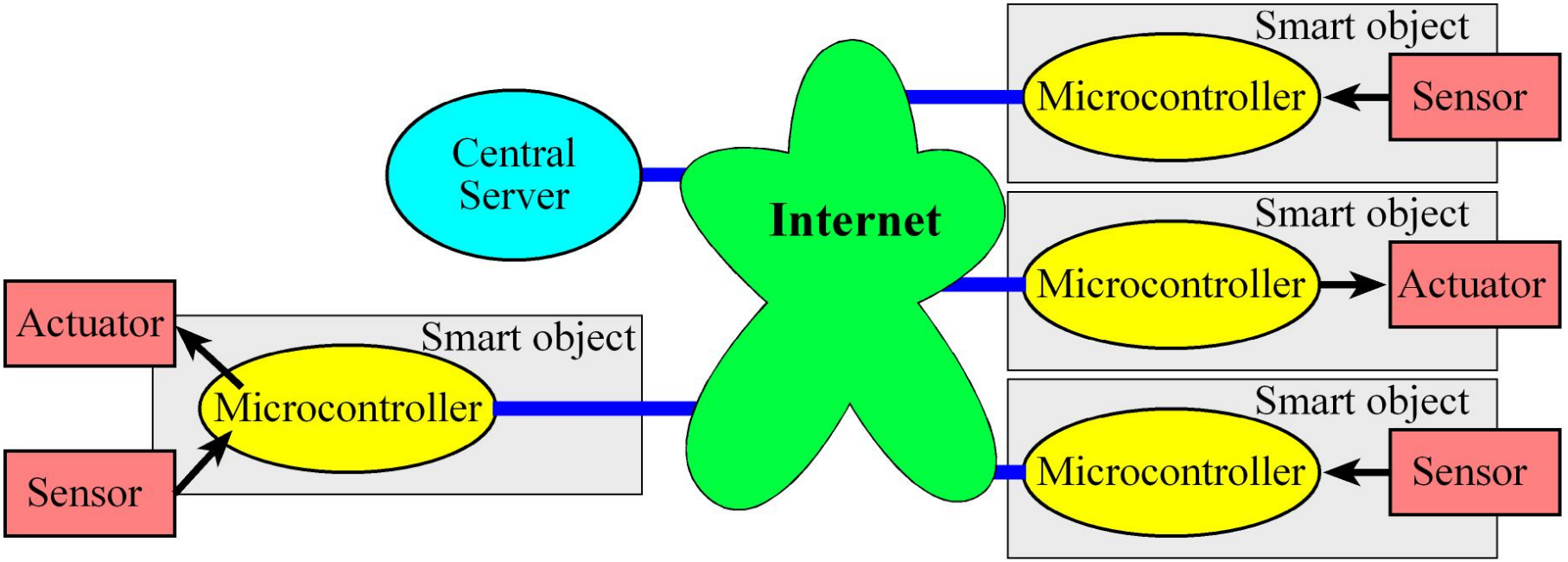
1) Open **Code Composer Studio** choose **ValvanoWareTM4C123**
2) Click on **CC3100GetWeather_4C123** project
3) Compile **Project->BuildProject**
4) Open **main.c** and edit lines 104, 105, 106
5) Compile again **Project->BuildProject**
6) Download and start debugger **Run->Debug**
7) Start **TExaSdisplay** (in **ValvanoWareTM4C123** folder)
8) Back to CCS, run program, check out weather in Austin
9) Overview fundamentals
   - Line 703 Connect to access point (name, password, type)
   - Line 670 Domain Name System (address to IP address)
   - Line 730 Send TCP (weather request)
   - Line 733 Receive TCP
10) Change line 98

6. See the Weather

1) TExaSdisplay
or 2) Memory look at 0x20000164, change mode to ASCII



6. Data Logger (follow handout)



6. Data Logger

1) Click on **CC3100DataLog_4C123** project
2) Compile Project->BuildProject
3) Open main.c and edit lines 138 and 140
4) Edit line 128 to specify city, id, greeting fields (leave ? & =)
5) Compile again Project->BuildProject
6) Download and start debugger Run->Debug
7) Start TExaSdisplay
8) Back to CCS, run program, logging your message into the server
9) Overview fundamentals (add breakpoint and observe input/output)
   - Line 746 Connect to access point (name, password, type)
   - Line 713 Domain Name System (address to IP address)
   - Line 691 Create Socket, open connection (IP address, port 80)
   - Line 832 Send TCP (log data to server)
   - Line 835 Receive TCP (receive acknowledgement from server)
10) Change line 128 to change message and log another data entry

6. Data Logger

MOOC server http://embsysmooc.appspot.com/

MOOC map http://embsysmooc.appspot.com/map

Jonathan Valvano