# Introduction: Outline & Logistics EEL 4745C: Microprocessor Applications II Fall 2022

Md Jahidul Islam

Last Modified: August 10, 2022



### This Course: uP2 (Microprocessor Applications II)

- ⇒ Implementation of a Real-Time Operating System (RTOS) on an ARM Cortex M processor
- ⇒ Introduction to single-board computing for IoT applications
- ⇒ Hands-on laboratory experience on RTOS and on-device programming

#### ⇒ Pre-Requisites

- EEL 4744C with minimum grade of C
- Fluent in C and assembly programming
- Proficiency in Python programming

#### ⇒ Fall 2022 Lectures

- M,W,F | Period 6
- MAEA 0303 | 12:50 PM 1:40 PM
- See ONE.UF for specific lab schedules





### **PIs:** Instructors and TAs

### Instructor: Md Jahidul Islam

- Office Hours: Friday 4:00 5:30 PM
- Email: jahid@ece.ufl.edu



### TA: Xuanhao Shi

- ⇒ Office Hours
  - M/W 10:40 11:30 AM
  - NEB 281 / NEB 222

#### ⇒ Email: <u>xshi1@ufl.edu</u>



### TA: Kevin D Mcgrath

#### ⇒ Office Hours

- Tu/Th 12:50 1:40 PM
- NEB 281 / NEB 222
- ⇒ Email: <u>kevin.mcgrath@ufl.edu</u>





### uP2: Then and Now

#### ⇒ Basic concepts of RTOS and ARM Cortex M4 processors

⇒ Programming RTOS components: THMSP432 LaunchPad TI Tiva C Series LaunchPad

- Implementing threads and schedulers
- Handling inter-process communication and semaphores
- Interfacing driver libraries for integrated peripherals
  - I2C RGB LEDs drivers, joysticks, LCD touchscreen display
- Mastering a IoT development board (modified daughter-board)
- Interfacing external IoT sensor boards
  - TI sensor booster pack

#### ⇒ Creating on-device IoT applications

- Interfacing CC3100 Wi-Fi chip with MSP432 TI BeagleBone Black Board
- Implementing a two-player game over WiFi on-device AI and TinyML applications



4

### People Who Helped!



Ghanshyam Sookai



Patrick Argento



Boxiao Yu



Michael V. Stapleton



Dr. Yier Jin



Dr. Eric Schwartz



Dr. Karl Gugel



### **Required Resources**

#### ⇒ Major hardware components

- TI Tiva C Series LaunchPad and TI Sensors Booster Pack
- IoT Development Board (with LEDs, LCD touch display and joystick)
- TI BeagleBone Black Board (modified daughter-board)

#### ⇒ Software platforms and environments

- TI Code Composer Studio 11 (and TivaWare)
- Beagle-Board firmware image (and OpenCV)
- Some relevant libraries and source code (will be provided in class)

#### ⇒ Textbook (pdf available online)

 Real-Time Operating Systems for ARM Cortex-M Microcontrollers (4th Edition) By Jonathan W Valvano. ISBN-13: 978-1466468863, ISBN-10: 1466468866.



### **Other Resources**

- ⇒ Tiva<sup>™</sup> TM4C123GH6PM Microcontroller: datasheet and user manual
- ⇒ Tiva™LP3943 LED Driver and Booster-pack datasheet
- ⇒ Tiva<sup>™</sup> BeagleBoard CookBook: beyond BeagleBone Black
- ⇒ The Digilent Analog Discovery 2 (DAD) board:
  - Available from the UF Bookstore; also in DigiKey / Adafruit
  - If you already have one from uP1/DSP courses that will do! (\*not a must have)

⇒ Operating system

- Our lectures/materials are based on Windows
- Mac/Linux should also work with the standard driver-level adjustments
- <u>Note</u>: some Linux distributions or newer Mac M1/M2 have some compatibility issues; in such cases, feel free to use virtual environments!





## **Grading Breakdown**

Item	Points	% of Final Grade
Hands-on Laboratory (lab1 - lab4)	4 x 10	40
Hands-on Laboratory (lab5 - lab6)	2 x 7.5	15
Mid-term	1 x 15	15
Milestone Quizzes (q1 - q3)	3 x 05	15
Final exam / project option	1 x 15	15
Total	100	

#### ⇒ Expectations and evaluation criteria

- Hands-on expertise toward learning RTOS concepts
  - Applying those concepts into working "G8RTOS" functionalities
  - Given datasheets, user manuals, and a sketch of the solution
- How to put together individual working components into an Al/IoT application

Make hundred mistakes, it'll be fine - just don't repeat the same mistakes



8

## **Grading Policy**

Percent	Grade	Grade Points
90 or More	A	4.00
86.0 - 89.9	A-	3.67
82.0 - 85.9	B+	3.33
79.0 - 81.9	В	3.00
76.0 - 78.9	B-	2.67
73.0 - 75.9	C+	2.33
70.0 - 72.9	С	2.00
66.0 - 69.9	C-	1.67
63.0 - 65.9	D+	1.33
60.0 - 62.9	D	1.00
55.0 - 59.9	D-	0.67
Below 55	E	0.00

#### ⇒ Grading and reporting

- Grades are periodically posted online
  - Please check your grades regularly
  - Grades are final after one week of posting
- Checkout the general UF grading policy

#### ⇒ Final rubric and grade assignment

- Lab grades are final once graded
- Mid-term, quizzes, and final exam grades will be curved if necessary

Pay attention to the feedback, and be present!





### Laboratory Outline: Lab 0-1-2

⇒ Lab 0 - Getting Things Ready [Week 1]

- Part A: Introduction and setup
- Part B: Blinking TIVA C on-board LEDs

⇒ Lab 1 - Basic Interfacing, Linking, and Communication [Week 1-2]

- Part A: Interfacing LED drivers, I2C communication
- Part B: ARM assembly checksums with the LED driver
- Part C: Basic UART communication for console I/O

#### ⇒ Lab 2 - G8RTOS Scheduler and Synchronizers [Week 3-5]

- Part A: Setting up board support package and drivers
- Part B: Creating basic OS structure
- Part C: Implementing threads and schedulers
- Part D: Implementing semaphores and peripheral controls
- Part E: Putting it all together!

### Laboratory Outline: Lab 3-4-5



- Part A: Implementing blocking, yielding, and sleeping
- Part B: Integrating periodic threads with background threads
- Part C: Enabling inter-process communication with FIFOs

Mid-term review, exam, and discussions [Week 8]

#### ⇒ Lab 4 - Dynamic Threads and LCD Interfacing [Week 9-10]

- Part A: Interfacing a touchscreen color LCD
- Part B: Incorporating aperiodic/dynamic threads in RTOS
- Part C: LCD interfacing based on sensory inputs

#### ⇒ Lab 5 - Dynamic Threads and LCD Interfacing [Week 11-12]

- Part A: Interfacing a touchscreen color LCD
- Part B: Incorporating aperiodic/dynamic threads in RTOS





### Laboratory Outline: Lab6 + Project

#### ⇒ Lab 6 - On-device AI; Project Options: [Week 13-14]

- Handwritten zip code recognition
- Face or audio keyword detection
- Two-player tic-tac-toe or
- Security camera feature integration
- Students may propose other options

#### Final exam / project showcase [Week 15]

- ⇒ You have a lot of freedom in Lab6 and project options
- ⇒ You can choose between final exam or personal project

Apply what you learn, have fun!







## **Laboratory Guidelines**

⇒ Timeliness and participation: don't miss any lab!

- Be present at the lab 5/10 minutes earlier
- Keep the lab worksheet with you
- Be organised, take notes, and don't lose hw

⇒ Honesty and integrity: don't cheat yourself!

- No place for any form of plagiarism in this course
- Seek help and collaborate with integrity
- We trust you, and we'll make sure nobody gets unfair/dishonest advantage

 $\Rightarrow$  Safety: don't put yourself & others in danger!

- Take soldering measures you learned in uP1
- If you are not sure, ask we are here to help!
- Report anything that needs attention

What to do if you are sick with Covid / *cannot* make it in-person?

#### What if you are stuck with no luck?

How to get help and collaborate without the risk of plagiarism?

What if you burn/damage something?



# Lab0: Introduction & Setup Soldering PCBs & Blinking Tiva

Hardware Setup







### **Get the Base PCB Ready**



Some are already soldered by us (we tested random samples)



EEL 4745C: Microprocessor Applications II



UF FLORIDA

### **Get the Modified PCB Ready**





16

### **Tricks for Good Alignment**



- **2.** Plug the modified PCB like this and solder the pins
- **3.** Then use a breadboard to solder male-male pins to the other side, for connecting Tiva/sensor-board & LCD headers





17

UF FLORIDA

### **Connect Tiva + Sensor board + LCD**





EEL 4745C: Microprocessor Applications II



### **Put Everything Together**





EEL 4745C: Microprocessor Applications II



### We Are All Set!







# Lab0: Introduction & Setup Soldering PCBs & Blinking Tiva

Software Setup







### Install CCS - Code Composer Studio

#### ⇒ Install Code Composer Studio (CCS) 11.1.0

- Download link: https://software-dl.ti.com/ccs/esd/documents/ccs\_downloads.html
- Instructions: https://software-dl.ti.com/ccs/esd/documents/users\_guide/ccs\_installation.html
- Select the custom install
- Select component: TM4C12x Arm Cortex M4F-based MCUs
- Debug probe: Stellairs\*
- For everything else: use the default choices and complete the installation

#### ⇒ Create a Workspace named CCS in your preferred directory

• Play around with basic example projects: build and run

#### ⇒ Install Tivaware

- Download: https://www.ti.com/tool/SW-TM4C
- Instructions: https://www.ti.com/lit/ug/spmu373/spmu373.pdf





22

## **Create Your Project**

File	Edit View Navigate Project Run Scripts Wi	ndow	Help	
	New Alt+Shift+N >	12	CCS Project	
0	Open File Open Projects from File System Recent Files	C C h	Project Create a new C/C++/ASM project and let CCS create Source Header File	nd 🗖
	Close Editor Ctrl+W	G	Class File from Template Folder Target Configuration File DSP/BIOS v5.x Configuration File	
	Close All Editors Ctrl+Shift+W	Ľ		
	Save Ctrl+S Save As Save All Ctrl+Shift+S			
	Revert	162	Other Child	
	Move		other Ctri+N	
	Rename F2			
88	Refresh F5 Convert Line Delimiters To >			
۵	Print Ctrl+P			
	Import Export			
	Properties Alt+Enter			
	Switch Workspace > Restart Exit	3	×   & & S 🖬 🖬	

#### ⇒ Keep things organized

- Maintain a clean workspace
- Use separate projects for separate tasks

	Tiva C Se	eries	~	Tiva TM4C123	GH6PM	<i></i>	
onnection:					~	Verify	
😭 Cortex	M [Arm]						
Project na	me:	Lab0_Blink					
Use def	fault locati	ion					
Lo	ocation:	C:\Users\Xahid\Docume	ents\CCS_Tr	/a\Lab0_Blink		Browse	
Compiler	version:	TI v20.2.5.LTS			~	More	
<ul> <li>Project t</li> <li>Project t</li> </ul>	type and to emplates	ool-chain and examples					
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> </ul>	type and to emplates r text	ool-chain and examples	Creat	es an empty pr	oject initialized	d for the 🔥	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>Em</li> </ul>	type and to templates r text npty Proje	ool-chain and examples	Creat selec empt	es an empty pro ted device. The y 'main.c' source	oject initialized project will co ce-file.	d for the 🔥 ^ ntain an	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>Enr</li> <li>Enr</li> </ul>	type and to templates r text npty Proje Empty Pr Empty Pr	ool-chain and examples cts oject oject (with main.c)	Creat selec empt	es an empty pro ted device. The y 'main.c' sourc	oject initializeo project will co ce-file.	d for the 🛛 ^ ntain an	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>En</li> </ul>	type and to templates r text npty Proje Empty Pr Empty Pr Empty As Empty RT	ool-chain and examples cts oject oject (with main.c) isembly-only Project SC Project	Creat selec empt	es an empty pri ted device. The y 'main.c' sourc	oject initializec project will co ce-file.	d for the 🛛 ^ ntain an	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>Em</li> <l< td=""><td>type and to templates r text npty Proje Empty Pr Empty As Empty RT isic Examp</td><td>col-chain and examples cts oject oject (with main.c) isembly-only Project SC Project les</td><td>Creat selec empt</td><td>es an empty pri ted device. The y 'main.c' sourc</td><td>oject initialize project will co ce-file.</td><td>d for the Antain an</td></l<></ul>	type and to templates r text npty Proje Empty Pr Empty As Empty RT isic Examp	col-chain and examples cts oject oject (with main.c) isembly-only Project SC Project les	Creat selec empt	es an empty pri ted device. The y 'main.c' sourc	oject initialize project will co ce-file.	d for the Antain an	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>En</li> <li>Ba</li> <li>Ba</li> <li>TI-</li> </ul>	type and to templates r text mpty Proje Empty Pr Empty Pr Empty As Empty RT sic Examp Hello Wo RTOS Exar	ool-chain and examples cts oject oject (with main.c) seembly-only Project SC Project iles rld mples	Creat selec empt	es an empty pr ted device. The y 'main.c' sourc	oject initializec project will co ce-file.	d for the Antain an	
<ul> <li>Project t</li> <li>Project t</li> <li>type filter</li> <li>En</li> <li>En</li> <li>Ba</li> <li>En</li> <li>TI-</li> </ul>	type and to templates r text mpty Proje Empty Pr Empty Pr Empty As Empty RT isic Examp Hello Wo -RTOS Exar	ool-chain and examples cts oject oject (with main.c) sembly-only Project SC Project les rld mples	Creat selec empt	es an empty pr ted device. The y 'main.c' sourc	oject initializec project will co ce-file.	d for the hand	



UF FLORIDA

## Add TivaWare and Build Project

1973 - 1973 - 1973			😚 CCS_Tiva - Lab0_Blink/main.c - Code Compo	oser Studio	
type filter text	Include Options		File Edit View Navigate Project Run	Scripts Window Help	
> Resource			📑 • 🔚 🐚 📮 🕸 • 🙆 • ≼ • )	Ø 🛷 • 🕼 🔳 🏷 🖓 (> • <>	-
v Build	Configuration: Debug [ Active ]	Manage Configurations	Project Explorer 💥 🖵 🛙	🗆 🖻 main.c 🔀	- 6
<ul> <li>Arm Compiler Processor Options Optimization Include Options ULP Advisor Predefined Symbols</li> <li>Advanced Options</li> <li>Arm Linker Arm Hex Utility [Disabled]</li> <li>Debug</li> </ul>	Add dir to #include search path (include_path, -1) S(PROJECT_ROOT) :::::::::::::::::::::::::::::::::::	<b>ହ କା ହା</b> ହା ହ କା ହ ଚିତ୍ୟ ହା ତ କା କା ତ ଚିତ୍ୟ ହା	► S  ► Sim A ► Sim A ► Simaries ► Debug ► Simaries ► Debug ► Simaries ► Debug ► Simaries ► Debug ► Simaries ► Simaries ► Simaries ► Simaries ► Debug ► Simaries <p< td=""><td><pre>8 1 2 #include "inc/tm4c123gh6p 3 /** 4 * main.c 5 */ 6 int main(void) 7 { ic 8 return 0; 9 } 10</pre></td><td>n. h"</td></p<>	<pre>8 1 2 #include "inc/tm4c123gh6p 3 /** 4 * main.c 5 */ 6 int main(void) 7 { ic 8 return 0; 9 } 10</pre>	n. h"
				<	>
				E Console 🔀	🗶 🕂 😯 😫 📰 🔐 = 🚉 🛃 🚽 🗨 😁 - 🗆
				CDT Build Console [Lab0_Blink]	
				Invoking: Arm Linker "C:/ti/ccs110/ccs/tools/comp <linking> Finished building target: "La</linking>	<pre>iler/ti-cgt-arm_20.2.5.LTS/bin/armcl" -mv704code_state=16 - b0_Blink.out"</pre>
Show advanced settings		Apply and Close Cancel	(	**** Build Finished ****	

#### ⇒ Project properties > Include options

- Include TivaWare path and any other BSPs (Board Support Packages) you will use (later)
- If you haven't used CCS before, try to familiarize yourself with these options!



UF FLORIDA

### **Blinking On-board Tiva LED**



#### ⇒ Build and flash the code!

- See how the GPIO pins are interfaced
- Play around with different options

#### *⇒* Bonus Point!

#### +1

- Control the LED with on-board switches
- On-off or faster-slower blinks





### A Closer Look





EEL 4745C: Microprocessor Applications II

26

UF FLORIDA

### **General Suggestions!**





EEL 4745C: Microprocessor Applications II

