Embedded Controller programming for Real Time Systems: ECE-40097

Course pre-requisite Information

Vijay Kumar

Main Topics

Prerequisite

Course Structure

Course Description

Course Outcomes

References

Course Wrap-up

Prerequisite

- Completion of course
  - ECE-40291 Embedded Controller Programming with Embedded C
  - ECE-40292 Embedded Systems Hardware Design
  - Or equivalent knowledge and experience
- Working development kit
  - STM32 IOT Discovery Node 915MHZ - Mfg. Part # B-L475E-IOT01A
- IDE installed on PC
  - STM32CubeIDE
    - https://www.st.com/en/development-tools/stm32cubeide.html
- Setup environment for SW development
  - Strongly recommend to test the board and ST-Link connectivity with any test software.
  - Either from previous course or my sample code
  - There are no programming assignment until after second lesson
- **Should go through the syllabus**

Course Structure

- Nine lessons
- One Quiz
- Few programming assignments
- Mid-term/Final assignment
- Discussion board topics – Each week

Assignment Guidelines

- Do create a discussion board if you have generic questions and other students could benefit
  - Otherwise, you could ask me directly
- Strongly recommend to design the first programming assignment right
  - You will build next assignment on top of the first assignment
- It is important to understand the concept and not just working code
  - This will take you long way in future
- Assignments are primarily in C but in few cases, you must write in assembly

Course Structure

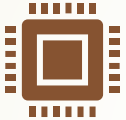
Lesson	Topic
1	Introduction to Real Time Embedded system
2	Microcontroller Architecture
3	Microcontroller Instruction sets
4	Introduction to assembly language
5	Introduction to Interrupts
6	Interrupt Programming
7	Timers
8	RTC/DMA
9	Embedded System design

Text Book

- I do have textbook for this course
  - Expect your feedback and comments
  - Do let me know if you come across any books for my evaluation

Course Description

Is advanced programming course



Will cover real-time event-driven applications with instant and reliable access to systems resources for embedded microcontrollers.

Will learn design of firmware that will respond, within a given time, to a multitude of possible inputs

Any of which may arrive at unpredictable times and in an unpredictable sequence

Will learn about low-level microcontroller programming,, interrupt-driven programming, GPIO's, timers and RTC.

Will briefly cover DMA and its use

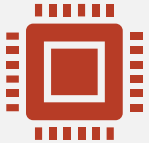
Course Outcome

Understand controller (Arm Coretx-M4), controller peripherals, programming the controller in C and Assembly language, Interrupts and Timers.

Provide hands on experiences of writing interrupt service routines (ISR), and subroutines in Assembly language

Develop practical knowledge of designing, and development of embedded software for real time embedded systems. This includes the best design practice with and without operating system.

Course Objective



Learn about interrupt controller and will be able to read vector table and write interrupt service routines for GPIO and reset button.

Learn about devices programming to use for application such as delay, counters, alarms, watchdog.

Learn about event driven embedded design and will be able to write software based on events, e.g., character received on UART, pressing reset button on the development Kit

References

- https://www.st.com/en/evaluation-tools/b-l475e-iot01a.html
- https://www.st.com/content/ccc/resource/technical/document/programming_manual/6c/3a/cb/e7/e4/ea/44/9b/DM00046982.pdf/files/DM00046982.pdf/jcr:content/translations/en.DM00046982.pdf
- https://developer.arm.com/documentation/
- https://www.st.com/resource/en/datasheet/stm32l475vg.pdf

Course Wrap-up

- Should be ready for next required course
  - **Embedded Real-Time Operating System (RTOS)**(ECE-40290)
  - Or other related courses
- Ready to design/enhance or work on any embedded system and/or software
- Ability to select right language and microcontroller for a real time system design