13B1WCI731: ARM Based Embedded System Design

Course Credit: 3
Semester: VII

Introduction

ARM designs the world's most popular processor architecture for embedded systems. The technology can be found at the heart of advanced digital products, from wireless, networking and consumer entertainment solutions to imaging, automotive, security and storage devices. This course is designed to provide the necessary knowledge to develop software for an ARM based system targeted in embedded software development. Topics cover the software aspects of ARM system which including assembly and C language programming and an introduction to the control and interfacing of embedded systems. Upon completion, students will understand the ARM architecture development, and will be familiar with supporting hardware and embedded system development process.

Course Objectives (Post-conditions)
Knowledge objectives:

1. Describe the architecture of a typical embedded RISC processor (e.g. ARM Cortex-M3)
2. Develop an understanding of the instruction set and addressing modes
3. Write programs that exercise a range of typical microcontroller peripherals (e.g. GPIO, USART, ADC, etc...)
4. Use a typical toolchain to implement and test simple embedded microcontroller applications in C and assembly language
5. Evaluate the requirements for embedded Real Time Operating Systems (RTOS)
6. Understand the structure of a RTOS (e.g. eLinux)

Application objectives:

1. Introduction to embedded systems, overview of the design flow
2. Embedded system specification and modeling
3. Embedded hardware platforms and peripherals
4. Interfacing to the external world through sensors and actuators
5. Design and synthesis of ASIC hardware
6. Software organization, scheduling, and execution
7. Embedded and real-time operating systems
8. Wired communication and bus protocols
9. Basics of wireless communication and embedded networking
10. Energy management and low-power design
11. Safety and reliability in embedded systems
12. Secure embedded system design

Expected Student Background (Preconditions)
C/C++ Programing, Assembly programming, Computer architecture, Digital design

Topics Outline:

S NO	Topics	Hrs
1	Computer Architecture Introduction	4
2	ARM Software Development Tools and ARM Architecture Fundamentals	4
3	ARM Instruction Set Architecture and Addressing Mode	5
4	ARM Assembly Language Programming Basics	4
5	More on ARM Assembly Language Programming	3
6	The Thumb Instruction Set	4
7	Exception and Interrupt Handling	3
8	ARM Processor Architecture	4
9	Startup Sequence and ARM Based Hardware Consideration	2
10	Introduction to Debug Tools for ARM Based Systems	2
11	ARM Test Revision	2
12	ARM on virtual machine	2
13	ARM Exam Revision	2
	Total	41

References

1. ARM Architecture Reference Manual (local copy)
   ARM7TDMI Technical Reference Manual (local copy)
   ARM7TDMI Instruction Set Reference (local copy)
2. ARM7TDMI Quick Reference (local copy)
3. An Introduction to the GNU Assembler
4. An Introduction to the GNU Compiler
5. An Introduction to GNU Debugger
6. ARMv4T Partial Instruction Set Summary
   Pete Cockerell has put the text of his 1987 book "ARM Assembly Language Programming" on the web (local copy)
7. A book by Peter Knaggs and Stephen Welsh, ARM: Assembly Language Programming @2004 - download for free (local copy)
8. ARM System-On-Chip Architecture (2nd Edition) by Steve Furber (the father of the ARM processor) - download for free (local copy)

Evaluation Scheme:

S.No	Examination	Marks
1	T-1	15
2	T-2	25
3	T-3	35
4	*Internal Marks	25

*Internal Marks Breakdown:

Assignments 9 marks (3x3)

Quizzes 12 marks (3x4)

Regularity 4 Marks