17ES010 RISC PROCESSORS ARCHITECTURE AND PROGRAMMING

Hours Per Week :

L	T	P	C
3	1	-	4

Total Hours :

L	T	P
45	15	-

WA/RA	SSH/HSH	CS	SA	S	BS
15	30	-	5	5	-

Course Objectives:

The RISC concept has led to a more thoughtful design of the microprocessor. Among design considerations are how well an instruction can be mapped to the clock speed of the microprocessor (ideally, an instruction can be performed in one clock cycle).The ARM architecture is the industry's leading 16/32-bit embedded RISC processor solution. ARM Powered microprocessors are being routinely designed into a wider range of products than any other 32-bit processor. This wide applicability is made possible by the ARM architecture, resulting in optimal system solutions at the crossroads of high performance, low power consumption and low cost.

- To understand the embedded system based on ARM processor and itshardware (ARM processor Core).
- To understand the techniques and rules for writing efficient C code andoptimizing ARM assembly code.
- To discuss various Cache technologies and Architecture that surrounds theARM cores and MMU.
- To Understand the architecture of ARM CORTEX-M3

Course Outcomes:

The student will be able to:

- Design an embedded system using ARM processor.
- . Write source code that will compile more efficiently in terms of increasedspeed and reduced code size.
- Develop an embedded system with optimized key subroutines to reduce system power consumption and clock speed needed for real time operation

SKILLS :

- 16/32 bit assembly language programming for ARM processor
- 16/32 bit C language programming for ARM processor
- Difference between RISC and CISC

ACTIVITIES:

- *Able to analyze design requirements.*
- *Multi byte(32/ 64/128) addition by using arm assembly language/C*
- *64 bit multiplication by ARM assembly language/C*
- *Keypad interfacing with ARM processor*

4. *LOAD and STORE operations (Address verifications)*

UNIT – I

ARM ARCHITECTURE : ARM Design Philosophy, Registers, Program Status Register, InstructionPipeline, Interrupts and Vector Table, Architecture Revision, ARM ProcessorFamilies.

UNIT – II

ARM PROGRAMMING MODEL – I : Instruction Set: Data Processing Instructions, Addressing Modes, Branch,Load -Store Instructions, PSR Instructions, Conditional Instructions.

UNIT – III

THUMB INSTRUCTION SET : Register Usage, Other Branch Instructions, DataProcessing Instructions, Single-Register and Multi Register Load-StoreInstructions, Stack, Software Interrupt Instructions

UNIT – IV

ARM PROGRAMMING : Simple C Programs using Function Calls, Pointers, Structures, Integer andFloating Point Arithmetic, Assembly Code using Instruction Scheduling,Register Allocation, Conditional Execution and Loops. Exception Handling ,Interrupts , Interrupt handling schemes, Firmware and boot loader.

UNIT – V

ARM CORTEX-M3 : ARM Cortex-M3 Processor –Architecture- Instruction Set Development-TheThumb-2 Technology and Instruction Set Architecture-CORTEX-M3Applications.

TEXT BOOKS:

1. Andrew N. Sloss, Dominic Symes, Chris Wright, John Rayfield 'ARM System Developer's Guide Designing and Optimizing System Software', Elsevier 2007
2. John H. Davies, "MSP430 Microcontroller Basics", Newnes (Elsevier Science), 2nd Edition, 2008.
3. ARM System on Chip Architecture – Steve Furber – 2nd ed., 2000,Addison Wesley Professional.
4. The indefinite guide to ARM CORTEX-M3.
5. Joseph Yiu "The Definitive Guide to the ARM Cortex-M0", Newnes, (Elsevier), 2011.

REFERENCE BOOKS:

1. Steve Furber, 'ARM system on chip architecture', Addision Wesley
2. ARM Architecture Reference Manual
3. LPC213x User Manual
4. ARM System developers guide-ELSEVIER publications