



Unit - I

Chapter 1 : ARM Introduction	1-1 to 1-16
1.1 Types of Computer Architectures	1-1
1.1.1 Von Neumann Memory Organization	1-1
1.1.2 Harvard Memory Organization	1-2
1.2 ISA's and ARM	1-2
1.3 Difference between RISC and CISC	1-3
1.4 RISC Design Philosophy	1-4
1.4.1 Register Window	1-4
1.4.2 Miscellaneous Features of RISC Systems	1-5
1.5 ARM Design Philosophy	1-8
1.6 History of ARM Microprocessor	1-8
1.6.1 Terms Related to ARM Instruction Set	1-11
1.7 ARM Processor Family	1-12
1.8 Development of ARM Architecture	1-13
1.9 Embedded System Hardware and Software	1-15
1.9.1 Embedded System Hardware	1-15
1.9.2 Embedded System Software	1-16

Unit - II

Chapter 2 : ARM Architecture and Pipeline Structure	2-1 to 2-45
2.1 The Acorn RISC Machine	2-1
2.2 ARM Core Data Flow Model	2-1
2.2.1 5 Stage ARM9 Core Architecture	2-1
2.2.2 Understanding the ARM Instruction Execution	2-2
2.2.3 Execution of Data Processing Instruction in ARM	2-3
2.2.4 Execution of the Branch Instructions in ARM Processor	2-4
2.3 Architectural Inheritance	2-5
2.4 The ARM7TDMI Programmer's Model: General Purpose Registers, CPSR, SPSR	2-7
2.4.1 General Purpose Registers	2-10
2.4.2 Program Status Registers (CPSR and SPSR)	2-12
2.4.3 Barrel Shifter	2-15
2.5 ARM Memory Map#	2-16
2.6 Data Format	2-17
2.7 Load and Store Architecture	2-18
2.7.1 Addressing Modes for Data Processing Operands (i. e. op1)	2-18
2.7.2 Addressing Modes for Memory Access Operands	2-19
2.8 ARM 3 and 5 Stage Pipeline	2-21
2.9 Pipeline Hazards	2-24
2.9.1 Methods to Resolve the Data Hazards and Advances in Pipelining	2-26



2.9.1(A)	Pipeline Stalls.....	2-26
2.9.1(B)	Data Forwarding: An Hardware Solution.....	2-26
2.9.1(C)	Dynamic Instruction Scheduling (or) Out-Of-Order (OOO) Execution	2-27
2.9.2	Handling of Branch Instructions to Resolve Control Hazards.....	2-27
2.9.2(A)	Pre-Fetch Target Instruction.....	2-27
2.9.2(B)	Branch Target Buffer (BTB).....	2-28
2.9.2(C)	Loop Buffer.....	2-28
2.9.2(D)	Branch Prediction.....	2-28
2.9.2(E)	Pipeline Stall (Delayed Branch).....	2-28
2.9.2(F)	Loop Unrolling Technique.....	2-28
2.9.2(G)	Software Scheduling or Software Pipelining	2-29
2.10	Stack Implementation in ARM.....	2-29
2.11	Endianness	2-29
2.12	Condition Codes.....	2-29
2.13	Processor Core Vs. CPU Core	2-30
2.14	ARM7TDMI.....	2-30
2.14.1	Features of ARM7TDMI.....	2-31
2.14.2	Block Diagram and ARM7TDMI Interface Signals.....	2-33

Unit - III

Chapter 3 : ARM7TDMI Assembly Language Instructions and Programming
3-1 to 3-44

3.1	Different Types of Instructions.....	3-1
3.2	ARM Instruction Set	3-1
3.3	Data Processing Instructions.....	3-2
3.4	Arithmetic and Logical Instructions	3-3
3.4.1	Comparison and Test Instructions.....	3-5
3.4.2	Logical Instructions.....	3-6
3.4.3	Shift Operation using RS Lower Byte	3-9
3.5	Rotate and Barrel Shifter.....	3-9
3.6	Program Control Flow/ Branch Instructions	3-11
3.7	Load and Store Instructions.....	3-13
3.7.1	Load and Store Word or Unsigned Byte Instruction.....	3-13
3.7.2	Load and Store Half Word and Load Signed Byte	3-14
3.7.3	Counting Leading Zeros Instruction	3-15
3.7.4	Semaphore / Swap Register Instructions.....	3-18
3.8	Software Interrupt Instructions.....	3-19
3.8.1	Co-processor Instructions.....	3-19
3.9	Program Status Register Instructions	3-22
3.10	Conditional Execution.....	3-24
3.11	Multiple Register Load and Store Instructions	3-25
3.12	Stack Instructions.....	3-26
3.13	Thumb Instruction Set.....	3-27
3.13.1	Introduction to Thumb Instruction Set.....	3-27



3.13.2	Switching between ARM State and Thumb State	3-27
3.13.3	Thumb Programmer's Model	3-28
3.13.3(A)	Branching Instructions of Thumb	3-29
3.13.3(B)	Thumb Software Interrupt Instruction	3-30
3.13.3(C)	Thumb Data Processing Instructions	3-30
3.13.4	Advantage of Thumb Instructions	3-31
3.13.5	Instruction Timing	3-31
3.14	Assembler Rules and Directives	3-32
3.15	Assembly Language Programs	3-33
3.16	Interrupt Latency	3-39
3.16.1	Interrupt Service Routine	3-39
3.16.2	Interrupt Priorities	3-39
3.16.3	Interrupt Nesting	3-40
3.16.4	Interrupt Latency	3-40
3.17	Interrupts/Exception Handling in ARM	3-41
3.17.1	Exception Process	3-41
3.17.1(A)	Reset Handling	3-42
3.17.1(B)	Undefined Instructions	3-42
3.17.1(C)	Software Interrupt	3-42
3.17.1(D)	Prefetch Abort Vector	3-42
3.17.1(E)	Data Abort	3-43
3.17.1(F)	Interrupt Request (IRQ)	3-43
3.17.1(G)	Fast Interrupt Request (FIQ)	3-43
3.17.1(H)	High Vectors	3-43
3.18	Exception Priorities	3-43
3.18.1	Interrupt Handlers and Interrupt Handling Schemes	3-43
3.18.2	Example of Interrupt Handler : NVIC (Nested Vector Interrupt Controller) in LPC17xx	3-44

Unit - IV

Chapter 4 : Embedded C Programming for ARM**4-1 to 4-72**

4.1	ARM Development Environment Embedded Software	4-1
4.2	Overview of C Compiler and Optimization	4-1
4.2.1	Disadvantages of Assembly Programming	4-2
4.2.2	C Programming	4-3
4.2.3	Using ARM GCC Tool Chain	4-3
4.3	Speed of Code or Optimization of Codes	4-4
4.4	Source Code Engineering Tools for Embedded C / C++	4-5
4.5	Integrated Development Environment (IDE)	4-5
4.6	Optimization of Memory Needs	4-6
4.7	Basic C Data Types	4-7
4.7.1	Character Set of C	4-8
4.7.2	Keywords	4-9
4.7.3	Identifiers	4-9



4.7.4	Scalar Data Types in C.....	4-10
4.7.5	Constants and Variables	4-11
4.7.6	Escape Sequences	4-11
4.7.7	Unary Operators	4-12
4.7.8	Binary Operators	4-15
4.7.9	Ternary Operator.....	4-17
4.7.10	Assignment Operators and Statements.....	4-18
4.7.11	Selection Operators.....	4-19
4.8	C Looping Structures.....	4-19
4.8.1	if-else Selective Statement.....	4-19
4.8.2	Looping Statements	4-21
4.9	C- Control Structures for Iteration.....	4-21
4.9.1	for Loop.....	4-21
4.10	Register Allocations.....	4-26
4.10.1	Storage Class.....	4-27
4.11	Function Calls	4-29
4.12	Pointers.....	4-32
4.12.1	Pointer Aliasing	4-34
4.13	Structure Arrangement.....	4-34
4.14	Bitfields	4-35
4.15	Unaligned Data and Endianness	4-36
4.15.1	Unaligned Data.....	4-36
4.15.2	Endianness	4-36
4.15.3	Division.....	4-36
4.16	Floating Point	4-37
4.17	Inline Functions and Inline Assembly	4-37
4.17.1	Inline Assembly.....	4-37
4.18	Portability Issues	4-39
4.19	LPC2148#	4-39
4.20	Features of LPC 2148#	4-40
4.21	Architecture of LPC2148#(Block Diagram and its Description).....	4-41
4.22	System Control Block (PLL and VPB Divider).....	4-49
4.22.1	PLL.....	4-49
4.22.2	VPB Divider.....	4-49
4.23	Memory Map	4-49
4.24	Fast General Purpose Parallel I/O (GPIO).....	4-50
4.24.1	Features of GPIO	4-50
4.25	Pin Connect Block.....	4-51
4.26	Timers	4-51
4.26.1	Features of Timers	4-51
4.27	Interfacing GPIO using Timer.....	4-52
4.27.1	Interfacing Light Emitting Diode (LED) and Seven Segment Display (SSD).....	4-52
4.28	Interfacing DC Motor Using PWM.....	4-57
4.29	Interfacing GSM Module Using UART	4-59



4.29.1	GSM Modem and its Interfacing with Microcontroller	4-59
4.29.2	Interfacing GSM with LPC2148.....	4-59
4.30	Interfacing EEPROM using I2C Communication Protocol.....	4-62
4.31	Interfacing Memory Using SPI Protocol	4-63
4.31.1	Interfacing SD-MMC with LPC2148 using SPI	4-64
4.32	Interfacing Internal ADC	4-64
4.32.1	Interfacing DAC.....	4-66
4.32.2	Interfacing Sensors to LPC 2148.....	4-70

Unit - V

Chapter 5 : Cache and Memory Management and Protection
5-1 to 5-32

5.1	Memory Technologies	5-1
5.2	Memory Hierarchy : Need and Organization	5-3
5.3	Cache Memory : Concept and Cache Consistency.....	5-4
5.3.1	Cache Operation	5-4
5.3.2	Principles of Locality of Reference	5-5
5.3.3	Cache Performance	5-6
5.3.4	Cache Design.....	5-7
5.3.4(A)	Look-Through Cache Designs	5-7
5.3.4(B)	Look-Aside Cache Designs	5-8
5.3.5	Cache Consistency (Also Known as Cache Coherency).....	5-8
5.3.6	Write Policy	5-9
5.3.7	Replacement Algorithms	5-11
5.3.8	Multiple Levels of Cache: Cost and Performance Measurement.....	5-15
5.3.9	Cache Coherency with L1 and L2 Cache.....	5-15
5.4	Cache Mapping Functions	5-16
5.4.1	Direct Mapping Technique.....	5-16
5.4.2	Fully Associative Mapping	5-18
5.4.3	Set Associative Mapping	5-19
5.5	Unified and Split Cache	5-20
5.6	ARM Memory Hierarchy	5-21
5.7	ARM Cache Features	5-21
5.7.1	Coprocessor 15 for Cache System Control and Various Processes	5-21
5.8	ARM Memory Map.....	5-22
5.9	Protected Systems	5-22
5.9.1	Memory Protection Unit (MPU) and ARM Processor with MPU	5-22
5.10	Memory Management Unit (MMU) Advantage	5-24
5.11	Virtual Memory	5-24
5.12	Virtual Memory Translation.....	5-25
5.12.1	Paging Mechanism or Memory Management Unit	5-27
5.12.2	Segmentation	5-28
5.12.3	Physical Memory vs. Virtual Memory.....	5-28
5.13	Multitasking with MMU.....	5-29



5.14	MMU Organization in Virtual Memory System	5-29
5.14.1	Page Tables	5-30
5.14.2	Translation Look Aside Buffer.....	5-31
5.14.3	ARM MMU.....	5-31
5.14.4	Fast Context Switch Extension	5-31
5.15	Tightly Coupled Memory (TCM).....	5-32

Unit - VI

Chapter 6 : ARM Peripherals and Versions**6-1 to 6-19**

6.1	Advanced Microprocessor Bus Architecture (AMBA) Bus System Overview	6-1
6.1.1	Advanced High-Performance Bus (AHB)	6-2
6.1.2	Advanced System Bus (ASB).....	6-2
6.1.3	Comparison of AHB and APB Buses of AMBA Architecture	6-2
6.1.4	Advanced Peripheral Bus (APB).....	6-3
6.1.5	A Typical AMBA-based Microcontroller.....	6-3
6.1.6	AHB Bus Transfer and APB Bridge	6-4
6.1.7	APB Bus Transfer.....	6-4
6.2	Overview of ARM Versions: ARM v5, ARM v6, ARM v7, ARM v8	6-5
6.3	ARM v7 : Cortex-A, Cortex-R and Cortex-M	6-10
6.3.1	Cortex-A Series	6-10
6.3.2	Cortex-R Series.....	6-11
6.3.3	Cortex-M Series.....	6-11
6.4	Cortex-A Processor Series Survey : Features, Versions and Applications.....	6-12
6.4.1	Cortex-A57	6-13
6.4.2	Cortex-A53	6-13
6.4.3	Cortex-A17	6-14
6.4.4	Cortex-A15	6-14
6.4.5	Cortex-A9	6-14
6.4.6	Cortex-A7	6-14
6.4.7	Cortex-A5	6-15
6.5	Cortex-R Processor Series Survey : Features, Versions and Applications.....	6-15
6.5.1	Comparison of Various Cortex-R Processors	6-15
6.5.2	Cortex-R4 Processor	6-16
6.5.3	Cortex-R5 Processor	6-16
6.5.4	Cortex-R7 Processor	6-17
6.5.5	Applications of ARM Cortex-R Processors	6-17
6.6	Cortex-M Processor Series Survey : Features, Versions and Applications	6-18
6.7	ARM Optimization Techniques	6-18
	• Appendix : Solved University Question Paper of Winter 2020	A-1 to A-2

