USN | | | | | | | | | | |

Internal Assessment Test 1 – July 2022

Sub:	Microcontroller and Embedded Systems					Sub Code:	18CS44	Branch:	ISE
Date:	07/07/2022	Duration:	90 min's	Max Marks:	50	Sem/Sec:	VI / A, B & C		OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	What is pipeline in ARM? Explain the different stages of ARM processor and ARM 9 processor.	[10]	CO1	L2
2	Explain ARM Core data flow model. Mention the different registers of ARM processor.	[10]	CO1	L2
3	Differentiate between i) RISC and CISC Processors ii) Microcontroller and microprocessor.	[10]	CO1	L2
4	With example, explain the following ARM instructions i) MUL ii) MOV iii) CMP iv) AND v) ROR with an example	[10]	CO1	L2
5 (a)	Write an ALP to find given number is prime or not.	[06]	CO2	L2
(b)	Write the applications of ARM processor.	[04]	CO1	L2
6(a)	Discuss the load store instruction with respect to i) Single register transfer ii) Multiple register transfer.	[06]	CO1	L1
6(b)	Write an ALP to add two 16-bit numbers and store the result in R1.	[04]	CO2	L3

Faculty Signature CCI Signature HOD Signature

Internal Assessment Test 1 – July 2022
Scheme of Evaluation

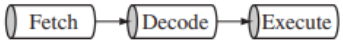
Sub:	Microcontroller and Embedded Systems					Sub Code:	18CS44	Branch:	ISE
Date:	07/07/2022	Duration:	90 min's	Max Marks:	50	Sem/Sec:	VI / A, B & C		OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	What is pipeline in ARM? Explain the different stages of ARM processor and ARM 9 processor	[10]	CO1	L2

Scheme: - Definition + Explanation of different stages of ARM processor = 3+7 M = 10M

Solution:-

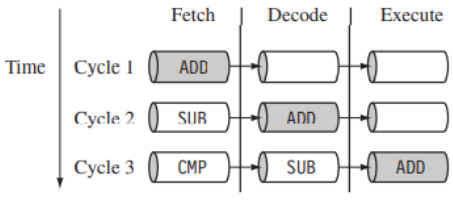
A pipeline is the mechanism a RISC processor uses to execute instructions. Using a pipeline speeds up execution by fetching the next instruction while other instructions are being decoded and executed.



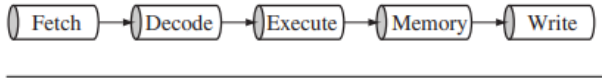
Explanation:

A three-stage pipeline includes: ■ Fetch loads an instruction from memory. ■ Decode identifies the instruction to be executed. ■ Execute processes the instruction and writes the result back to a register.

The pipeline using a simple example is illustrated here. It shows a sequence of three instructions being fetched, decoded, and executed by the processor. Each instruction takes a single cycle to complete after the pipeline is filled. The three instructions are placed into the pipeline sequentially. In the first cycle the core fetches the ADD instruction from memory. In the second cycle the core fetches the SUB instruction and decodes the ADD instruction. In the third cycle, both the SUB and ADD instructions are moved along the pipeline. The ADD instruction is executed, the SUB instruction is decoded, and the CMP instruction is fetched. This procedure is called filling the pipeline. The pipeline allows the core to execute an instruction every cycle. As the pipeline length increases, the amount of work done at each stage is reduced, which allows the processor to attain a higher operating frequency. This in turn increases the performance. The system latency also increases because it takes more cycles to fill the pipeline before the core can execute an instruction. The increased pipeline length also means there can be data dependency between certain stages.

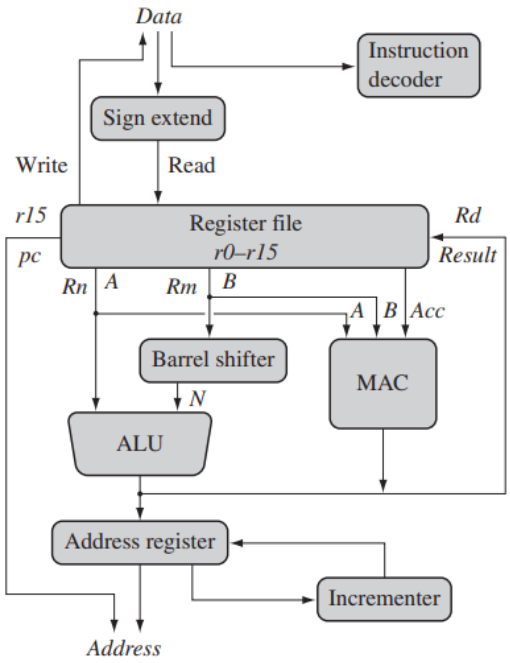


Pipelined instruction sequence.

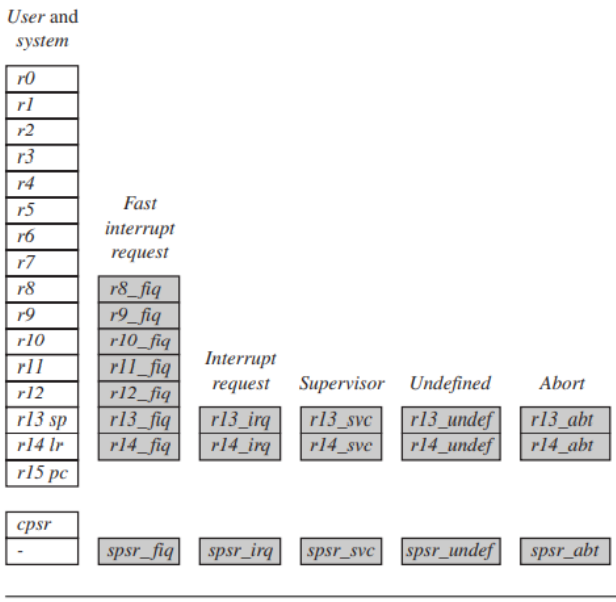


ARM9 five-stage pipeline

2	**Explain ARM Core data flow model. Mention the different registers of ARM processor.** **Scheme: -** Explanation of ARM core dataflow model + Diagram (2) = 6+4 =10 Marks **Solution: -**	[10]	CO1	L2



ARM core dataflow model

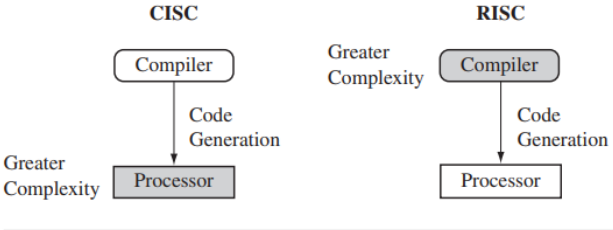


Complete ARM register set

There are up to 18 active registers: 16 data registers and 2 processor status registers. The data registers are visible to the programmer as r0 to r15.

3	**Differentiate between i) RISC and CISC Processors ii) Microcontroller and microprocessor.** **Scheme: -** (i) Difference between RISC & CISC Processor – 5 Marks (ii) Difference between Microcontroller & Microprocessor - 5 Marks **Solution: - (i) Any 5 differences**	[10]	CO1	L2

RISC	CISC
It is a Reduced Instruction Set Computer.	It is a Complex Instruction Set Computer.
It emphasizes on software to optimize the instruction set.	It emphasizes on hardware to optimize the instruction set.
It is a hard wired unit of programming in the RISC Processor.	Microprogramming unit in CISC Processor.
It requires multiple register sets to store the instruction.	It requires a single register set to store the instruction.
RISC has simple decoding of instruction.	CISC has complex decoding of instruction.
Uses of the pipeline are simple in RISC.	Uses of the pipeline are difficult in CISC.
It uses a limited number of instruction that requires less time to execute the instructions.	It uses a large number of instruction that requires more time to execute the instructions.
It uses LOAD and STORE that are independent instructions in the register-to-register a program's interaction.	It uses LOAD and STORE instruction in the memory-to-memory interaction of a program.
RISC has more transistors on memory registers.	CISC has transistors to store complex instructions.
The execution time of RISC is very short.	The execution time of CISC is longer.
RISC architecture can be used with high-end applications like telecommunication, image processing, video processing, etc.	CISC architecture can be used with low-end applications like home automation, security system, etc.
It has fixed format instruction.	It has variable format instruction.
The program written for RISC architecture needs to take more space in memory.	Program written for CISC architecture tends to take less space in memory.
Example of RISC: ARM, PA-RISC, Power Architecture, Alpha, AVR, ARC and the SPARC.	Examples of CISC: VAX, Motorola 68000 family, System/360, AMD and the Intel x86 CPUs.



CISC vs. RISC. CISC emphasizes hardware complexity. RISC emphasizes compiler complexity

Solution: - (ii) Any 5 differences

Sl. No	Microprocessor	Microcontroller
1.	We need to connect peripherals externally. So it makes circuit bulky.	The presence of peripherals such as RAM, ROM, Input-output, and Timers are In-built. So It is available on a single chip.
2.	It increases the overall cost of the system high.	The overall cost of the system is less.
3.	We can connect external memory in ranges of Mbytes and even Gbytes. But speed is less.	The inbuilt finite memory helps to improve the speed of operations.
4.	You can't use it in a compact system.	You can use it in compact systems.
5.	Due to external components, the total power consumption is high. Therefore, it is not ideal for the devices running on stored power like batteries.	As external components are low, total power consumption is less. So it can be used with devices running on stored power like batteries.
6.	Most of the microprocessors do not have power-saving features.	Most of the microcontrollers offer power-saving mode.
7.	The microprocessor has a smaller number of registers, so more operations are memory-based.	The microcontroller has more register. Hence the programs are easier to write.
8.	These are based on the von Neumann model where program and data are stored in the same memory module.	These are based on Harvard architecture where program memory and data memory are separate.
9.	It is a central processing unit on a single silicon-based integrated chip.	It is a byproduct of the development of microprocessors with a CPU along with other peripherals.
10	It uses an external bus to interface to RAM, ROM, and other peripherals.	It uses an internal controlling bus.
11	Microprocessor-based systems can run at a very high speed because of the technology involved.	Microcontroller based systems run up to 200MHz or more depending on the architecture.
12	It's useful for general purpose applications that allow you to handle loads of data.	It's useful for application-specific systems.
13	It's complex and expensive, with a large number of instructions to process.	It's simple and inexpensive with less number of instructions to process.

4 **With example, explain the following ARM instructions i) MUL ii) MOV iii) CMP iv) AND v) ROR with an example**

[10] CO1 L2

Scheme: - Each instruction 2 Marks with example. (2*5 = 10 Marks)

Solution: -

(i)MUL

simple multiply instruction that multiplies registersr1 and r2 together and places the result into register r0. In this example, register r1 is equal to the value 2, and r2 is equal to 2. The result, 4, is then placed into register r0

```
PRE    r0 = 0x00000000
       r1 = 0x00000002
       r2 = 0x00000002

       MUL   r0, r1, r2   ; r0 = r1*r2

POST   r0 = 0x00000004
       r1 = 0x00000002
       r2 = 0x00000002
```

(ii)MOV

The MOV instruction takes the contents of register r5 and copies them into register r7, in this case, taking the value 5, and overwriting the value 8 in register r7.

```
PRE     r5 = 5
        r7 = 8
        MOV     r7, r5     ; let r7 = r5
POST    r5 = 5
        r7 = 5
```

(iii)CMP

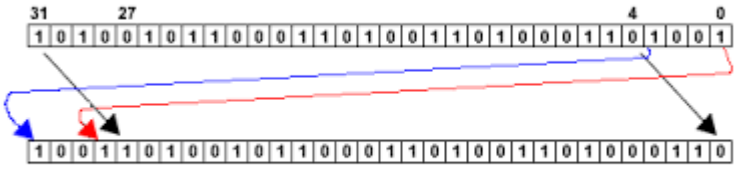
The value of the z flag prior to execution is 0 and is represented by a lowercase z. After execution the z flag changes to 1 or an uppercase Z. This change indicates equality.

```
PRE     cpsr = nzcvqiFt_USER
        r0 = 4
        r9 = 4

        CMP    r0, r9

POST    cpsr = nZcvqiFt_USER
```

(iv)AND

```
AND          R0, R0, R1
```

(v)ROR



ROR #4

5 (a)	**Write an ALP to find given number is prime or not**
	Scheme:- Program = 6 Marks
	Solution:-
	AREA Prime_or_Not, code, readonly
	ENTRY
	MOV R0, #15; Number which you want to test
	CMP R0, #01; Comparing with 01
	BEQ PRIME; If equal declare directly as prime
	CMP R0, #02; Compare with 02
	BEQ PRIME; If equal declare directly as prime
	MOV R1,R0;Copy test number in R1
	MOV R2,#02;Initial divider
	UP
	BL DIVISION;Call for division sub-function
	CMP R8,#00 ;Compare remainder with 0
	BEQ NOTPRIME ;If equal then its not prime
		[06]	CO2	L2

```
ADD R2,R2,#01          ;If not increment divider and check
CMP R2,R1              ;Compare divider with test number
BEQ PRIME              ;All possible numbers are done means It's prime
B UP                   ;If not repeat until end
NOTPRIME
LDR R3,=0x11111111     ;Declaring test number is not prime
B STOP                 ;Jumping to infinite looping
PRIME
LDR R3,=0xFFFFFFFF     ;Declaring test number is prime number
STOP B STOP            ;Infinite looping
DIVISION               ;Function for division operation
MOV R8,R0              ;Copy of data from main function
MOV R9,R2              ;Copy of divider from main function
LOOP
SUB R8,R8,R9           ;Successive subtraction for division
ADD R10,R10,#01        ;Counter for holding the result of division
CMP R8,R9              ;Compares for non-zero result
BPL LOOP               ;Repeats the loop if subtraction is still needed
MOV PC,LR              ;Return back to main function
END
```

(b) **Write the applications of ARM processor.** [04] CO1 L2

Scheme:- Write the applications of ARM processor : 5 Marks

Solution:-

The operating system schedules applications—code dedicated to handling a particular task. An application implements a processing task; the operating system controls the environment. An embedded system can have one active application or several applications running simultaneously. ARM processors are found in numerous market segments, including networking, automotive, mobile and consumer devices, mass storage, and imaging.

Within each segment ARM processors can be found in multiple applications. For example, the ARM processor is found in networking applications like home gateways, DSL modems for high-speed Internet communication, and 802.11 wireless communication. The mobile device segment is the largest application area for ARM processors because of mobile phones.

ARM processors are also found in mass storage devices such as hard drives and imaging products such as inkjet printers—applications that are cost sensitive and high volume. In contrast, ARM processors are not found in applications that require leading-edge high performance. Because these applications tend to be low volume and high cost, ARM has decided not to focus designs on these types of applications.

.

6(a)	**Discuss the load store instruction with respect to i) Single register transfer ii) Multiple register transfer.** **Scheme: -** Single register transfer (3 Marks) + Multiple register transfer (3 Marks) = 6 Marks. **Solution: -** The ARM has three sets of instructions which interact with main memory. These are: • Single register data transfer (LDR/STR) • Block data transfer (LDM/STM) • Single Data Swap (SWP) The basic load and store instructions are: Load and Store Word or Byte or Halfword LDR / STR / LDRB / STRB / LDRH / STRH - Explanation	[06]	CO1	L1

6(b)	**Write an ALP to add two 16-bit numbers and store the result in R1.**
	
	Scheme: - Program logic + code = 4
	
	AREA ADDITION, CODE, READONLY
	
	ENTRY
	
	MOV R5,#6
	
	MOV R0,#0
	
	LDR R1,=VALUE1
	
	LOOP LDRH R2,[R1],#2
	
	ADD R0,R0,R2
	
	SUBS R5,R5,#1
	
	BNE LOOP
	
	LDR R4,=RESULT
	
	STR R0,[R4]
	
	STOP B STOP
	
	VALUE1 DCW 0X1111,0X2222,0X3333,0X4444,0X3333,0X5555
	
	AREA DATA2,DATA,READWRITE
	
	RESULT DCD 0X0
	
	END
		[04]	CO2	L3

Faculty Signature CCI Signature HOD Signature