KSIT, BENGALURU

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

COURSE FILE

NAME OF THE STAFF	:	**Dr. B SUDASHAN**
SUBJECT CODE/NAME	:	**18EC35- COMPUTER ORGANIZATIO AND ARCHITECTURE**
SEMESTER/YEAR	:	**III/II**
ACADEMIC YEAR	:	**2020 – 21**
BRANCH	:	**ECE**

COURSE IN-CHARGE **MODULE COORDINATOR** **HOD**

K. S. INSTITUTE OF TECHNOLOGY

VISION

"To impart quality technical education with ethical values, employable skills and research to achieve excellence".

MISSION

- To attract and retain highly qualified, experienced & committed faculty.
- To create relevant infrastructure.
- Network with industry & premier institutions to encourage emergence of new ideas by providing research & development facilities to strive for academic excellence.
- To inculcate the professional & ethical values among young students with employable skills & knowledge acquired to transform the society.

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

VISION

"To achieve excellence in academics and research in Electronics & Communication Engineering to meet societal need".

MISSION

- To impart quality technical education with the relevant technologies to produce industry ready engineers with ethical values.
- To enrich experiential learning through active involvement in professional clubs & societies.
- To promote industry-institute collaborations for research & development.

K.S. INSTITUTE OF TECHNOLOGY

Department of Electronics and Communication Engg.

PROGRAM EDUCATIONAL OBJECTIVES (PEO'S)

- Excel in professional career by acquiring domain knowledge.
- Motivation to pursue higher Education & research by adopting technological innovations by continuous learning through professional bodies and clubs.
- To inculcate effective communication skills, team work, ethics and leadership qualities.

PROGRAM SPECIFIC OUTCOMES (PSO'S)

PSO1: Graduate should be able to understand the fundamentals in the field of Electronics & Communication and apply the same to various areas like Signal processing, embedded systems, Communication & Semiconductor technology.

PSO2: Graduate will demonstrate the ability to design, develop solutions for Problems in Electronics & Communication Engineering using hardware and software tools with social concerns.

K S INSTITUTE OF TECHNOLOGY

PROGRAM OUTCOMES (PO'S)

Engineering Graduates will be able to:

PO1 :Engineering knowledge: Apply the knowledge of mathematics, science, engineeringfundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2 : Problem analysis: Identify, formulate, review research literature, and analyze complexengineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 : Design/development of solutions: Design solutions for complex engineering problems anddesign system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4 : Conduct investigations of complex problems: Use research-based knowledge and researchmethods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 : Modern tool usage: Create, select, and apply appropriate techniques, resources, and modernengineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 : The engineer and society: Apply reasoning informed by the contextual knowledge to assesssocietal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 : Environment and sustainability: Understand the impact of the professional engineering solutionsin societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 : Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms ofthe engineering practice.

PO9 :Individual and team work: Function effectively as an individual, and as a member or leader indiverse teams, and in multidisciplinary settings.

PO10 :Communication: Communicate effectively on complex engineering activities with the engineeringcommunity and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 ;Project management and finance: Demonstrate knowledge and understanding of theengineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage inindependent and life-long learning in the broadest context of technological change.

K. S. INSTITUTE OF TECHNOLOGY

#14, Raghuvanahalli, Kanakapura Main Road, Bengaluru-5600109

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Course: COMPUTER ORGANIZATION AND ARCHITECTURE			
Name of the staff : Dr. B Sudarshan			
Type: Core		Course Code:18EC35	
No of Hours per week			
Theory (Lecture Class)	Practical/Field Work/Allied Activities	Total/Week	Total teaching hours
4	0	4	50
Marks			
Internal Assessment	Examination	Total	Credits
40	60	100	3

Aim/Objective of the Course:

1. Explain the basic sub systems of a computer, their organization, structure and operation.
2. Illustrate the concept of programs as sequences of machine instructions.
3. Demonstrate different ways of communicating with I/O devices
4. Describe memory hierarchy and concept of virtual memory.
5. Illustrate organization of simple pipelined processor and other computing systems.

Course Learning Outcomes:

After completing the course, the students will be able to,

CO1	**Categorize** the operations of major subsystems of computer	Analyzing (K4)
CO2	**Analyze** different types of semiconductor memories and secondary memories.	Analyzing (K4)
CO3	**Analyze** ALU and control unit operations.	Analyzing (K4)
CO4	**Analyze** the working of stacks, queues, subroutines and handling different types of interrupts.	Analyzing (K4)
CO5	**Apply** the concepts of hardwired control and micro programmed control.	Applying (K3)

Syllabus Content:

Syllabus Content:	
Module 1 **Basic Structure of Computers:** Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Software, Performance – Processor Clock, Basic Performance Equation **Machine Instructions and Programs:** Numbers, Arithmetic Operations and Characters, IEEE standard for Floating point Numbers, Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing LO: At the end of this session the student will be able to, 1. Explain the basic structure of a computer with its functions. 2. Explain machine instructions. 3. Explain different types of operations and instructions.	CO1, CO3 10 hrs
Module 2: Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions LO: At the end of this session the student will be able to, 1. Explain the Addressing Modes. 2. Explain Assembly language and its usage. 3. Explain the usage of additional instructions using stacks and subroutines.	CO3,CO4 10 hrs.
Module 3: **Input/Output Organization:** Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Direct Memory Access LO: At the end of this session the student will be able to, 1. Explain different ways of accessing interrupts. 2. Explain different ways of accessing I/O devices. 3. Explain Direct Memory Access.	CO4 10 hrs

Module 4: **Memory System:** Basic Concepts, Semiconductor RAM Memories-Internal organization of memory chips, Static memories, Asynchronous DRAMS, Read Only Memories, Cash Memories, Virtual Memories, Secondary Storage-Magnetic Hard Disks LO: At the end of this session the student will be able to, 1. Explain the basic concepts of memory system. 2. Explain the different types of semiconductor memories. 3. Explain different types of secondary storage memories.	CO1,CO2 10 hrs

Module 5: **Basic Processing Unit:** Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hardwired Control, Microprogrammed Control LO: At the end of this session the student will be able to, 1. Explain the fundamental concepts of basic processing unit. 2. Explain the simple processor organization based on hardwired control. 3. Explain simple processor organization based on micro programmed control.	CO3,CO5 10 hrs

Text Books:

Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002.

Reference Books:

1. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier, 2009.
2. William Stallings: Computer Organization & Architecture, 7th Edition, PHI, 2006.
3. Vincent P. Heuring & Harry F. Jordan: Computer Systems Design and Architecture, 2nd Edition, Pearson Education, 2004.

Assessment:
Type of test/examination: Written examination
Continuous Internal Evaluation(CIE) : 40 marks (Average of all three tests will be considered)

Semester End Exam(SEE) : 60 marks (students have to answer all main questions)
Test duration: 1 :30 hr
Examination duration: 3 hrs

CO - PO Mapping

PO1: Science and engineering Knowledge **PO2:** Problem Analysis **PO3:** Design & Development **PO4:**Investigations of Complex Problems **PO5:** Modern Tool Usage **PO6:** Engineer & Society	**PO7:**Environment and Sustainability **PO8:**Ethics **PO9:**Individual & Team Work **PO10:** Communication **PO11:**Project Mngmt & Finance **PO12:**Life long Learning

PSO1: Graduate should be able to understand the fundamentals in the field of Electronics Communication and apply the same to various areas like Signal processing, Embedded system Communication & Semiconductor technology.

PSO2: Graduate will demonstrate the ability to design, develop solutions for problems in Electronics & Communication Engineering using hardware and software tools with social concern

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	-	-	-	-	-	-	-	-	1	-	-
CO2	3	2	1	-	-	-	-	-	-	-	-	1	1	1
CO3	3	2	1	-	-	-	-	-	-	-	-	1	-	-
CO4	3	2	1	-	-	-	-	-	-	-	-	1	-	-
CO5	3	2	1	-	-	-	-	-	-	-	-	1	-	-
18EC 35	3	2	1	-	-	-	-	-	-	-	-	1	1	1

JUSTIFICATION FOR CO-PO MAPPING

CO1	**PO1 (3) - Students will be able to categorize subsystems of a computer.** **PO2 (2) – Students will be able to write simple assembly language instructions .** **PO3 (1) - Students will be able to design bigger systems using some basic knowledge.** **PO12 (1) – Students will be using computer systems entire life**
CO2	**PO1 (3) – Students will be able to identify and learn the usage of different types of memory.** **PO2 (2) – Students will be able solve problems related to I/O mapping.** **PO3 (1) - Students will be able to design bigger systems using some basic knowledge.** **PO12 (1) – Students will be using I/O and memory entire life** **PSO1, PSO2 (1) - Students will be learning about semiconductor memories and so their knowledge is necessary while using hardware and software tools.**
CO3	**PO1 (3) – Students will be able to analyze arithmetic and logical operations and timing signals.** **PO2 (2) – Students will be able to solve problems related to timing signals.** **PO3 (1) - Students will be able to design bigger systems using some basic knowledge.** **PO12 (1) – Students will be using computer functionalities entire life**
CO4	**PO1 (3) – Student will be able to understand and learn on accessing I/O devices.** **PO2 (2) – Students will be able to write simple assembly language instructions using subroutines.**

	PO3 (1) - Students will be able to design bigger systems using some basic knowledge. **PO12 (2) – Students will be able to use the concept of interrupts**
CO5	**PO1 (3) – Students will be able to compare between hardwired and microprogrammed control unit.** **PO2 (2) – Students will be able to decide when to use which type of control unit.** **PO3 (1) - Students will be able to design bigger systems using some basic knowledge.** **PO12 (1) – Students will be using computer systems and their control units entire life**

Signature of staff **Signature of module coordinator** **Signature of HOD**

K.S INSTITUT E OF TECHNOLOGY, Bengaluru-109

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

TENTATIVE CALENDAR OF EVENTS: ODD SEMESTER (2020-2021)

SESSION: SEP 2020 – JAN 2021

Week No.	Month	Day						Days	Activities	Department Activities Tentative Dates
		Mon	Tue	Wed	Thu	Fri	Sat			
1	SEP		1*	2	3	4	5	5	1*-Commencement of Higher Semester	
2	SEP	7	8	9	10	11	12	6		10 - Technical Talk IEEE, ASH
3	SEP	14	15	16	[redacted]	18	19	5	17- Mahalaya Amavasya	15- Engineers Day IEEE 19 - Guest Lecture
4	SEP	21	22	23	24	25	26 TA	6		21 - Technical Talk 23 - Valedictory IEEE
5	SEP / OCT	28 T1	29 T1	30 T1	1	[redacted]	3	5	2- Mahatma Gandhi Jayanthi	
6	OCT	5	6 BV	7 ASD	8	9	10	6	5-10 First Feed Back	5 - IEEE Day 10 - Guest Lecture
7	OCT	12	13	14	15	16	[redacted]	5		15 - Tachnical Talk
8	OCT	19	20	21	22	23	24	6	24 Monday Time table	19 - Tachnical Talk 20 - Technical Talk
9	OCT	[redacted]	27	28	29	[redacted]	[redacted]	3	26- Vijayadashami 30- Eid-Milad 31- Maharishi Valmiki	
10	NOV	2	3	4	5	6	7TA	6	7-Wednesday Time Table	
11	NOV	9	10	11	12	13	[redacted]	5		12 - Technical Talk
12	NOV	[redacted]	17 T2	18 T2	19 T2	20	21	5	16 - Balipadyami Deepavalli 18-21 Second Feed Back 21- Friday Time Table	20 - Simulation
13	NOV	23	24 BV	25 ASD	26	27	[redacted]	5		
14	NOV /DEC	30	1	2	[redacted]	4	5	5	3- Kanakadasa Jayanti 5- Monday Time Table	
15	DEC	7	8	9	10	11	[redacted]	5		
16	DEC	14	15	16	17	18	19	6	19- Monday Time Table	
17	DEC	21	22	23	24	[redacted]	[redacted]	4	25- Chrismas	
18	DEC/JAN	28	29	30	31	1TA	2	6	2- Tuesday Time Table	
19	JAN	4LT	5LT	6LT	7LT	8	[redacted]	5		
20	JAN	11T3	12T3	13T3	[redacted]	15	16	5	14- Makara Sankranthi 16* -Last Working Day	
Total No of Working Days : 106										

Total Number of working days (Excluding holidays and Tests)=87

H	Holiday
BV	Blue Book Verification
T1, T2, T3	Tests 1,2, 3
ASD	Attendance & Sessional Display
DH	Declared Holiday
LT	Lab Test
TA	Test attendance

Monday	16
Tuesday	16
Wednesday	16
Thursday	16
Friday	17
Saturday	6
Total	**87**

30/11/2020

K. S INSTITUTE OF TECHNOLOGY, BENGALURU-560109

TENTATIVE CALENDAR OF EVENTS: ODD SEMESTER (2020-2021)

SESSION: SEP 2020 - JAN 2021

Week No.	Month	Day						Days	Activities
		Mon	Tue	Wed	Thu	Fri	Sat		
1	SEP		1*	2	3	4	5	5	1*-Commencement of Higher Semester
2	SEP	7	8	9	10	11	12	6	
3	SEP	14	15	16	17 H	18	19	5	17- Mahalaya Amavasya
4	SEP	21	22	23	24	25	26TA	6	
5	SEP / OCT	28 T1	29 T1	30 T1	1	2 H	3	5	2- Mahatma Gandhi Jayanthi
6	OCT	5	6BV	7ASD	8	9	10	6	5-10 First Feed Back
7	OCT	12	13	14	15	16	17 DH	5	
8	OCT	19	20	21	22	23	24	6	24 - Monday Time Table
9	OCT	26 H	27	28	29	30 H	31 H	3	26- Vijayadashami 30- Eid-Milad 31- Maharishi Valmiki Jayanti
10	NOV	2	3	4	5	6	7 TA	6	7 - Wednesday Time Table
11	NOV	9	10	11	12	13	14 DH	5	
12	NOV	16 H	17 T2	18 T2	19 T2	20	21	5	16 - Balipadyami Deepavalli 18 - 21 Second Feed Back 21 - Friday Time Table
13	NOV	23	24BV	25ASD	26	27	28 DH	5	
14	NOV /DEC	30	1	2	3 H	4	5	5	3- Kanakadasa Jayanti 5 - Monday Time Table
15	DEC	7	8	9	10	11	12 DH	5	
16	DEC	14	15	16	17	18	19	6	19- Monday Time Table
17	DEC	21	22	23	24	25 H	26 DH	4	25-Christmas
18	DEC/ JAN	28	29	30	31	1 TA	2	6	2Thursday Time Table
19	JAN	4 LT	5 LT	6 LT	7 LT	8	9 DH	5	
20	JAN	11 T3	12 T3	13 T3	14 H	15	16 *	5	14- Makara sankaranthi 16* -Last Working Day
									Total No of Working Days : 106

Total Number of working days (Excluding holidays and Tests)=87

H	Holiday
BV	Blue Book Verification
T1,T2, T3	Tests 1,2, 3
ASD	Attendance & Sessional Display
DH	Declared Holiday
LT	Lab Test
TA	Test attendance

Monday	16
Tuesday	16
Wednesday	16
Thursday	16
Friday	17
Saturday	6
Total	87

PRINCIPAL
K.S. INSTITUTE OF TECHNOLOGY
BENGALURU - 560 109.

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE -109

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

INDIVIDUAL TIME TABLE FOR THE YEAR - 2020 (ODD SEMESTER)

W.E.F. : 1/9/2020

NAME OF THE FACULTY : Dr. B. SUDHARSHAN

DESIGNATION: PROFESSOR

PERIOD	1	2	11.00 AM 11.15 AM	3	4	1.15 PM 1.45 PM	5	6
TIME	9.00 AM 10.00 AM	10.00 AM 11.00 AM		11.15 AM 12.15 PM	12.15 PM 1.15 PM		1.45 PM 2.45 PM	2.45 PM 3.45 PM
DAY								
MON			TEA BREAK			LUNCH BREAK		PEDAGOGY CO & A - A (18EC35) -
TUE	CO & A - B (18EC35)							CO & A - B (18EC35)
WED		CO & A - A (18EC35)					DSD LAB (18ECL38)	
THU	CO & A - B (18EC35)				CO & A - A (18EC35)			
FRI		CO & A - B (18EC35)			CO & A - A (18EC35)			
SAT	PROJECT WORK PHASE - I (17ECP78)							

	Subject Code	Subject Name	Sem	Section	Work Load
Subject 1	18EC35	Computer Organization & Architecture	III	A&B	8
Lab -1	18ECL38	Digital System Design Laboratory	III	A&B	9
Project	17ECP78	Project Work Phase–I + Project work Seminar	VII		2
ADDITIONAL WORK: MENTORING AND OTHERS					
TOTAL LOAD = 19 Hrs/Week					

Time Table Co-ordinator

HOD

26/8

Principal

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE -109

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

III SEMESTER TIME TABLE FOR THE YEAR 2020 (ODD SEMESTER)

W.E.F. : 1/9/2020

ONLINE TIME TABLE

SEC : 'B'

CLASS TEACHER : Mrs. PRAGATI . P

PERIOD	1	2		3	4		5	6
TIME / DAY	9.00 AM 10.00 AM	10.00 AM 11.00 AM	11.00 AM 11.30 AM	11.30 AM 12.30 PM	12.30 PM 1.30 PM	1.30 PM 2.00 PM	2.00 PM 3.00 PM	3.00 PM 4.00 PM
MON	DSD (18EC34)	PE&I (18EC36)		MATHS (18MAT31)	NT (18EC32)		PEDAGOGY (DSD)	Vyavaharika Kannada (18KVK39/49) Aadalitha Kannada (18KAK39/49)
TUE	CO&A (18EC35)	ED (18EC33)		MATHS (18MAT31)	NT (18EC32)		← PLACEMENT →	
WED	ED (18EC33)	NT (18EC32)		MATHS (18MAT31)	DSD (18EC34)		← DSD LAB(18ECL38) →	
THU	CO&A (18EC35)	PE&I (18EC36)		MATHS (18MAT31)	ED (18EC33)		← ED&I LAB(18ECL37) →	
FRI	NT (18EC32)	CO&A (18EC35)		DSD (18EC34)	PE&I (18EC36)		PEDAGOGY (PE&I/ CO&A)	DIP MATHS 18MATDIP31
SAT	PE&I (18EC36)	DSD (18EC34)		ED (18EC33)	CO&A (18EC35)		PEDAGOGY (NT)	PEDAGOGY (ED)

BREAK — LUNCH BREAK

Sub-Code	Subject Name	Faculty Name
18MATDIP31	Additional Mathematics - I	
18MAT31	Transform Calculus, Fourier Series and Numerical Techniques	Mr. Venkataramana
18EC32	Network Theory	Mrs. Pragati. P
18EC33	Electronic Devices	Mrs. Sahana Salagare
18EC34	Digital System Design	Mr. B.R.Santhosh Kumar
18EC35	Computer Organization & Architecture	Dr. B.Sudharshan
18EC36	Power Electronics & Instrumentation	Mr. Christo Jain
18ECL37	Electronic Devices & Instrumentation Laboratory	Mrs. Pragati. P, Mrs. Sahana Salagare, Mr. ChristoJain
18ECL38	Digital System DesignLaboratory	Dr. B.Sudharshan, Mr. B. R. Santhosh Kumar
18KVK39/49	Vyavaharika Kannada (Kannada for communication)/	Mr. Thrimurthy
18KAK39/49	Aadalitha Kannada (Kannada for Administration)	

Time Table Co-ordinator

HOD

Principal

PRINCIPAL
K.S. INSTITUTE OF TECHNOLOGY
BENGALURU - 560 109

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE -109

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

III SEMESTER TIME TABLE FOR THE YEAR 2020 (ODD SEMESTER)

W.E.F. : 1/9/2020 **ONLINE TIME TABLE** **CLASS TEACHER : Mrs. SAHANA SALAGARE**

SEC : 'A'

PERIOD	1	2	11.00 AM 11.30 AM	3	4	1.30 PM 2.00 PM	5	6
TIME / DAY	9.00 AM 10.00 AM	10.00 AM 11.00 AM		11.30 AM 12.30 PM	12.30 PM 1.30 PM		2.00 PM 3.00 PM	3.00 PM 4.00 PM
MON	NT (18EC32)	ED (18EC33)	BREAK	MATHS (18MAT31)	DSD (18EC34)	LUNCH BREAK	Vyavaharika Kannada (18KVK39/49) Aadalitha Kannada (18KAK39/49)	PEDAGOGY ((PE&I/ CO&A))
TUE	NT (18EC32)	PE&I (18EC36)		MATHS (18MAT31)	ED (18EC33)		← PLACEMENT →	
WED	DSD (18EC34)	CO&A (18EC35)		MATHS (18MAT31)	PE&I (18EC36)		← ED&I LAB(18ECL37) →	
THU	ED (18EC33)	DSD (18EC34)		MATHS (18MAT31)	CO&A (18EC35)		← DSD LAB(18ECL38) →	
FRI	PE&I (18EC36)	ED (18EC33)		NT (18EC32)	CO&A (18EC35)		PEDAGOGY (DSD)	DIP MATHS 18MATDIP31
SAT	CO&A (18EC35)	NT (18EC32)		PE&I (18EC36)	DSD (18EC34)		PEDAGOGY (NT)	PEDAGOGY (ED)

Sub-Code	Subject Name	Faculty Name
18MATDIP31	Additional Mathematics - I	Mrs. Jalaja . P
18MAT31	Transform Calculus, Fourier Series and Numerical Techniques	Mrs. Pragati. P
18EC32	Network Theory	Mrs. Sahana Salagare
18EC33	Electronic Devices	Mr. B.R.Santhosh Kumar
18EC34	Digital System Design	Dr. B.Sudharshan
18EC35	Computer Organization & Architecture	Mr. Christo Jain
18EC36	Power Electronics & Instrumentation	Mrs. Pragati. P, Mrs. Sahana Salagare, Mr. ChristoJain
18ECL37	Electronic Devices & Instrumentation Laboratory	Dr. B.Sudharshan, Mr. B. R. Santhosh Kuamr
18ECL38	Digital System DesignLaboratory	Mr. Thrimurthy
18KVK39/49	Vyavaharika Kannada **(Kannada for communication)**	
18KAK39/49	Aadalitha Kannada **(Kannada for Administration)**	

Time Table Co-ordinator

HOD

Principal

PRINCIPAL
K.S. INSTITUTE OF TECHNOLOGY
BENGALURU - 560 109

B. E. (EC / TC) ChoiceBasedCreditSystem(CBCS)andOutcomeBasedEducation(OBE) SEMESTER – III			
COMPUTER ORGANIZATION AND ARCHITECTURE			
Course Code	**18EC35**	**CIE Marks**	**40**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**60**
Total Number of Lecture Hours	**40 (08Hours per Module)**	**Exam Hours**	**03**
CREDITS– 03			

Course Learning Objectives: This course will enable students to:

- Explainthebasicsubsystemsofacomputer,theirorganization,structureandoperation.
- Illustrate the concept of programs as sequences of machineinstructions.
- Demonstrate different ways of communicating with I/Odevices
- Describe memory hierarchy and concept of virtualmemory.
- Illustrate organization of simple pipelined processor and other computingsystems.

Module 1	RBT Level
Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Software, Performance – Processor Clock, Basic Performance Equation **(upto 1.6.2 of Chap 1 of Text).** **Machine Instructions and Programs:** Numbers, Arithmetic Operations and Characters, IEEE standard for Floating point Numbers, Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing **(upto 2.4.6 of Chap 2 and 6.7.1 of Chap 6 of Text).**	L1, L2, L3
Module2	
Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions **(from 2.4.7 of Chap 2, except 2.9.3, 2.11 & 2.12 of Text).**	L1, L2, L3
Module3	
Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, EnablingandDisablingInterrupts,HandlingMultipleDevices,ControllingDeviceRequests, DirectMemoryAccess**(upto4.2.4and4.4except4.4.1ofChap4ofText).**	L1, L2, L3
Module4	
Memory System: Basic Concepts, Semiconductor RAM Memories-Internal organization of memory chips, Static memories, Asynchronous DRAMS, Read Only Memories, Cash Memories,VirtualMemories,SecondaryStorage-MagneticHardDisks**(5.1,5.2,5.2.1,5.2.2, 5.2.3, 5.3, 5.5 (except 5.5.1 to 5.5.4), 5.7 (except 5.7.1), 5.9, 5.9.1 of Chap 5 of Text).**	L1, L2, L3
Module5	
Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hardwired Control, Microprogrammed Control **(upto 7.5 except 7.5.1 to 7.5.6 of Chap 7 of Text).**	L1,L2, L3

Course Outcomes: After studying this course, students will be able to:

- Categorize the operations of major subsystems of computer
- Analyze different types of semiconductor memories and secondary memories.
- Analyze ALU and control unit operations.
- Analyze the working of stacks, queues, subroutines and handling different types of interrupts.
- Analyze the aspects of hardwired control and microprogrammed control

Question paper pattern:

- Examination will be conducted for 100 marks with question paper containing 10 full questions, each of 20 marks.
- Each full question can have a maximum of 4 subquestions.
- There will be 2 full questions from each module covering all the topics of the module.
- Students will have to answer 5 full questions, selecting one full question from each module
- The total marks will be proportionally reduced to 60 marks as SEE marks is 60.

Text Book:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002.

Reference Books:

1. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier, 2009.
2. William Stallings: Computer Organization & Architecture, 7th Edition, PHI, 2006.
3. Vincent P. Heuring & Harry F. Jordan: Computer Systems Design and Architecture, 2nd Edition, Pearson Education, 2004.

Web links and Video Lectures:

1. https://www.classcentral.com/course/swayam-computer-organization-and-architecture-a-pedagogical-aspect-9824
2. https://online-learning.harvard.edu/course/computer-architecture-0

K S INSTITUTE OF TECHNOLOGY BANGALORE

DEPARTMENT OF ELECRONICS & COMMUNICATION ENGINEERING

NAME OF THE STAFF : Dr. B Sudarshan

SUBJECT CODE/NAME : 18EC35/COMPUTER ORGANIZATION AND ARCHITECTURE

SEMESTER/YEAR/SEC : III / II/ A & B

ACADEMIC YEAR : 2020-2021

Sl. No.	Topic to be covered	Mode of Delivery	Teaching Aid	No. of Periods	Cumulative No. of Periods	Proposed Date A Section	Proposed Date B Section
MODULE 1: Basic Structure of Computers, Machine Instructions and Programs							
1	**Basic Structure of Computers:** Computer Types, Functional Units, Booting process	L+D	BB, LCD	1	1	2/9/20	1/9/20
2	Basic Operational Concepts, Bus Structures	L+D	BB	1	2	3/9/20	3/9/20
3	Software, Performance – Processor Clock	L+ D	BB	1	3	4/9/20	4/9/20
4	Basic Performance Equation	L+D	BB	1	4	5/9/20	5/9/20
5	**Machine Instructions and Programs:** Numbers, Arithmetic Operations and Characters	L+D	BB	1	5	9/9/20	8/9/20
6	Memory Location and Addresses, Memory Operations	L+D	BB	1	6	10/9/20	10/9/20
7	Instructions and Instruction Sequencing	L+D	BB	1	7	11/9/20	11/9/20
8	IEEE standard for Floating point Numbers	L+D	BB	1	8	12/9/20	12/9/20
9	Writing simple machine instruction	L+D	BB	1	9	16/9/20	15/9/20
10	Branching and condistion codes	L+D	BB	1	10	18/9/20	18/9/20

MODULE 2: Addressing Modes							
11	Addressing Modes	L+ D	BB	1	11	19/9/20	19/9/20
12	Addressing Modes	L+D	BB	1	12	23/9/20	22/9/20
13	Assembly Language	L+D	BB	1	13	24/9/20	24/9/20
14	Basic Input and Output Operations	L+D	BB	1	14	25/9/20	25/9/20
15	Basic Input and Output Operations	L+D	BB	1	15	26/9/20	26/9/20
16	Stacks and Queues	L+D	BB	1	16	01/10/20	01/10/20
17	Subroutines	L+D	BB	1	17	03/10/20	03/10/20
18	Suroutines – parameter passing using stack	L+D	BB	1	18	07/10/20	06/10/20
19	Additional Instructions-Logical-shift, Rotate & multiply/division instructions	L+D	BB	1	19	08/10/20	08/10/20
20	Solving problems on machine instructions with different addressing modes	L+D	BB	1	20	09/10/20	09/10/20
Module 3: Input/output Organization							
21	**Input/Output Organization:** Accessing I/O Devices	L+D	BB	1	21	10/10/20	10/10/20
22	Interrupts – Interrupt Hardware	L+D	BB	1	22	14/10/20	13/10/20
23	Interrupts – Interrupt Hardware	L+I	BB	1	23	15/10/20	15/10/20
24	Enabling and Disabling Interrupts	L+D	BB	1	24	16/10/20	16/10/20
25	Handling Multiple Devices	L+D	BB	1	25	17/10/20	17/10/20
26	Handling Multiple Devices	L+D	BB	1	26	21/10/20	20/10/20
27	Controlling Device Requests	L+D	BB	1	27	22/10/20	22/10/20
28	Controlling Device Requests	L+D	BB	1	28	23/10/20	23/10/20
29	Direct Memory Access	L+D	BB	1	29	24/10/20	24/10/20
30	Direct Memory Access	L+D	BB	1	30	28/10/20	27/10/20
Module 4: Memory System							
31	Basic Concepts	L+D	BB	1	31	29/10/20	29/10/20
32	Semiconductor RAM memories-Internal organization of memory chips	L+D	BB	1	32	5/11/20	5/11/20
33	Static memories	L+D	BB	1	33	6/11/20	6/11/20

34	Asynchronous DRAMs	L+D	BB	1	34	7/11/20	7/11/20
35	Read Only Memories	L+D	BB	1	35	11/11/20	10/11/20
36	Cache Memories	L+D	BB	1	36	12/11/20	12/11/20
37	Virtual Memories	L+D	BB	1	37	13/11/20	13/11/20
38	Secondary storage magnetic hard disks	L+D	BB	1	38	14/11/20	14/11/20
39	Videos on hard disks and its parts	L+I	LCD	1	39	18/11/20	17/11/20
40	Videos on hard disks and its parts	L+D	BB	1	40	19/11/20	19/11/20
	MODULE 5: Basic Processing Unit						
41	Basic Processing Unit: Some fundamental concepts	L+D	BB	1	41	20/11/20	20/11/20
42	Execution of complete instruction	L+D	BB	1	42	21/11/20	21/11/20
43	Multiple bus organization	L+D	BB	1	43	25/11/20	24/11/20
44	Multiple bus organization	L+D	BB	1	44	26/11/20	26/11/20
45	Hardwired control	L+D	BB	1	45	27/11/20	27/11/20
46	Hardwired control	L+D	BB	1	46	28/11/20	28/11/20
47	Microprogrammed control	L+D	BB	1	47	2/12/20	1/12/20
48	Microprogrammed control	L+D	BB	1	48	4/12/20	4/12/20
49	Timing analysis of some simple programs	L+D	BB	1	49	5/12/20	5/12/20
50	Timing analysis of some simple programs	L+D	BB	1	50	16/12/20	15/12/20

Signature of Course Incharge **Signature of Module Coordinator** **Signature of HO**

K S INSTITUTE OF TECHNOLOGY, Bangalore

DEPARTMENT OF ELECTRONICS AMD COMMUNICATION ENGINEERING

ASSIGNMENT QUESTIONS

Academic Year	**2020-21**		
Batch	**2019-23**		
Year/Semester/section	**2/3/A & B**		
Course Code-Title	**18EC35-Computer Organization and Architecture**		
Name of the Instructor	**Dr. B Sudarshan**	**Dept**	**EC**

Assignment No: 1 **Total marks:10**
Date of Issue: 26.9.2020 **Date of Submission: 2.10.2020**

Sl.No	Assignment Questions	K Level	CO	Marks
1.	**Identify** different types of computer & their features.	Applying-K3	CO1	1
2.	**Make use of** a neat block diagram, explain the basic functional blocks of a computer?	Applying-K3	CO1	1
3.	**Obtain** single precision & double precision IEEE floating point representation of **number 85.125**.	Applying-K3	CO1	1
4.	**Make use of** examples to Illustrate Instruction and instruction sequencing.	Applying-K3	CO1	1
5.	**Make use of** examples to Explain byte addressing, big endian and little endian assignment..	Applying-K3	CO1	1
6.	**Identify** the parameters and their relative values for the improvement of computer performance using performance equation of the processor.	Applying-K3	CO1	1
7.	**Make use of** examples to explain different addressing modes .	Applying-K3	CO2	1
8.	**Make use of** examples to explain Assembler Directives.	Applying-K3	CO2	1
9.	**Develop** an assembly language program to add the scores of N students in each test and store it in memory location SUM1, SUM2, SUM3?	Applying-K3	CO2	1
10.	**Make use of** examples to explain use of decimal, binary and hexadecimal numbers in Instructions.	Applying-K3	CO2	1

Course In charge **HOD**

K S INSTITUTE OF TECHNOLOGY, Bangalore

DEPARTMENT OF ELECTRONICS AMD COMMUNICATION ENGINEERING

ASSIGNMENT – 1 SCHEME

Academic Year	**2020-21**		
Batch	**2019-23**		
Year/Semester/section	**2/3/A & B**		
Course Code-Title	**18EC35-Computer Organization and Architecture**		
Name of the Instructor	**Dr. B Sudarshan**	**Dept**	**EC**

Sl.No	Assignment Questions
1.	1 Personal Computers, 2.. Notebook Computers 3 Workstations 4 Enterprise Systems. 4a. Servers 4b. Supercomputers Explanation of above briefly.
2.	Figure showing below five functional units of computer. 1. Input, 2, output, 3. Memory, 4. ALU, 5 Control Brief explanation of above units.
3.	**(i) 85.125** in binary ;floating point representation – single precision **85.125 = $1010101.001_{(2)}$ = 1.010101001x2^6 , E = 6, so, E' = E+127 = 133 = 10000101 (8 bits)** **= 0(S(+/-) -1bit) 10000101 (E'-8 bits) 0101010010...0(M-23 bits) = +1.0101010010...0** (23 bits)x 2^6 (ii) In double precision, **E' = E+1023 = 6+1023 = 1029 = 1000 0000 101(11 bits)** = 0(+) **10000000101(E'-11 bits) 0101010010...0(M-52 bits) = +1.0101010010...0** (52 bits)x2^6
4.	Computer must have instructions capable of performing 4 types of operations. • Data transfer between the memory and the registers • ALU operations on data • Program sequencing and control • I.O transfers Explanation of a sequential execution of a program with example
5.	In **byte addressing**, successive addresses refer to successive byte locations in the memory. Byte locations have addresses 0,1,2,. So, a computer can access individual bytes stored in computer memory by using byte addresses. While in word addressing either little endian or Big endian assignment is used. **Big endian** : Lower byte address are used for the more significant bytes of the word/ **Little endian**: Lower byte address are used for the less significant bytes of the word Examples.
6.	T – Processor time required to execute a program that has been prepared in high-level language N – Number of actual machine language instructions needed to complete the execution (note: loop) S – Average number of basic steps needed to execute one machine instruction. Each step completes in one clock cycle R – Clock rate The program execution time T is given by **T = (NxS)/R** Parameters, N, S & R are dependent on each other. To achieve high performance, the computer designer must reduce the value of T which means reducing N & S, increasing R. The value of N is reduced if the source program is compiled into fewer machine instructions. The value of S is reduced in instruction have a smaller number of basic steps to perform. The value of R can be increased by using a higher frequency clock. Care has to be taken while modifying values since changes in the parameter may affect the other. List two possibilities for increasing the clock rate R.

7.

Table 2.1 Generic addressing modes

Name	Assembler syntax	Addressing function
Immediate	#Value	Operand = Value
Register	Ri	EA = Ri
Absolute (Direct)	LOC	EA = LOC
Indirect	(Ri) (LOC)	EA = [Ri] EA = [LOC]
Index	X(Ri)	EA = [Ri] + X
Base with index	(Ri,Rj)	EA = [Ri] + [Rj]
Base with index and offset	X(Ri,Rj)	EA = [Ri] + [Rj] + X
Relative	X(PC)	EA = [PC] + X
Autoincrement	(Ri)+	EA = [Ri]; Increment Ri
Autodecrement	-(Ri)	Decrement Ri; EA = [Ri]

EA = effective address
Value = a signed number

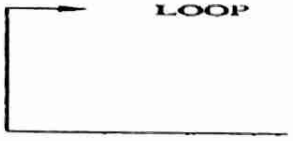
8.

ASSEMBLER DIRECTIVES

- **Directives** are the assembler commands to the assembler concerning the program being assembled.
- These commands are not translated into machine opcode in the object-program.
- **EQU** informs the assembler about the value of an identifier (Figure: 2.18).
  Ex: *SUM EQU 200* ;Informs assembler that the name SUM should be replaced by the value 200.
- **ORIGIN** tells the assembler about the starting-address of memory-area to place the data block.
  Ex: ORIGIN 204 ;Instructs assembler to initiate data-block at memory-locations starting from 204.
- **DATAWORD** directive tells the assembler to load a value into the location.
  Ex: *N DATAWORD 100* ;Informs the assembler to load data 100 into the memory-location N(204).
- **RESERVE** directive is used to reserve a block of memory.
  Ex: *NUM1 RESERVE 400* ;declares a memory-block of 400 bytes is to be reserved for data.
- **END** directive tells the assembler that this is the end of the source-program text.
- **RETURN** directive identifies the point at which execution of the program should be terminated.
- Any statement that makes instructions or data being placed in a memory-location may be given a **label**. The label(say N or NUM1) is assigned a value equal to the address of that location.

Examples

9.



```
        Move        #LIST,R0
        Clear       R1
        Clear       R2
        Clear       R3
        Move        N,R4
LOOP    Add         4(R0),R1
        Add         8(R0),R2
        Add         12(R0),R3
        Add         #16,R0
        Decrement   R4
        Branch>0    LOOP
        Move        R1,SUM1
        Move        R2,SUM2
        Move        R3,SUM3
```

Figure 2.15 Indexed addressing used in accessing test scores in the list in Figure 2.14.

10.

Most assemblers allow numerical values to be specified indifferent ways.
For example, Consider a number 93 is used as an immediate operand
It can be given in following ways:

- Decimal: ADD#93,R1
- Binary: ADD#%01011101,R1
- Hexadecimal: ADD#$5D,R1

Course In charge

HOD

K S Institute of Technology, Bangalore

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ASSIGNMENT -2

Academic Year	**2020-21**		
Batch	**2019-23**		
Year/Semester/section	**2/3/A & B**		
Course Code-Title	**18EC35-Computer Organization and Architecture**		
Name of the Instructor	**Dr. B Sudarshan**	**Dept**	**ECE**

Assignment No: 2 **Total marks: 10**

Date of Issue: 23.10.2020 **Date of Submission: 6.11.2020**

Sl. No	Assignment Questions	K Level	CO	Marks
1.	a. **Develop** an assembly language program to read a line of character and display it. b. **Identify** the differences between Stack and Queue with examples	Applying-K3	CO2	1
2.	**(i).** Registers R1 and R2 of a computer contain the decimal values 1200 and 4600. **Obtain** the effective address of the memory operand in each of the following instructions **(a) Load 20(R1),R5** **(b) Move #3000,R5** **(c) Store R5,30(R1,R2)** **(d) Add -(R2),R5** **(e) Subtract (R1)+,R5** **(ii).** Make **use of** examples to explain different types of shift and rotate operations with examples.	Applying-K3	CO2	1
3.	**.(i) Make use of** figures to explain the registers in keyboard and display interfaces **(ii) Identify** the differences between Subroutine and Interrupt Service Routine.	Applying-K3	CO3	1
4.	**Identify** the methods of enabling and disabling interrupts & also define Interrupts	Applying-K3	CO3	1
5.	**Identify** the differences between Daisy Chain scheme with the scheme with combination of Daisy Chain , individual request and acknowledge lines for handling multiple interrupt requests.	Applying-K3	CO3	1
6.	**(i).** What is interrupt nesting? **(ii). Make use of** a neat block diagram to explain the implementation of Interrupt priority using individual request and acknowledge lines.	Applying-K3	CO3	1
7.	**Develop** an Assembly language program to read an input line from the keyboard and store the characters in successive byte locations in the memory starting at location LINE.	Applying-K3	CO3	1
8.	**Make use of** a neat diagram to explain DMA controller	Applying-K3	CO3	1
9.	**Make use of** figure to explain memory access by the processor	Applying-K3	CO4	1
10.	**Make use of** a neat diagram to Illustrate the organization of 1Kx1 memory chip	Applying-K3	CO4	1

Course in charge **HOD**

K S Institute of Technology, Bangalore

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ASSIGNMENT–2-Answer Keys

Academic Year	2020-21		
Batch	2019-23		
Year/Semester/section	2/3/A & B		
Course Code-Title	18EC35-Computer Organization and Architecture		
Name of the Instructor	Dr. B Sudarshan	Dept	EC

Sl.No: 1.

a. **Develop** an assembly language program to read a line of character and display it.

```
        Move      #LOC,R0        Initialize pointer register R0 to point to the
                                   address of the first location in memory
                                   where the characters are to be stored.
READ    TestBit   #3,INSTATUS    Wait for a character to be entered
        Branch=0  READ             in the keyboard buffer DATAIN.
        MoveByte  DATAIN,(R0)    Transfer the character from DATAIN into
                                   the memory (this clears SIN to 0).
ECHO    TestBit   #3,OUTSTATUS   Wait for the display to become ready.
        Branch=0  ECHO
        MoveByte  (R0),DATAOUT   Move the character just read to the display
                                   buffer register (this clears SOUT to 0).
        Compare   #CR,(R0)       Check if the character just read is CR
                                   (carriage return). If it is not CR, then
        Branch≠0  READ             branch back and read another character.
                                   Also, increment the pointer to store the
                                   next character.
```

b. **Identify** the differences between Stack and Queue with examples

Stack	Queue
One end is fixed while other end rises and falls as data are pushed and popped.	Both ends of queues move to higher addresses as data is pushed and popped.
A single pointer is needed to point to the top of the stack	Two pointers are needed to keep track of two ends of the queue.
In stack care must be taken to see to it that nothing is pushed into a stack which is full and nothing is popped out from the stack which is empty.	Queue would continuously move through the memory of the computer in the direction of higher addresses. This can be limited by using a *circular buffer*. Let us assume that memory addresses from BEGINNING to END are assigned to the queue. The first entry in the queue is entered into BEGINNING and successive entries are appended at higher addresses. By the time queue reaches END, space would be created in the beginning. So back pointer is reset to BEGINNING and the process continues.

Sl.No: 2.

(i). Registers R1 and R2 of a computer contain the decimal values 1200 and 4600. **Obtain** the effective address of the memory operand in each of the following instructions

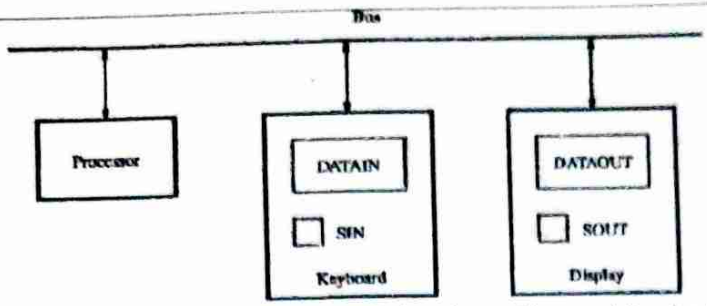
(a) Load 20(R1),R5
(b) Move #3000,R5
(c) Store R5,30(R1,R2)
(d) Add -(R2),R5
(e) Subtract (R1)+,R5

LOAD 20(R1), R5 E A = 20+[R1] = 20+1200 = 1220
MOVE #3000, R1 E A = 3000
STORE R5, 30(R1,R2) E A = 30+[R1]+[R2] = 30+1200+4600 = 5830
ADD -(R2), R5 EA = 4600-1 = 4599
SUBTRACT (R1)+, R5 E A = 1200

(ii). Make **use of** examples to explain different types of shift and rotate operations with examples.
Explanation of Logical Shift , Arithmatic shift(left and Right), Rotate & Rotate with & without carry(left & Right)

Sl.No: 3.

.(i) Make use of figures to explain the registers in keyboard and display interfaces



(ii) Identify the differences between Subroutine and Interrupt Service Routine.

Subroutine	Interrupt Service Routine
It is a portion of the code within a large program which performs a specific task.	ISR are to handle hardware interrupts.
It runs when we call it.	It runs whenever a certain signal occurs.
We know where the subroutine runs because we call it.	We don't know when ISR will be executed.

4. **Identify** the methods of enabling and disabling interrupts & also define Interrupts.

The first possibility is to have the processor hardware ignore the interrupt request line until the execution of the first instruction of the ISR has been completed. The programmer can ensure that no further interruptions will occur by using interrupt-disable instruction as the first instruction of the ISR. The interrupt-enable instruction is the last instruction of the ISR before the Return from interrupt instruction.

In second option (for processor with only one interrupt request line), the processor can automatically disable interrupts before starting the execution of the ISR. One bit of program status register (PS), called interrupt enable indicates whether interrupts are enabled. After saving the contents of PS on stack, the processor clears the interrupt enable bit in its PS register thus disabling further interrupts. When the Return-from-Interrupts instruction is executed, the contents of PS are restored from the stack setting the Interrupt-enable bit back to 1.

In the third option, the processor has a special interrupt request line for which the interrupt handling circuit responds only to the leading edge of the signal. Such a line is said to be *edge triggered*. In this case, the processor will receive only one request. Hence no danger of multiple interruptions.

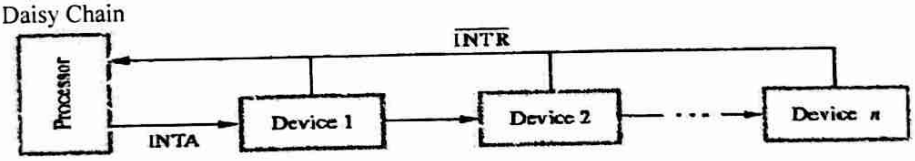
Definiition

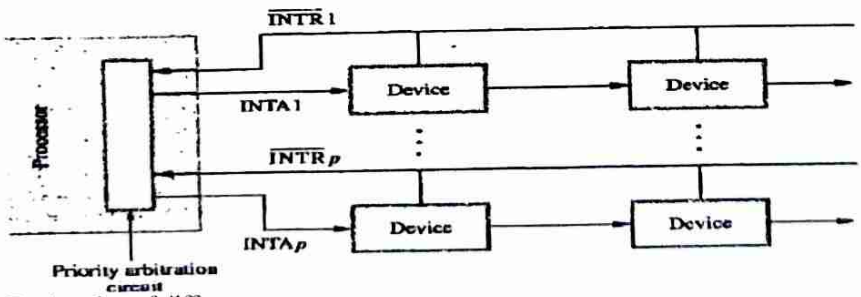
Sometimes the processor time will be wasted while waiting for an I/O device to become ready.

This time can be eliminated by I/O device by sending a hardware signal called an *interrupt* to the processor. A bus control line called *interrupt signal line* is dedicated for this purpose.

5. **Identify** the differences between Daisy Chain scheme with the scheme with combination of Daisy Chain , individual request and acknowledge lines for handling multiple interrupt requests.

Daisy Chain





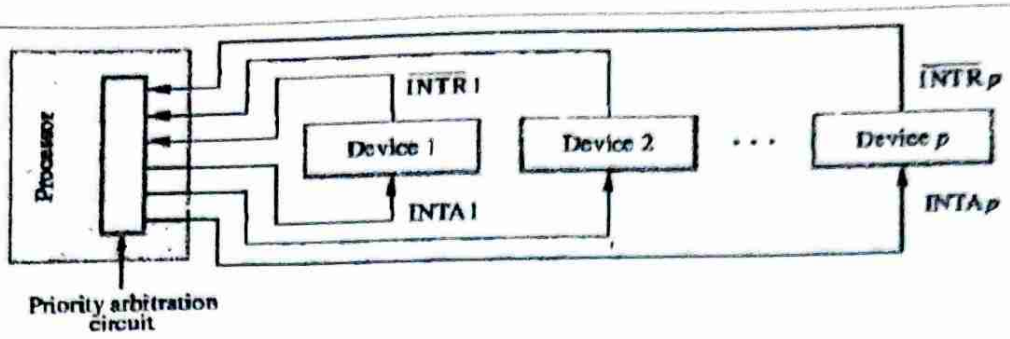
Explanation of differences

6. (i). What is interrupt nesting?

Occurance and servicing of an interrupt while the processor is already running interrupt service routing of another interrupt.

Explanation with example.

(ii). Make use of a neat block diagram to explain the implementation of Interrupt priority using individual request and acknowledge lines.



Explanation

7. **Develop** an Assembly language program to read an input line from the keyboard and store the characters in successive byte locations in the memory starting at location LINE.

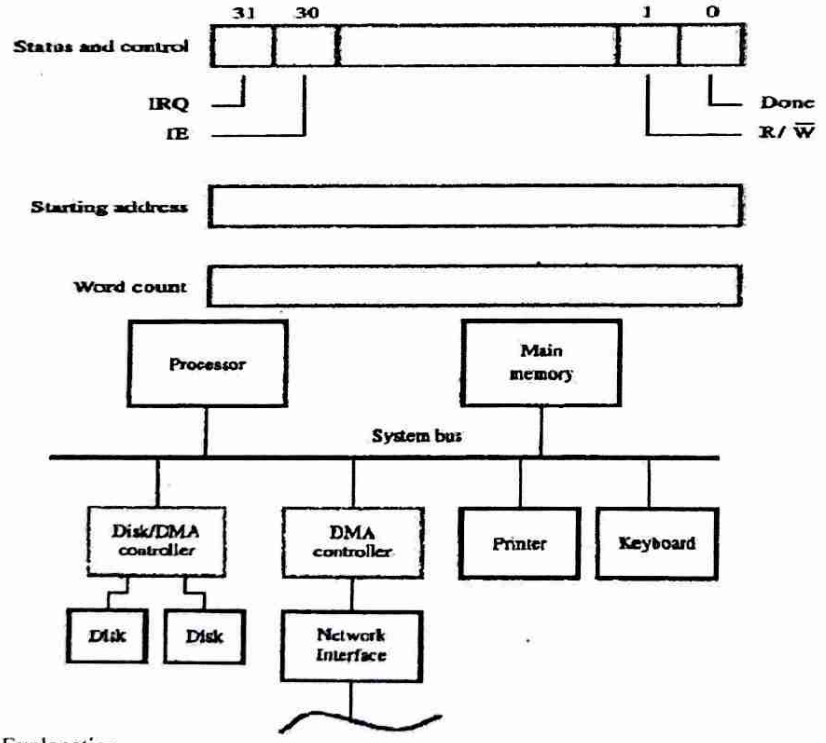
Main Program

```
        Move        #LINE,PNTR      Initialize buffer pointer.
        Clear       EOL             Clear end-of-line indicator.
        BitSet      #2,CONTROL      Enable keyboard interrupts.
        BitSet      #0,PS           Set interrupt-enable bit in the PS.
        :
```

Interrupt-service routine

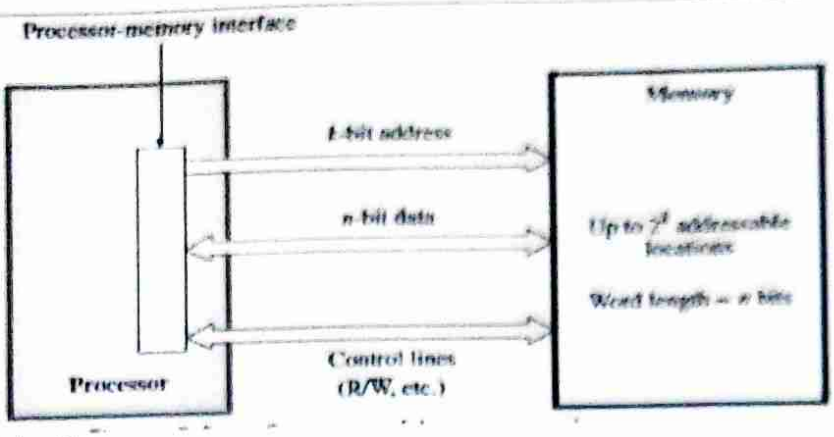
```
READ    MoveMultiple    R0-R1,-(SP)     Save registers R0 and R1 on stack.
        Move            PNTR,R0         Load address pointer.
        MoveByte        DATAIN,R1       Get input character and
        MoveByte        R1,(R0)+          store it in memory.
        Move            R0,PNTR         Update pointer.
        CompareByte     #$0D,R1         Check if Carriage Return.
        Branch≠0        RTRN
        Move            #1,EOL          Indicate end of line.
        BitClear        #2,CONTROL      Disable keyboard interrupts.
RTRN    MoveMultiple    (SP)+,R0-R1     Restore registers R0 and R1.
        Return-from-interrupt
```

8. **Make use of** a neat diagram to explain DMA controller.



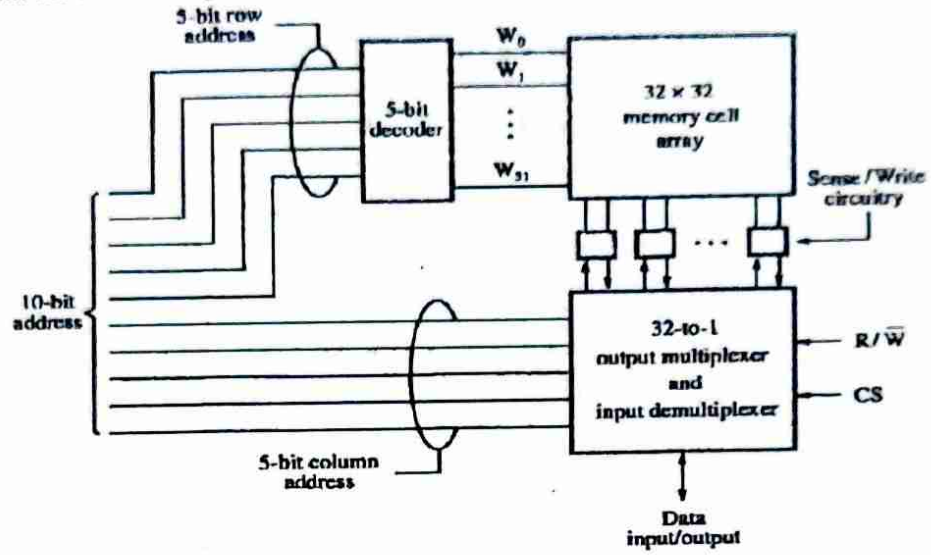
Explanation

9. **Make use of** figure to explain memory access by the processor.



Explanation

10. **Make use of** a neat diagram to Illustrate the organization of 1Kx1 memory chip.



Explanation

Course In charge

HOD

K S Institute of Technology, Bangalore

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ASSIGNMENT - 3

Academic Year	2020-21		
Batch	2019-23		
Year/Semester/section	2/3/A & B		
Course Code-Title	18EC35-Computer Organization and Architecture		
Name of the Instructor	Dr. B Sudarshan	Dept	ECE

Assignment No: 3 **Total marks: 10**

Date of Issue: 30.10.2020 **Date of Submission: 20.12.2020**

Sl. No	Assignment Questions	K Level	CO	Marks
1.	**Make use of** a neat diagram & explain the working of CMOS memory cell	Applying-K3	CO4	1
2.	**Analyze** different types of ROMs..	Applying-K3	CO4	1
3.	**Make use of** figure & explain the principle of working of magnetic disk.	Analyzing-K4	CO4	1
4.	**Identify and** explain RAID levels of RAID Disk Arrays	Applying-K3	CO4	1
5.	**Make use of** figure & explain the single bus organization of data path inside a processor	Applying-K3	CO5	1
6.	**Make use of** figure & explain the implementation of 1 bit register.	Applying-K3	CO5	1
7.	**Construct** the sequence of operations and timing diagram for the instruction MOV (R1), R2	Applying-K3	CO5	1
8.	a. **Make use of** figure & explain multiple bus organization of CPU. b. **Construct** the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization.	Applying-K3	CO5	1
9.	**Make use of** figure & explain the organization of control unit for unconditional branch instruction	Applying-K3	CO5	1
10.	**Make use of** figure & explain Hardwired Control Unit Organization in a processing unit.	Applying-K3	CO5	1

Course in charge

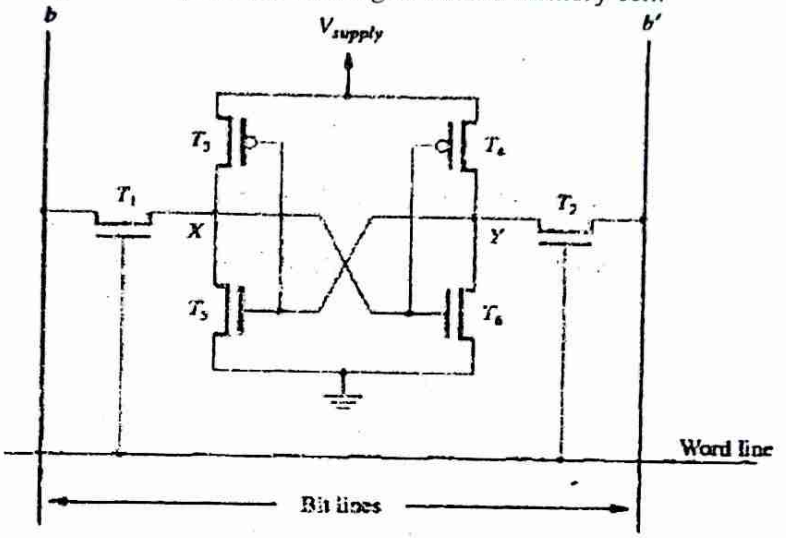
HOD

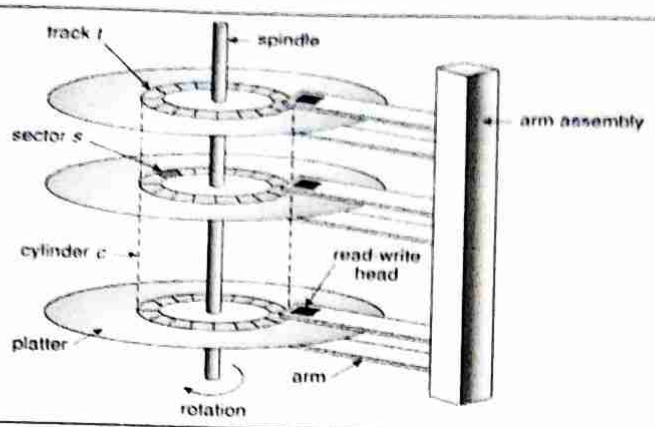
K S Institute of Technology, Bangalore

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

ASSIGNMENT – 3 Answer Keys

Academic Year	2020-21		
Batch	2019-23		
Year/Semester/section	2/3/A & B		
Course Code-Title	18EC35-Computer Organization and Architecture		
Name of the Instructor	Dr. B Sudarshan	Dept	EC

Sl.No	Answer keys
1.	**Make use of** a neat diagram & explain the working of CMOS memory cell.  Explaination
2.	**Analyze** different types of ROMs. ROM – Read Only Memory-Programmed during manufacturing PROM- Programmable Read Only Memory-Programmed by burning fuses by user only once EPROM – Erasable Programmable Read Only Memory-can be Programmed by inducing charge and erased by UV light by user of many times A disadvantage of EPROMs is that the chip must be physically removed from the circuit for reprogramming and erasing. Hence Electrically Erasable PROMs (EEPROMs) are used. EEPROM – Electrically Erasable Programmable Read Only Memory In EEPROMs the chips need not be removed and it is possible to erase the cell contents selectively. The only disadvantage of EEPROMs is that different voltages are needed for erasing, writing and reading the stored data
3.	**Make use of** figure & explain the principle of working of magnetic disk. Hard disk stores information in the form of magnetic fields. Data is stored digitally in the form of tiny magnetized regions on the platter where each region represents a bit. To write a data on the hard disk, a magnetic field is placed on the tiny field in one of these two polarities: N-S – If North Pole arrives before the south pole and S-N – if the south pole arrives before the north pole while the field is accessed. An orientation in the one direction (like N-S) can represent the '1' while the opposite orientation (S-N) represents "0". This polarity is sensed by integrated controllers built within the hard disk. 1. It consists of one or more disks mounted on a common spindle. 2. A thin magnetic film is deposited on either side of each disk. 3. The disks are placed in a rotary drive so that the magnetized surfaces move in close Proximity to read/write heads. This is shown in the figures below.



4. **Identify and** explain RAID levels of RAID Disk Arrays

RAID stands for **Redundant Array of Inexpensive Disks**. It is a storage system based on multiple disks proposed by University of California-Berkeley.

It has six different configurations known as RAID levels.

RAID 0 is the basic configuration to enhance performance

o In RAID 0 configuration a single large file is broken up into smaller pieces and are stored on different disks. This is called as **data striping**.

o All disks can deliver their data in parallel.

o The total transfer time of the file is equal to the transfer time that would be required in a single disk system divided by the number of disks used in the array.

o As each disk operates independently, access times vary and buffering of accessed pieces of data is needed so that the complete file can be reassembled.

RAID 1 is intended to provide better reliability by storing identical copies of data on two disks rather than just one. If one disk drive fails, the other one can be used. This is however costly because of duplication.

RAID 2, RAID 3, RAID 4 levels achieve increased reliability through various parity checking schemes without duplication.

RAID 5 also has parity based error recovery schemes but this information is distributed among all schemes.

Some hybrid arrangements are also available which are the combination of some of the above mentioned configurations.

5. **Make use of** figure & explain the single bus organization of data path inside a processor.

The figure below shows single bus organization of the data path inside a processor.

The data and address lines of external memory bus are connected to processor via MDR and MAR respectively.

MDR has two inputs and two outputs. It can receive inputs as well as send outputs on memory bus or internal processor bus.

The input of MAR is internal bus and output is external bus.

The control lines of the memory bus are connected to the instruction decoder and control logic block. This unit is responsible for issuing signals that control operations of all units.

The registers R0 to R(n-1) are general purpose registers. Their numbers vary from one processor to another processor.

There are some special purpose registers like index register and stack pointer.

The registers Y, Z and Temp in the figure are used by processor for temporary storage during execution of some instruction. They are never used for storing data generated by one instruction for later use by another instruction.

The Multiplexer selects either the output of register Y or the constant value 4 to provide as input to the ALU. Constant 4 is used to increment the contents of program counter.

The instruction decoder and control logic unit is responsible for implementing the actions specified by the instructions loaded in the IR register.

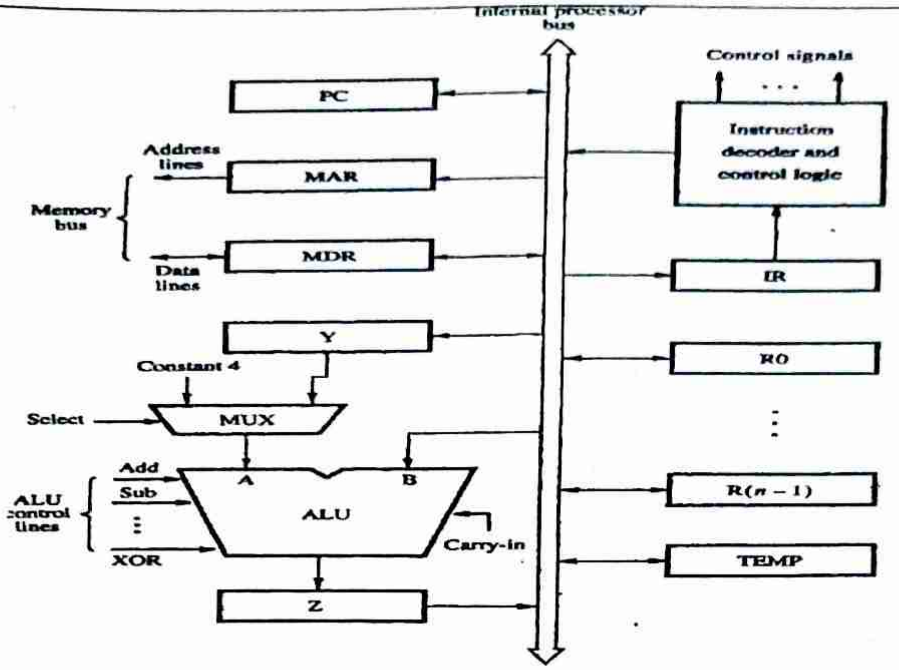
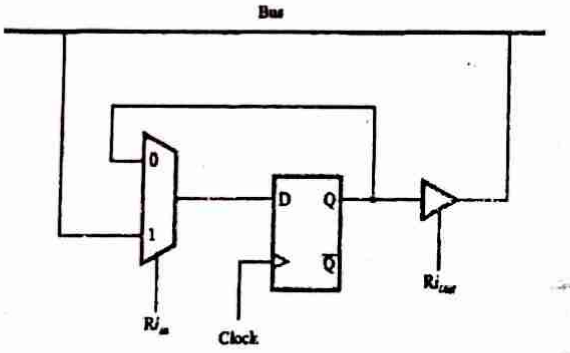


Figure 7.1 Single-bus organization of the datapath inside a processor.

6. **Make use of** figure & explain the implementation of 1 bit register.

The implementation of one bit register Ri is shown in figure below.

- o A two input multiplexer is used to select the data applied to the input of edge triggered D flipflop.
- o When Rin is 1, mux selects data on the bus. This data will be loaded into the flip flop in the rising edge of the clock.
- o When Rin is 0, mux feeds back the value currently stored in the flipflop
- o The Q output of flip flop is connected to the bus via a tristate gate.
- o When Riout is 0, gate output is in high impedance state (electrically disconnected).
- o When Riout is 1, gate drives the bus to 0 or 1 depending on the value of Q



7. **Construct** the sequence of operations and timing diagram for the instruction MOV (R1), R2

1. **MAR ← [R1]**
2. **Start a Read operation on the memory bus**
3. **Wait for the MFC response from the memory**
4. **Load MDR from the memory bus**
5. **R2 ← [MDR]**

8. **Make use of** figure & explain multiple bus organization of CPU.

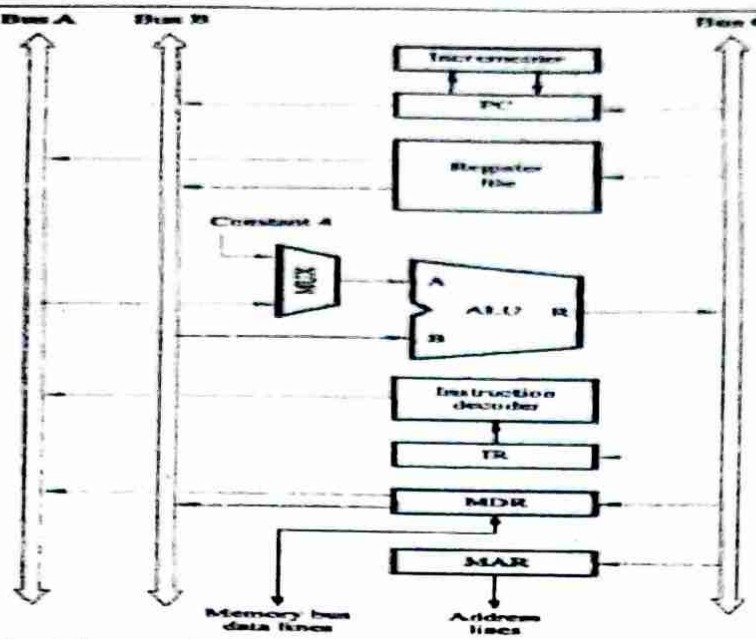
The fig below shows three bus structure used to connect registers and ALU.

All general purpose registers are combined into a single block called **register file**.

The register file has 3 ports. The two output ports allow the contents of two different registers to be placed on bus A and B. The input port allows the contents of bus C to be placed into a third register during the same clock cycle.

Buses A and B are normally used for passing input operands to ALU. Bus C passes the result to the destination. This arrangement avoids the need for registers Y and Z. If needed ALU may simple pass one of its two operands unmodified to bus C by using control signals R=A or R=B.

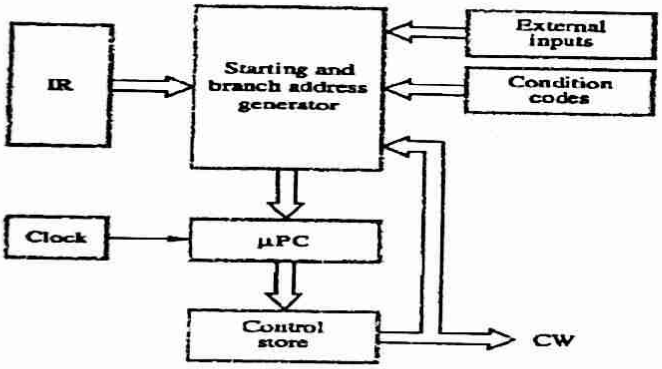
The second feature in the figure below is Incrementer unit which is used to increment PC



Construct the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization.

Step	Action
1	PC_{out}, R=B, MAR_{in}, Read, IncPC
2	WMFC
3	MDR_{outB}, R=B, IR_{in}
4	$R4_{outA}$, $R5_{outB}$, SelectA, Add, $R6_{in}$, End

9. **Make use of** figure & explain the organization of control unit for unconditional branch instruction.

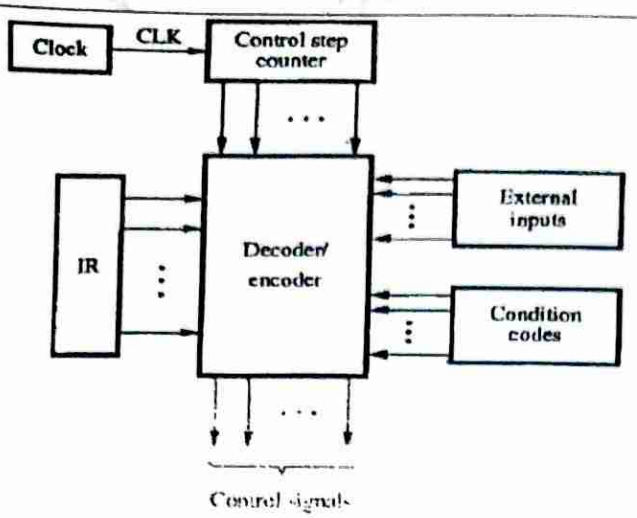


The starting address generator becomes the starting and the branch address generator. This block loads a new address into the µPC when a microinstruction instructs it to do so. The inputs to this block consists of the external inputs and condition codes as well as contents of IR.

The µPC is incremented every time a new microinstruction is fetched from the microprogram memory except

- o When new instruction is loaded into IR, its starting address is loaded into µPC.
- o During branch microinstructions, µPC is loaded with branch address.
- o When End microinstruction is encountered, µPC is loaded with the address of first CW in the micro routine.

10. **Make use of** figure & explain Hardwired Control Unit Organization in a processing unit.



The decoder/encoder block in the above figure is a combinational circuit that generates the required control outputs depending on the state of all its inputs. A more detailed block diagram by separating decoder and encoder blocks is shown in the fig. below.

The step decoder provides a separate signal line for each step/time slot.

The output of Instruction Decoder consists of separate lines for each machine instruction.

Depending on the instruction any one of INS1 to INSm is set to 1 and all other lines are set to 0.

The input signals to the encoder block are combined to generate individual control signals Yin, PCout, Add, End and so on.

Course In charge

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

I SESSIONAL TEST QUESTION PAPER 2020-21 ODD SEMESTER

Set A

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration : 90 Minutes

Semester: III A & B
Course Code: 18EC35
Date: 07-10-2020
Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Make use of** a neat block diagram to explain the basic functional blocks of a computer?	6	CO1	Applying-K3
(b)	**Obtain** (i) **single precision** (ii) **double precision** IEEE floating point representation for the number **85.125**.	6	CO1	Applying-K3
(c)	**Make use of** examples to explain byte addressing, big endian and little endian assignment..	6	CO1	Applying-K3
	OR			
2(a)	**Identify** different types of computer & their features.	6	CO1	Applying-K3
(b)	**Identify** the parameters and their relative values for the improvement of computer performance using performance equation of the processor.	6	CO1	Applying-K3
(c)	**Make use of** figures to explain bus structures	6	CO1	Applying-K3
	PART-B			
3(a)	**Make use of** examples to explain following addressing modes. (i) Immediate (ii) Register (iii) Absolute (iv) Indirect	6	CO2	Applying-K3
(b)	**Make use of** examples to explain Assembler Directives.	6	CO2	Applying-K3
	OR			
4(a)	**Develop** an assembly language program to add the scores of N students in each test and store it in memory location SUM1, SUM2, SUM3?	6	CO2	Applying-K3
(b)	**Make use of** examples to explain following addressing modes. (i) Index Iii) Relative (iii) Auto increment (iv) Auto decrement	6	CO2	Applying-K3

Email answer scripts with file name format **USN_18EC35_IA1.pdf** to **18ec35ia1@gmail.com**

Course in charge **Module Coordinator** **HOD**

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

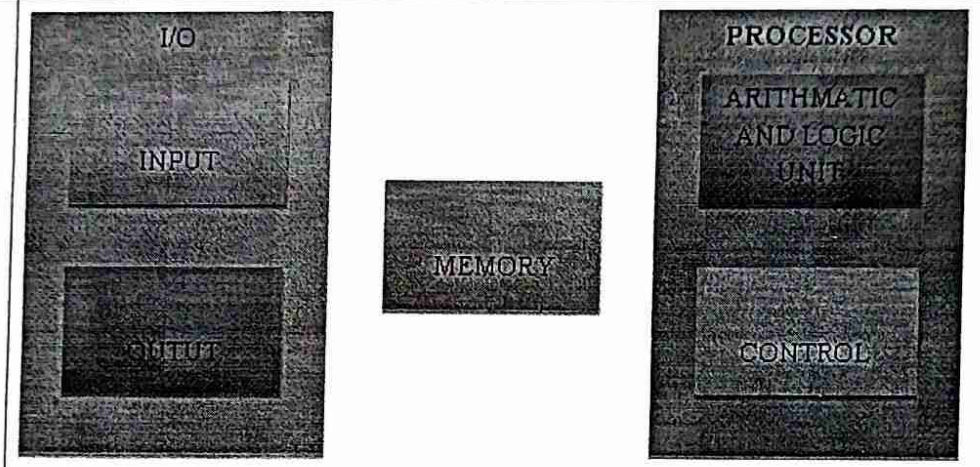
I SESSIONAL TEST SCHEME 2020-21 ODD SEMESTER

Set A

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE

Semester: III A & B
Course Code: 18EC35
Max. Marks: 30

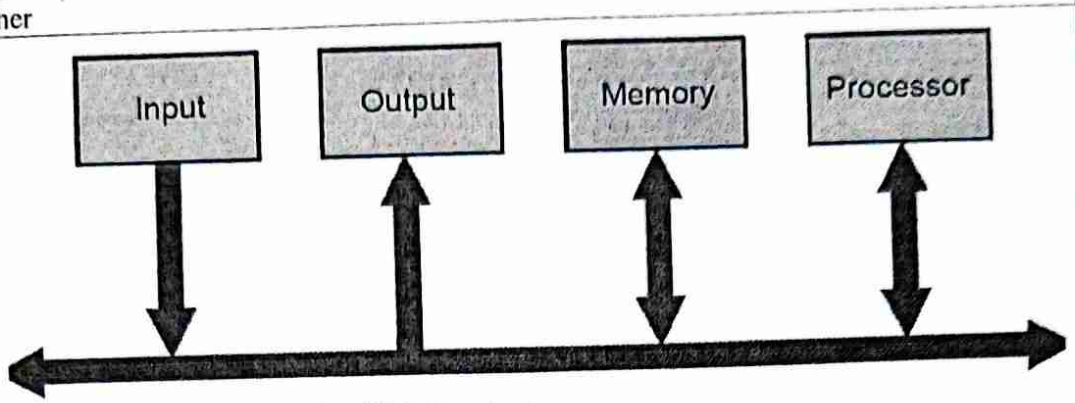
Note: Answer ONE full question from each part.

Q No.	Point	Marks
1(a)	 Explanation of Input, output, mrmoty ALU and Control unit.	6
(b)	(i) 85.125 in binary ;floating point representation – single precision 85.125 = $1010101.001_{(2)}$ = $1.010101001x2^6$, E = 6, so, E' = E+127 = 133 = 10000101 (8 bits) = 0(S(+/-) -1bit) 10000101 (E'-8 bits) 0101010010...0(M-23 bits) = +1.0101010010...0 (23 bits)x 2^6 (ii) In double precision, E' = E+1023 = 6+1023 = 1029 = 1000 0000 101(11 bits) = 0(+) 10000000101(E'-11 bits) 0101010010...0(M-52 bits) = +1.0101010010...0 (52 bits)x2^6	6
(c)	**In byte addressing**, successive addresses refer to successive byte locations in the memory. Byte locations have addresses 0,1,2,. So, a computer can access individual bytes stored in computer memory by using byte addresses. While in word addressing either little endian or Big endian assignment is used. **Big endian** : Lower byte address are used for the more significant bytes of the word/ **Little endian**: Lower byte address are used for the less significant bytes of the word Examples.	6
2(a)	1 Personal Computers, 2.. Notebook Computers 3 Workstations 4 Enterprise Systems. 4a. Servers 4b. Supercomputers Explanation of above briefly.	6
(b)	T – Processor time required to execute a program that has been prepared in high-level language N – Number of actual machine language instructions needed to complete the execution (note: loop) S – Average number of basic steps needed to execute one machine instruction. Each step	6

$$T = (NxS)/R$$

Parameters, N, S & R are dependent on each other.
To achieve high performance, the computer designer must reduce the value of T which means reducing N & S, increasing R.
The value of N is reduced if the source program is compiled into fewer machine instructions.
The value of S is reduced in instruction have a smaller number of basic steps to perform.
The value of R can be increased by using a higher frequency clock.
Care has to be taken while modifying values since changes in the parameter may affect the other

(c)



Explanation of single bus and multiple bus structures. — 6

3(a)

Name	Assembler syntax	Addressing function
Immediate	#Value	Operand = Value
Register	Ri	EA = Ri
Absolute (Direct)	LOC	EA = LOC
Indirect	(Ri) (LOC)	EA = [Ri] EA = [LOC]

6

(b)

ASSEMBLER DIRECTIVES

- **Directives** are the assembler commands to the assembler concerning the program being assemb
- These commands are not translated into machine opcode in the object-program.

	Memory address label	Operation	Addressing or data information
Assembler directives	SUM	EQU	200
		ORIGIN	204
	N	DATAWORD	100
	NUM1	RESERVE	400
		ORIGIN	100
Statements that generate machine instructions	START	MOVE	N,R1
		MOVE	#NUM1,R2
		CLR	R0
	LOOP	ADD	(R2),R0
		ADD	#4,R2
		DEC	R1
		BGTZ	LOOP
		MOVE	R0,SUM
Assembler directives		RETURN	
		END	START

6

4(a)			6

```
                Move        #LIST,R0
                Clear       R1
                Clear       R2
                Clear       R3
                Move        N,R4
        LOOP    Add         4(R0),R1
                Add         8(R0),R2
                Add         12(R0),R3
                Add         #16,R0
                Decrement   R4
                Branch>0    LOOP
                Move        R1,SUM1
                Move        R2,SUM2
                Move        R3,SUM3
```

(b)

Name	Assembler syntax	Addressing function
Index	X(R*i*)	EA = [R*i*] + X
Relative	X(PC)	EA = [PC] + X
Autoincrement	(R*i*)+	EA = [R*i*]; Increment R*i*
Autodecrement	–(R*i*)	Decrement R*i*; EA = [R*i*]

EA = effective address
Value = a signed number

6

Course in charge

Module Coordinator

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109
I SESSIONAL TEST QUESTION PAPER 2020-21 ODD SEMESTER

Set B

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration : 90 Minutes

Semester: III A & B
Course Code: 18EC35
Date:07-10-2020

Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Make use of** a neat diagram to explain the basic operational concept of a computer?	6	CO1	Applying-K3
(b)	**Identify** the parameters and their relative values for the improvement of computer performance using performance equation of the processor.	6	CO1	Applying-K3
(c)	**Make use of** examples to explain byte addressing, big endian and little endian assignment..	6	CO1	Applying-K3
	OR			
2(a)	**Identify** different types of computer & their features.	6	CO1	Applying-K3
(b)	**Obtain** (i) **single precision** & (ii) **double precision** IEEE floating point representation for the number **85.125**.	6	CO1	Applying-K3
(c)	**Make use of** figures to explain bus structures	6	CO1	Applying-K3
	PART-B			
3(a)	**Make use of** examples to explain following addressing modes. (i) Immediate (ii) Register (iii) Absolute (iv) Indirect	6	CO2	Applying-K3
(b)	**Make use of** examples to explain Assembler Directives.	6	CO2	Applying-K3
	OR			
4(a)	**Make use of** examples to explain following addressing modes. (i) Index Iii) Relative (iii) Auto increment (iv) Auto decrement	6	CO2	Applying-K3
(b)	**Develop** an assembly language program to add the scores of N students in each test and store it in memory location SUM1, SUM2, SUM3?	6	CO2	Applying-K3

Email answer scripts with file name format **USN_18EC35_IA1.pdf** to **18ec35ia1@gmail.com**

Course in charge Module Coordinator HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

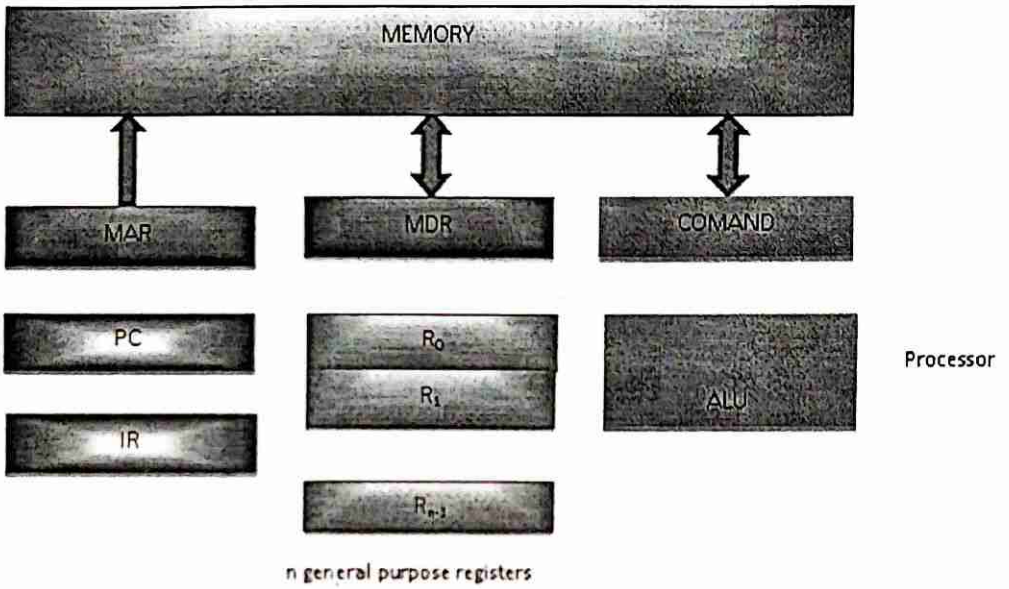
I SESSIONAL TEST SCHEME 2020-21 ODD SEMESTER

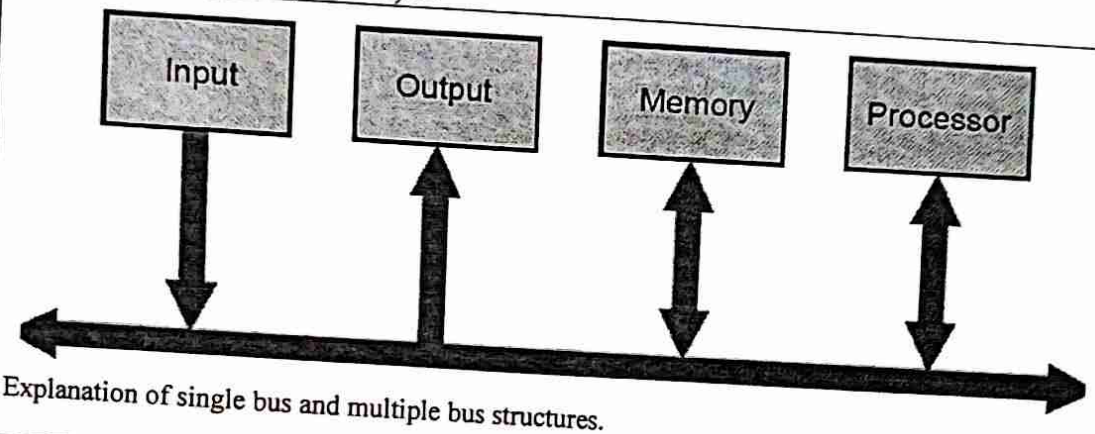
Set B

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE

Semester: III A & B
Course Code: 18EC35
Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Point	Marks
1(a)	 Explanation of Memory, MAR, MDR, PC, GPR's, ALU and command unit.	6
(b)	T – Processor time required to execute a program that has been prepared in high-level language N – Number of actual machine language instructions needed to complete the execution (note: loop) S – Average number of basic steps needed to execute one machine instruction. Each step completes in one clock cycle R – Clock rate The program execution time T is given by $T = (N \times S)/R$ Parameters, N, S & R are dependent on each other. To achieve high performance, the computer designer must reduce the value of T which means reducing N & S, increasing R. The value of N is reduced if the source program is compiled into fewer machine instructions. The value of S is reduced in instruction have a smaller number of basic steps to perform. The value of R can be increased by using a higher frequency clock. Care has to be taken while modifying values since changes in the parameter may affect the other	6
(c)	**In byte addressing**, successive addresses refer to successive byte locations in the memory. Byte locations have addresses 0,1,2,. So, a computer can access individual bytes stored in computer memory by using byte addresses.	

	Examples.	
2(a)	1 Personal Computers, 2.. Notebook Computers 3 Workstations 4 Enterprise Systems. 4a. Servers 4b. Supercomputers Explanation of above briefly.	6
(b)	(i) 85.125 in binary ;floating point representation – single precision 85.125 = $1010101.001_{(2)} = 1.010101001x2^6$, E = 6, so, E' = E+127 = 133 = 10000101 (8 bits) = 0(S(+/-) -1bit) 10000101 (E'-8 bits) 0101010010...0(M-23 bits) = +1.0101010010...0 (23 bits)x 2^6 (ii) In double precision, E' = E+1023 = 6+1023 = 1029 = 1000 0000 101(11 bits) = 0(+) 10000000101(E'-11 bits) 0101010010...0(M-52 bits) = +1.0101010010...0 (52 bits)x2^6	6
(c)	 Explanation of single bus and multiple bus structures.	6
3(a)		6

Name	Assembler syntax	Addressing function
Immediate	#Value	Operand = Value
Register	Ri	EA = Ri
Absolute (Direct)	LOC	EA = LOC
Indirect	(Ri)	EA = [Ri]
	(LOC)	EA = [LOC]

(b)

ASSEMBLER DIRECTIVES

- **Directives** are the assembler commands to the assembler concerning the program being assemb
- These commands are not translated into machine opcode in the object-program.

6

	Memory address label	Operation	Addressing or data information
Assembler directives	SUM	EQU	200
		ORIGIN	204
	N	DATAWORD	100
	NUM1	RESERVE	400
		ORIGIN	100
Statements that generate machine instructions	START	MOVE	N,R1
		MOVE	#NUM1,R2
		CLR	R0
	LOOP	ADD	(R2),R0
		ADD	#4,R2
		DEC	R1
		BGTZ	LOOP
		MOVE	R0,SUM
Assembler directives		RETURN	
		END	START

4(a)		6
(b)		6

Name	Assembler syntax	Addressing function
Index	X(Ri)	EA = [Ri] + X
Relative	X(PC)	EA = [PC] + X
Autoincrement	(Ri)+	EA = [Ri]; Increment Ri
Autodecrement	−(Ri)	Decrement Ri; EA = [Ri]

EA = effective address
Value = a signed number

```
                Move        #LIST,R0
                Clear       R1
                Clear       R2
                Clear       R3
                Move        N,R4
        LOOP    Add         4(R0),R1
                Add         8(R0),R2
                Add         12(R0),R3
                Add         #16,R0
                Decrement   R4
                Branch>0    LOOP
                Move        R1,SUM1
                Move        R2,SUM2
                Move        R3,SUM3
```

Course in charge

Module Coordinator

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109
II SESSIONAL TEST QUESTION PAPER 2020-21ODD SEMESTER

Set A

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration :90Minutes

Semester: III A& B
Course Code: 18EC35
Date:19-11-2020

Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Make use of** figures to explain the registers in keyboard and display interfaces	6	CO3	Applying-K3
(b)	**Identify** the differences between Subroutine and Interrupt Service Routine.	6	CO3	Applying-K3
(c)	**Make use of** a neat diagram to explain DMA controller	6	CO3	Applying-K3
	OR			
2(a)	**Identify** the methods of enabling and disabling interrupts & also define Interrupts	6	CO3	Applying-K3
(b)	**Identify** the differences between Daisy Chain scheme with the scheme with combination of Daisy Chain , individual request and acknowledge lines for handling multiple interrupt requests.	6	CO3	Applying-K3
(c)	**Develop** an Assembly language program to read an input line from the keyboard and store the characters in successive byte locations in the memory starting at location LINE	6	CO3	Applying-K3
	PART-B			
3(a)	**Identify** the differences between Stack and Queue with examples	6	CO2	Applying-K3
(b)	**Make use of** figure to explain memory access by the processor	6	CO4	Applying-K3
	OR			
4(a)	Make use of examples to explain different types of shift operations with examples.	6	CO2	Applying-K3
(b)	**Make use of** a neat diagram to Illustrate the organization of 1Kx1 memory chip	6	CO4	Applying-K3

Email answer scripts with file name format **USN_18EC35_IA2.pdf** to **18ec35ia1@gmail.com**

Course in charge

Module Coordinator

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

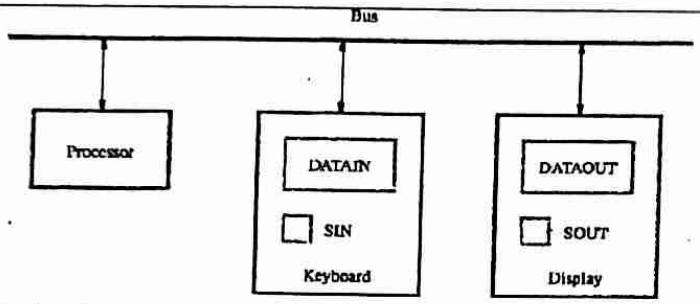
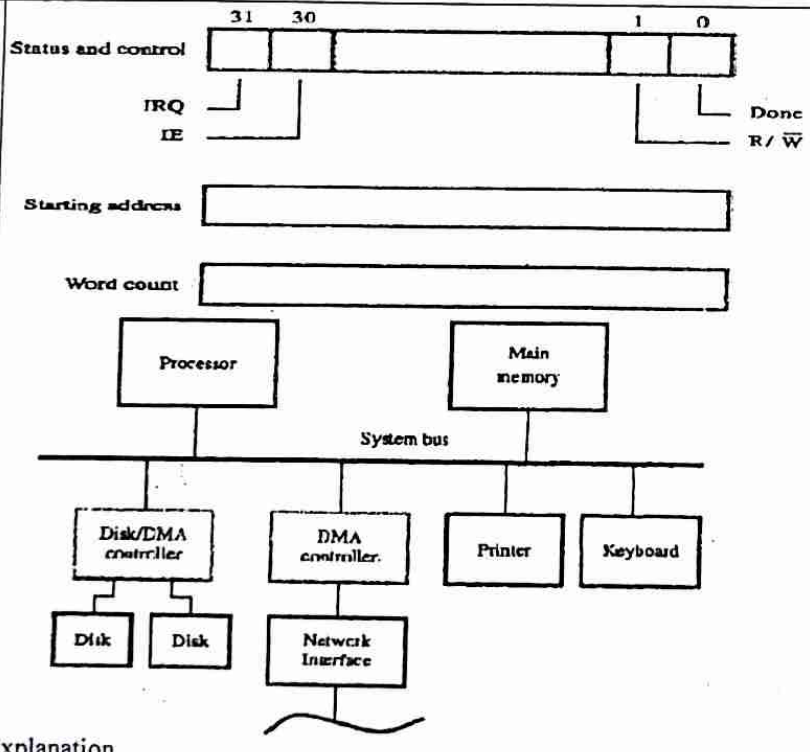
II SESSIONAL TEST SCHEME 2020-21ODD SEMESTER

Set A

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE

Semester: III A& B
CourseCode: 18EC35
Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Point	Marks
1(a)	 Explanation	3+3
(b)	<table><tr><th>Subroutine</th><th>Interrupt Service Routine</th></tr><tr><td>It is a portion of the code within a large program which performs a specific task.</td><td>ISR are to handle hardware interrupts.</td></tr><tr><td>It runs when we call it.</td><td>It runs whenever a certain signal occurs.</td></tr><tr><td>We know where the subroutine runs because we call it.</td><td>We don't know when ISR will be executed.</td></tr></table>	4 2
(c)	 Explanation	2+2+2

2(a)

The first possibility is to have the processor hardware ignore the interrupt request line until the execution of the first instruction of the ISR has been completed. The programmer can ensure that no further interruptions will occur by using interrupt-disable instruction as the first instruction of the ISR. The interrupt-enable instruction is the last instruction of the ISR before the Return from interrupt instruction.

In second option (for processor with only one interrupt request line), the processor can automatically disable interrupts before starting the execution of the ISR. One bit of program status register (PS), called interrupt enable indicates whether interrupts are enabled. After saving the contents of PS on stack, the processor clears the interrupt enable bit in its PS register thus disabling further interrupts. When the Return-from-interrupts instruction is executed, the contents of PS are restored from the stack setting the interrupt-enable bit back to 1. 5+1

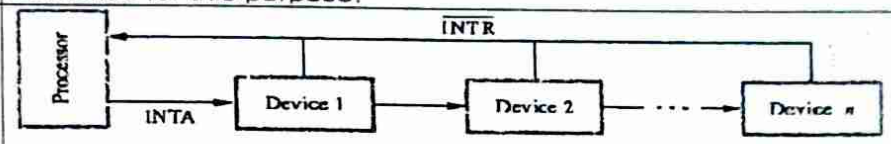
In the third option, the processor has a special interrupt request line for which the interrupt handling circuit responds only to the leading edge of the signal. Such a line is said to be *edge triggered*. In this case, the processor will receive only one request. Hence no danger of multiple interruptions.

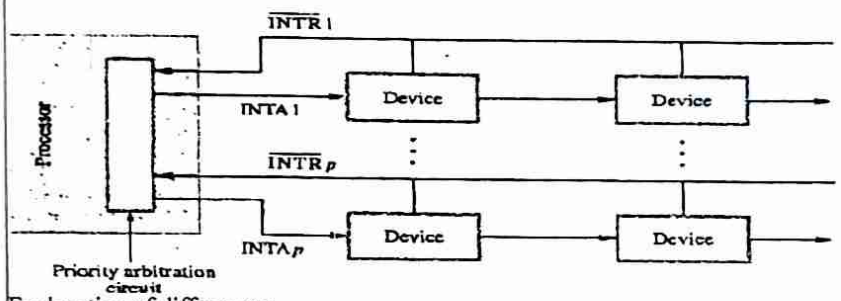
Definiition

Sometimes the processor time will be wasted while waiting for an I/O device to become ready.

This time can be eliminated by I/O device by sending a hardware signal called an *interrupt* to the processor. A bus control line called *interrupt signal line* is dedicated for this purpose.

(b)





2+2+2

Explanation of differences

(c)

Main Program

```
        Move        #LINE,PNTR      Initialize buffer pointer.
        Clear       EOL             Clear end-of-line indicator.
        BitSet      #2,CONTROL      Enable keyboard interrupts.
        BitSet      #9,PS           Set interrupt-enable bit in the PS.
        :
```

2

Interrupt-service routine

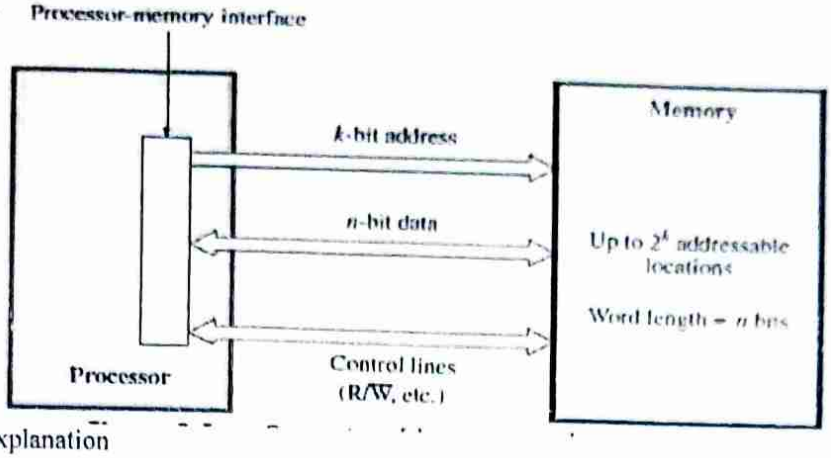
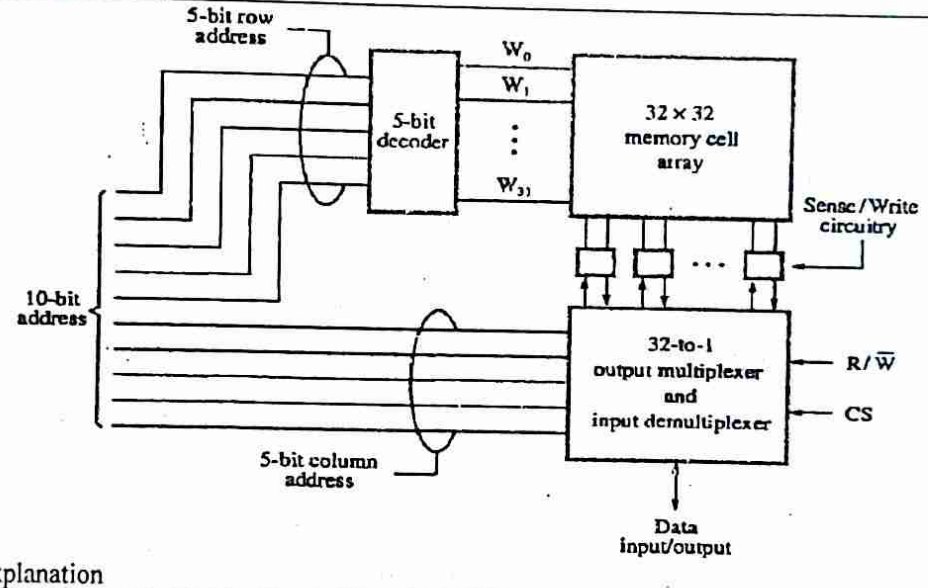
```
READ    MoveMultiple    R0–R1,–(SP)     Save registers R0 and R1 on stack.
        Move            PNTR,R0         Load address pointer.
        MoveByte        DATAIN,R1       Get input character and
        MoveByte        R1,(R0)+          store it in memory.
        Move            R0,PNTR         Update pointer.
        CompareByte     #$0D,R1         Check if Carriage Return.
        Branch≠0        RTRN
        Move            #1,EOL          Indicate end of line.
        BitClear        #2,CONTROL      Disable keyboard interrupts.
RTRN    MoveMultiple    (SP)+,R0–R1     Restore registers R0 and R1.
        Return-from-interrupt
```

4

3(a)

Stack	Queue
One end is fixed while other end rises and falls as data are pushed and popped.	Both ends of queues move to higher addresses as data is pushed and popped.
A single pointer is needed to point to the top of the stack	Two pointers are needed to keep track of two ends of the queue.
In stack care must be taken to see to it that nothing is pushed into a stack which is full and nothing is popped out from the stack which is empty.	Queue would continuously move through the memory of the computer in the direction of higher addresses. This can be limited by using a *circular buffer*. Let us assume that memory addresses from BEGINNING to END are assigned to the queue. The first entry in the queue is entered into BEGINNING and successive entries are appended at higher addresses. By the time queue reaches END, space would be created in the beginning. So back pointer is reset to BEGINNING and the process continues.

6

(b)	 Explanation	4
4(a)	Examples for following shift operations Logical shift left Logical shift right Arithmetic shift left Arithmeticshift right	2 1.5x4
(b)	 Explanation	3+3

Course in charge

Module Coordinator

HOD
E&C

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

II SESSIONAL TEST QUESTION PAPER 2020-21ODD SEMESTER

Set B

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration :90Minutes

Semester: III A & B
Course Code: 18EC35
Date: 19-11-2020

Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Make use of** figures to explain the registers in keyboard and display interfaces	6	CO3	Applying-K3
(b)	**Identify** the differences between Subroutine and Interrupt Service Routine.	6	CO3	Applying-K3
(c)	.**Make use of** a neat diagram to explain DMA controller	6	CO3	Applying-K3
	OR			
2(a)	**Identify** the methods of enabling and disabling interrupts & also define Interrupts	6	CO3	Applying-K3
(b)	**Identify** the differences between Daisy Chain scheme with the scheme having combination of Daisy Chain, individual request and acknowledge lines for handling multiple interrupt requests.	6	CO3	Applying-K3
(c)	**Develop** an Assembly language program to read an input line from the keyboard and store the characters in successive byte locations in the memory starting at location LINE	6	CO3	Applying-K3
	PART-B			
3(a)	). Registers R1 and R2 of a computer contain the decimal values 1200 and 4600. **Obtain** the effective address of the memory operand in each of the following instructions (a) Load 20(R1),R5 (b) Move #3000,R5 (c) Store R5,30(R1,R2) (d) Add -(R2),R5 (e) Subtract (R1)+,R5	6	CO2	Applying-K3
(b)	**Make use of** a neat diagram to Illustrate the organization of 1Kx1 memory chip.	6	CO4	Applying-K3
	OR			
4(a)	Make use of examples to explain different types of rotate operations with examples.	6	CO2	Applying-K3
(b)	**Make use of** figure to explain memory access by the processor.	6	CO4	Applying-K3

Email answer scripts with file name format USN_18EC35_IA1.pdf to 18ec35ia1@gmail.com

Course in charge

Module Coordinator

HOD

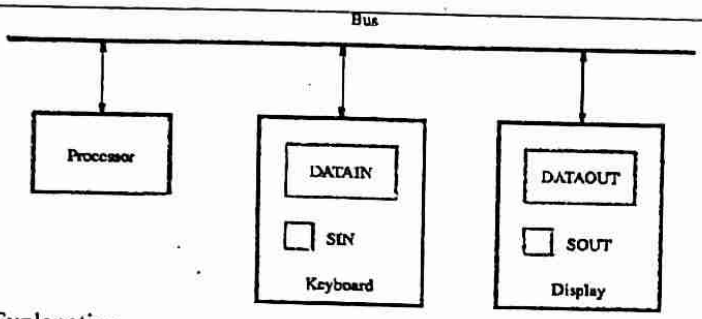
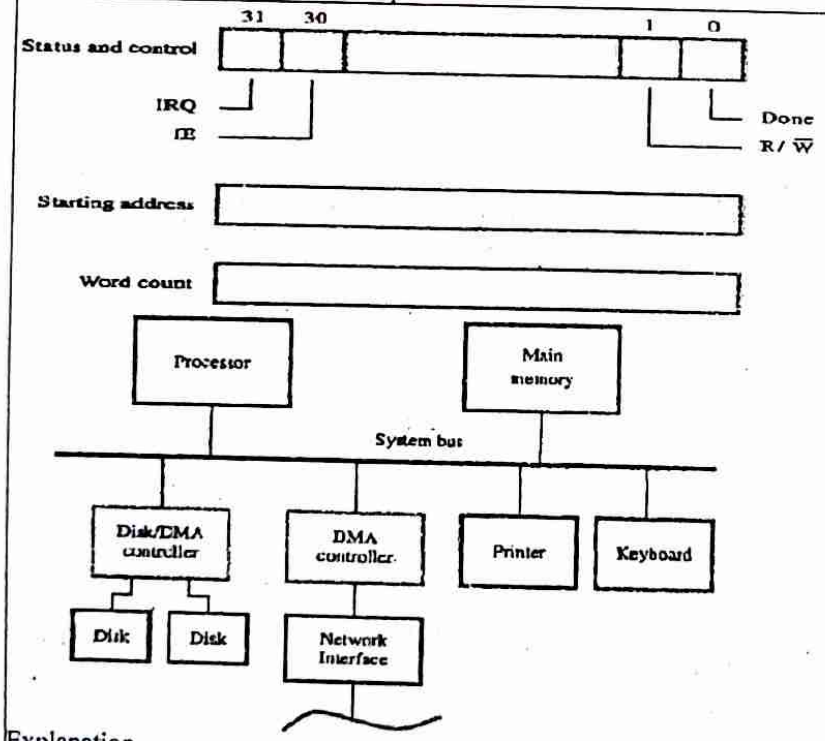
K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109
II SESSIONAL TEST SCHEME 2020-21ODD SEMESTER

Set B

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE

Semester: III A& B
CourseCode: 18EC35
Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Point	Marks
1(a)	 Explanation	3+3
(b)	<table><tr><th>Subroutine</th><th>Interrupt Service Routine</th></tr><tr><td>It is a portion of the code within a large program which performs a specific task.</td><td>ISR are to handle hardware interrupts.</td></tr><tr><td>It runs when we call it.</td><td>It runs whenever a certain signal occurs.</td></tr><tr><td>We know where the subroutine runs because we call it.</td><td>We don't know when ISR will be executed.</td></tr></table> Mainly above three points with explanation	4 2
(c)	 Explanation	2+2+2

2(a)

The first possibility is to have the processor hardware ignore the interrupt request line until the execution of the first instruction of the ISR has been completed. The programmer can ensure that no further interruptions will occur by using interrupt-disable instruction as the first instruction of the ISR. The interrupt-enable instruction is the last instruction of the ISR before the Return from interrupt instruction.

In second option (for processor with only one interrupt request line), the processor can automatically disable interrupts before starting the execution of the ISR. One bit of program status register (PS), called interrupt enable indicates whether interrupts are enabled. After saving the contents of PS on stack, the processor clears the interrupt enable bit in its PS register thus disabling further interrupts. When the Return-from-interrupts instruction is executed, the contents of PS are restored from the stack setting the interrupt-enable bit back to 1.

In the third option, the processor has a special interrupt request line for which the interrupt handling circuit responds only to the leading edge of the signal. Such a line is said to be *edge triggered*. In this case, the processor will receive only one request. Hence no danger of multiple interruptions.

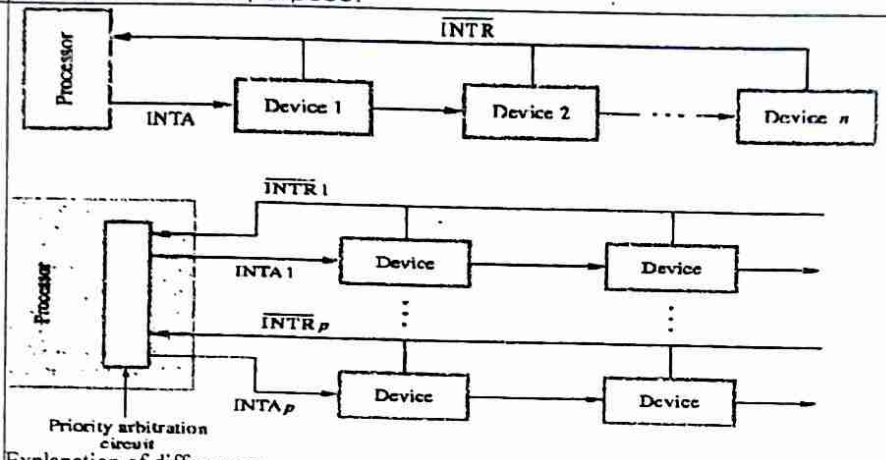
5+1

Definiition

Sometimes the processor time will be wasted while waiting for an I/O device to become ready.

This time can be eliminated by I/O device by sending a hardware signal called an *interrupt* to the processor. A bus control line called *interrupt signal line* is dedicated for this purpose.

(b)



2+2+2

Explanation of differences

(c)

Main Program

```
        Move        #LINE,PNTR      Initialize buffer pointer.
        Clear       EOL             Clear end-of-line indicator.
        BitSet      #2,CONTROL      Enable keyboard interrupts.
        BitSet      #9,PS           Set interrupt-enable bit in the PS.
        ⋮
```

2

Interrupt-service routine

```
READ    MoveMultiple    R0–R1,–(SP)     Save registers R0 and R1 on stack.
        Move            PNTR,R0         Load address pointer.
        MoveByte        DATAIN,R1       Get input character and
        MoveByte        R1,(R0)+          store it in memory.
        Move            R0,PNTR         Update pointer.
        CompareByte     #$0D,R1         Check if Carriage Return.
        Branch≠0        RTRN
        Move            #1,EOL          Indicate end of line.
        BitClear        #2,CONTROL      Disable keyboard interrupts.
RTRN    MoveMultiple    (SP)+,R0–R1     Restore registers R0 and R1.
        Return-from-interrupt
```

4

3(a)

LOAD 20(R1), R5 E A = 20+[R1] = 20+1200 = 1220

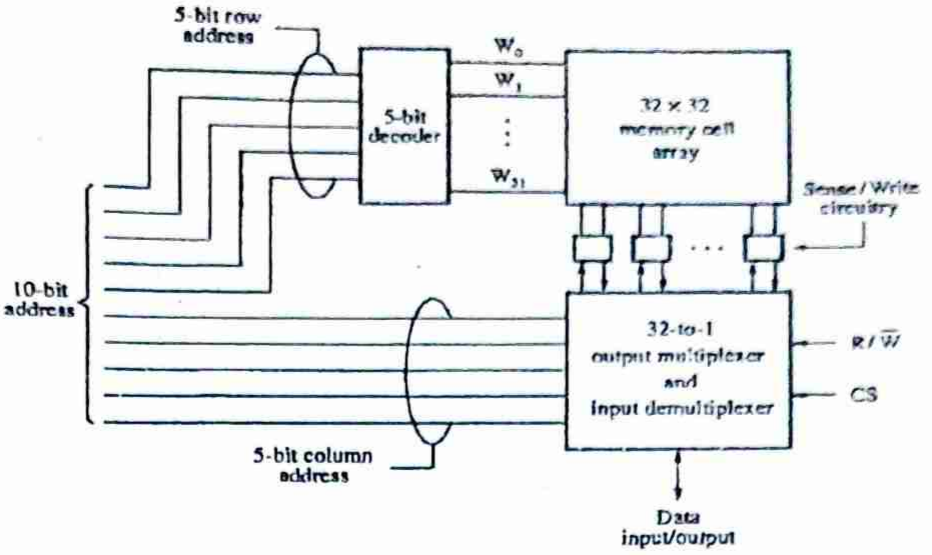
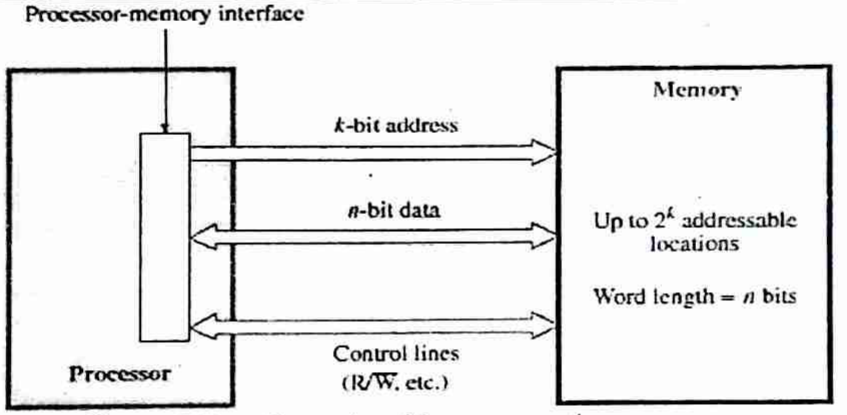
MOVE #3000, R1 E A = 3000

STORE R5, 30(R1,R2) E A = 30+{R1]+[R2] = 30+1200+4600 = 5830

ADD -(R2), R5 EA = 4600-1 = 4599

SUBTRACT (R1)+, R5 E A = 1200

6

(b)	 Explanation	4 2
4(a)	Examples for following Rotate operations Rotate left Rotate right Rotate left through carry Rotate rightthrough carry	1.5x4
(b)	 Explanation	3+3

Course in charge

Module Coordinator

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

III SESSIONAL TEST QUESTION PAPER 2020-21ODD SEMESTER

Set A

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration :90Minutes

Semester: III A& B
CourseCode: 18EC35
Date: 8-1-2021

Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Make use of** figure & explain the single bus organization of data path inside a processor	6	CO5	Applying-K3
(b)	**Make use of** figure & explain the implementation of 1 bit register.	6	CO5	Applying-K3
(c)	**Construct** the sequence of operations and timing diagram for the instruction MOV (R1), R2	6	CO5	Applying-K3
	OR			
2(a)	**Make use of** figure & explain multiple bus organization of CPU.	6	CO5	Applying-K3
(b)	**Make use of** figure & explain the organization of control unit for unconditional branch instruction	6	CO5	Applying-K3
(c)	**Make use of** figure & explain Hardwired Control Unit Organization in a processing unit.	6	CO5	Applying-K3
	PART-B			
3(a)	**Make use of** figure & explain the principle of working of magnetic disk.	6	CO4	Applying-K3
(b)	**Identify and** explain RAID levels of RAID Disk Arrays	6	CO4	Applying-K3
	OR			
4(a)	**Make use of** a neat diagram & explain the working of CMOS memory cell	6	CO4	Applying-K3
(b)	**Analyze** different types of ROMs..	6	CO4	Analyze-K4

Email answer scripts with file name format **USN_18EC35_IA3.pdf** to **18ec35ia1@gmail.com**

Course in charge

Module Coordinator

HOD

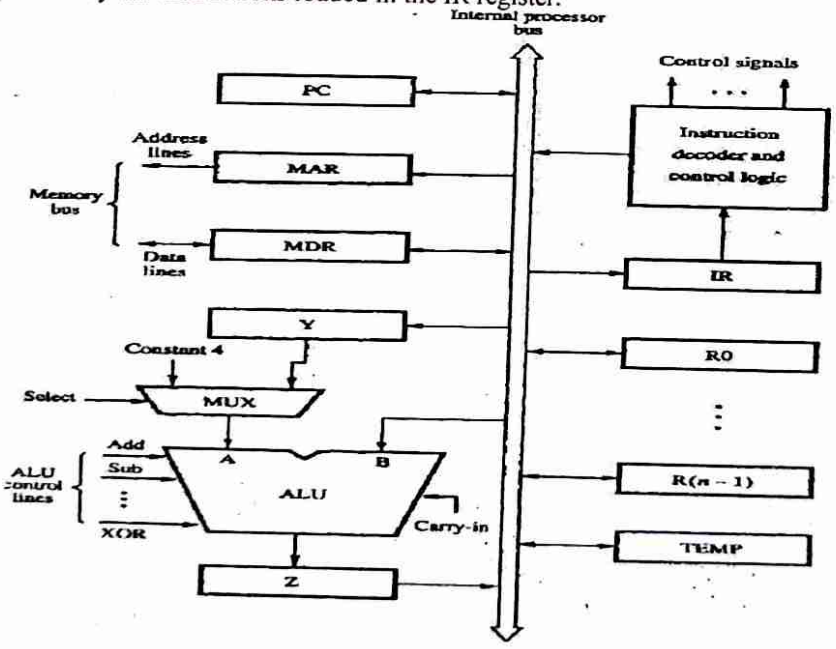
K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

III SESSIONAL TEST SCHEME 2020-21ODD SEMESTER

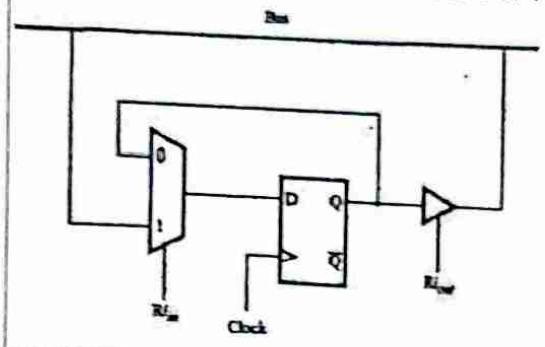
Set A

Degree	: B.E	Semester:	III A& B
Branch	: E &CE	CourseCode:	18EC35
Course Title	: COMPUTER ORGANIZATION & ARCHITECTURE	Max. Marks:	30

Note: Answer ONE full question from each part.

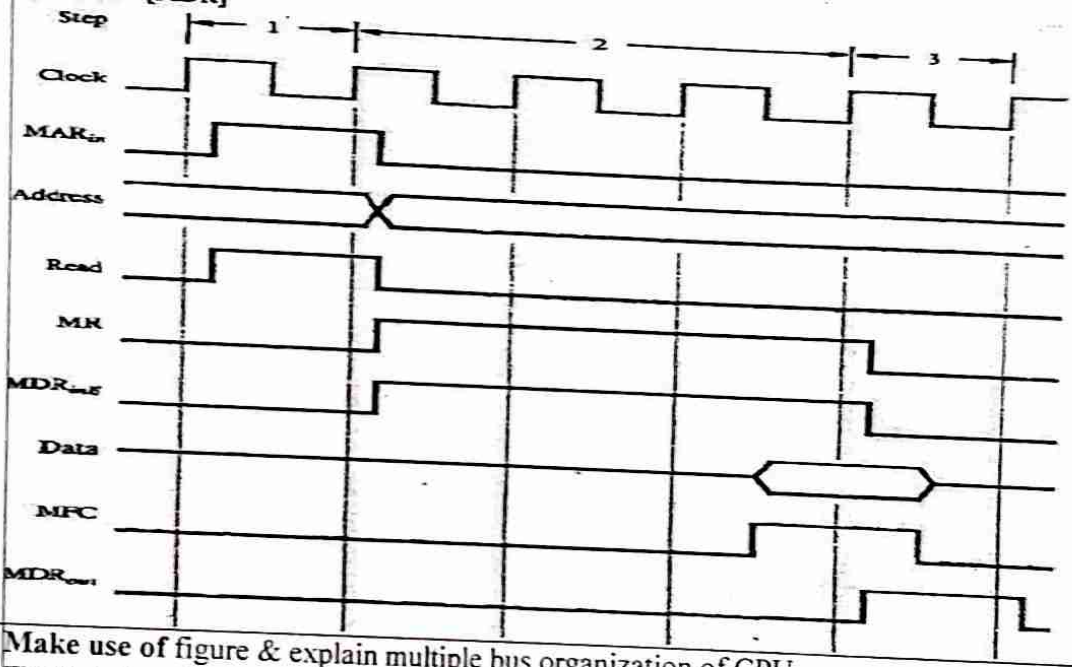
Q No.	Point	Marks
1(a)	**Make use of** figure & explain the single bus organization of data path inside a processor. The figure below shows single bus organization of the data path inside a processor. □The data and address lines of external memory bus are connected to processor via MDR and MAR respectively. □MDR has two inputs and two outputs. It can receive inputs as well as send outputs on memory bus or internal processor bus. □The input of MAR is internal bus and output is external bus. □The control lines of the memory bus are connected to the instruction decoder and control logic block. This unit is responsible for issuing signals that control operations of all units. □The registers R0 to R(n-1) are general purpose registers. Their numbers vary from one processor to another processor. □There are some special purpose registers like index register and stack pointer. The registers Y, Z and Temp in the figure are used by processor for temporary storage during execution of some instruction. They are never used for storing data generated by one instruction for later use by another instruction. □The Multiplexer selects either the output of register Y or the constant value 4 to provide as input to the ALU. Constant 4 is used to increment the contents of program counter. □The instruction decoder and control logic unit is responsible for implementing the actions specified by the instructions loaded in the IR register.  Figure 7.1 Single-bus organization of the datapath inside a processor.	3+3
(b)	**Make use of** figure & explain the implementation of 1 bit register. The implementation of one bit register Ri is shown in figure below. o A two input multiplexer is used to select the data applied to the input of edge triggered D flipflop.	4+2

- The Q output of flip flop is connected to the bus via a tristate gate.
- When Riout is 0, gate output is in high impedance state (electrically disconnected).
- When Riout is 1, gate drives the bus to 0 or 1 depending on the value of Q



(c) **Construct** the sequence of operations and timing diagram for the instruction MOV (R1), R2. 3+3

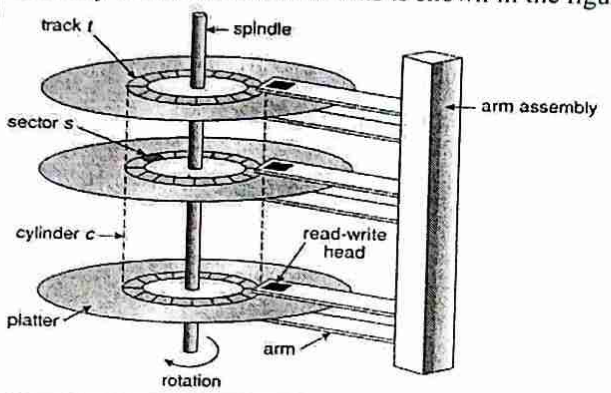
1. MAR ← [R1]
2. Start a Read operation on the memory bus
3. Wait for the MFC response from the memory
4. Load MDR from the memory bus
5. R2 ← [MDR]

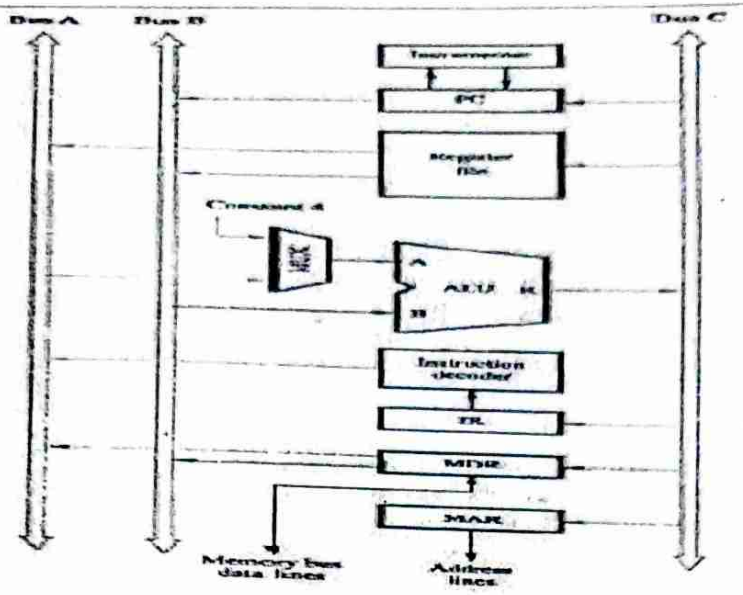


2(a) **Make use of** figure & explain multiple bus organization of CPU. 3+3

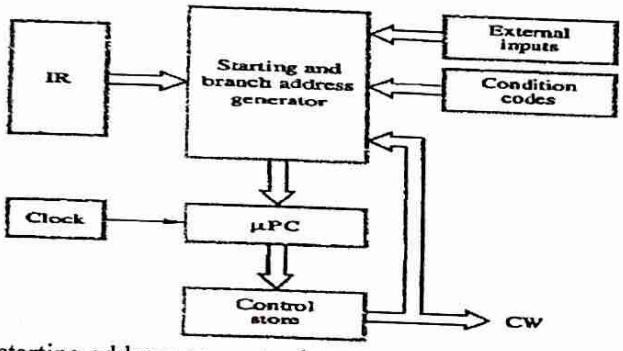
The fig below shows three bus structure used to connect registers and ALU.

- All general purpose registers are combined into a single block called **register file.**
- The register file has 3 ports. The two output ports allow the contents of two different registers to be placed on bus A and B. The input port allows the contents of bus C to be placed into a third register during the same clock cycle.
- Buses A and B are normally used for passing input operands to ALU. Bus C passes the result to the destination. This arrangement avoids the need for registers Y and Z. If needed ALU may simple pass one of its two operands unmodified to bus C by using control signals R=A or R=B.
- The second feature in the figure below is Incrementer unit which is used to increment PC by 4. This eliminates the need to add 4 to PC every time using the main ALU.

	□The step decoder provides a separate signal line for each step/time slot. □The output of Instruction Decoder consists of separate lines for each machine instruction. □Depending on the instruction any one of INS1 to INSm is set to 1 and all other lines are setto 0. □The input signals to the encoder block are combined to generate individual control signals Yin, PCout, Add, End and so on.	
3(a)	**Make use of** figure & explain the principle of working of magnetic disk. Hard disk stores information in the form of magnetic fields. Data is stored digitally in the form of tiny magnetized regions on the platter where each region represents a bit. To write a data on the hard disk, a magnetic field is placed on the tiny field in one of these two polarities: N-S – If North Pole arrives before the south pole and S-N – if the south pole arrives before the north pole while the field is accessed. An orientation in the one direction (like N-S) can represent the '1' while the opposite orientation (S-N) represents "0". This polarity is sensed by integrated controllers built within the hard disk. 1. It consists of one or more disks mounted on a common spindle. 2. A thin magnetic film is deposited on either side of each disk. 3. The disks are placed in a rotary drive so that the magnetized surfaces move in close Proximity to read/write heads. This is shown in the figures below.  	3+3
(b)	**Identify and** explain RAID levels of RAID Disk Arrays. RAID stands for **Redundant Array of Inexpensive Disks**. It is a storage system based on multiple disks proposed by University of California-Berkeley. □It has six different configurations known as RAID levels. □**RAID 0** is the basic configuration to enhance performance o In RAID 0 configuration a single large file is broken up into smaller pieces and are stored on different disks. This is called as **data striping**. o All disks can deliver their data in parallel. o The total transfer time of the file is equal to the transfer time that would be required in a single disk system divided by the number of disks used in the array. o As each disk operates independently, access times vary and buffering of accessed pieces of data is needed so that the complete file can be reassembled. □**RAID 1** is intended to provide better reliability by storing identical copies of data on two disks rather than just one. If one disk drive fails, the other one can be used. This is however costly because of duplication. □**RAID 2**, RAID 3, RAID 4 levels achieve increased reliability through various parity checking schemes without duplication. □**RAID 5** also has parity based error recovery schemes but this information is distributed among all schemes. □Some hybrid arrangements are also available which are the combination of some of the above mentioned configurations.	6
4(a)	**Make use of** a neat diagram & explain the working of CMOS memory cell.	3+3



(b) **Make use of** figure & explain the organization of control unit for cconditional branch instruction



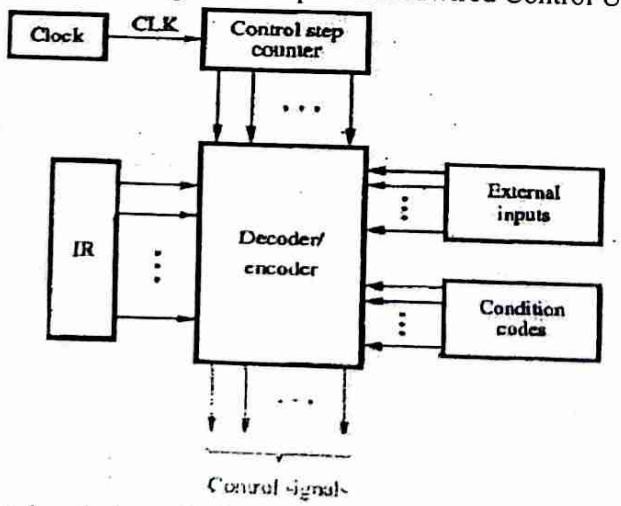
3+3

The starting address generator becomes the starting and the branch address generator. This block loads a new address into the μPC when a microinstruction instructs it to do so. The inputs to this block consists of the external inputs and condition codes as well as contents of IR.

□The μPC is incremented every time a new microinstruction is fetched from the microprogram memory except

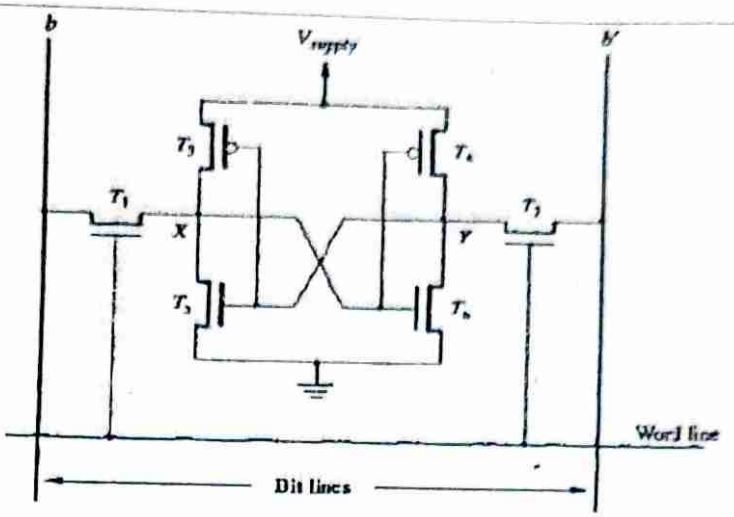
- o When new instruction is loaded into IR, its starting address is loaded into μPC.
- o During branch microinstructions, μPC is loaded with branch address.
- o When End microinstruction is encountered, μPC is loaded with the address of first CW in the micro routine.

(c) **Make use of** figure & explain Hardwired Control Unit Organization in a processing unit.



3+3

The decoder/encoder block in the above figure is a combinational circuit that generates the required control outputs depending on the state of all its inputs. A more detailed block diagram by separating decoder and encoder blocks is shown in the fig. below.



Explanation

(b) **Analyze** different types of ROMs. — 6

ROM – Read Only Memory-Programmed during manufacturing

PROM- Programmable Read Only Memory-Programmed by burning fuses by user only once

EPROM – Erasable Programmable Read Only Memory-can be Programmed by inducing charge and erased by UV light by user of many times

A disadvantage of EPROMs is that the chip must be physically removed from the circuit for reprogramming and erasing. Hence Electrically Erasable PROMs (EEPROMs) are used.

EEPROM – Electrically Erasable Programmable Read Only Memory

- In EEPROMs the chips need not be removed and it is possible to erase the cell contents selectively.
- The only disadvantage of EEPROMs is that different voltages are needed for erasing, writing and reading the stored data

Course in charge **Module Coordinator** **HOD**

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

III SESSIONAL TEST QUESTION PAPER 2020-21ODD SEMESTER

Set B

USN	1	K	S			E	C			

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE
Duration :90Minutes

Semester: III A& B
CourseCode: 18EC35
Date: 8-1-2021

Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Question	Marks	CO map ping	K-Level
	PART-A			
1(a)	**Construct** control sequence for execution of the instruction Add (R3), R1	6	CO5	Applying-K3
(b)	**Make use of** figure & explain the connection and control signals for register MDR.	6	CO5	Applying-K3
(c)	**Construct** the control sequence for an unconditional branch instruction.	6	CO5	Applying-K3
	OR			
2(a)	**Construct** the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization.	6	CO5	Applying-K3
(b)	**Make use of** figure & explain the generation of **End** control signal given by $End = T_7 \cdot ADD + T_5 \cdot BR + (T_5 \cdot N + T_4 \cdot \overline{N}) \cdot BRN$	6	CO5	Applying-K3
(c)	**Make use of** figure & explain Basic organization of a microprogrammed control Unit.	6	CO5	Applying-K3
	PART-B			
3(a)	**Make use of** figure & explain the organization and accessing of data on a DISK.	6	CO4	Applying-K3
(b)	**Make use** of a figure and explain the organization of Virtual memory.	6	CO4	Applying-K3
	OR			
4(a)	**Make use of**figure and explain Cache Memory.	6	CO4	Applying-K3
(b)	**Identify** the differences between **flash** drives and Hard disk drives.	6	CO4	Applying-K3

Email answer scripts with file name format **USN_18EC35_IA3.pdf** to**18ec35ia1@gmail.com**

Course in charge

Module Coordinator

HOD

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

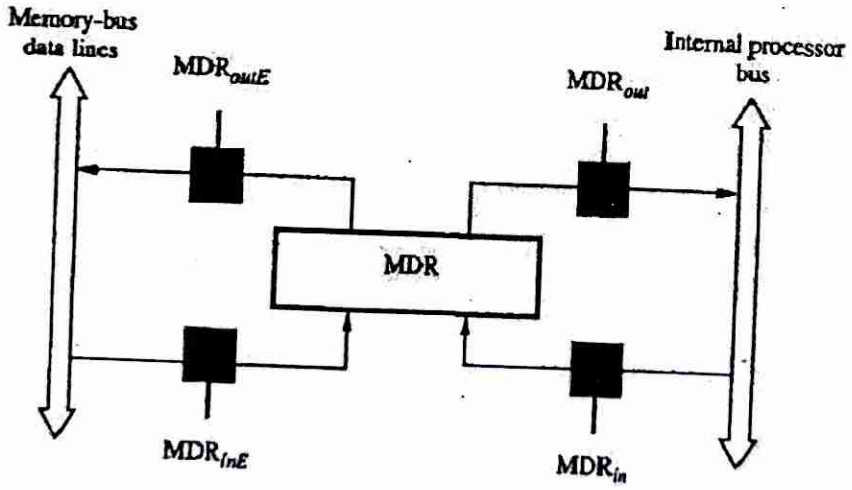
III SESSIONAL TEST SCHEME 2020-21ODD SEMESTER

Set B

Degree : B.E
Branch : E &CE
Course Title : COMPUTER ORGANIZATION & ARCHITECTURE

Semester: III A& B
CourseCode: 18EC35
Max. Marks: 30

Note: Answer ONE full question from each part.

Q No.	Point	Marks
1(a)	Construct the control sequence for execution of the instruction Add (R3), R1 **Step Action** 1 PC_{out}, MAR_{in}, Read, Select4, Add, Z_{in} 2 Z_{out}, PC_{in}, Y_{in}, WMFC 3 MDR_{out}, IR_{in} 4 $R3_{out}$, MAR_{in}, Read 5 $R1_{out}$, Y_{in}, WMFC 6 MDR_{out}, SelectY, Add, Z_{in} 7 Z_{out}, $R1_{in}$, End **Figure 7.6** Control sequence for execution of the instruction Add (R3),R1. Explanation	3+3
(b)	Make use of figure & explain the connection and control signals for register MDR.  **Figure 7.4** Connection and control signals for register MDR. Explanation	3+3
(c)	**Construct** the control sequence for an unconditional branch instruction.	3+3

Step	Action
1	PC_{out}, MAR_{in}, Read, Select4, Add, Z_{in}
2	Z_{out}, PC_{in}, Y_{in}, WMFC
3	MDR_{out}, IR_{in}
4	Offset-field-of-IR_{out}, Add, Z_{in}
5	Z_{out}, PC_{in}, End

Figure 7.7 Control sequence for an unconditional Branch instruction.

Explanation

2(a) Construct the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization.

Step	Action
1	PC_{out}, R=B, MAR_{in}, Read, IncPC
2	WMFC
3	MDR_{outB}, R=B, IR_{in}
4	$R4_{outA}$, $R5_{outB}$, SelectA, Add, $R6_{in}$, End

3+3

Figure 7.9 Control sequence for the instruction Add R4,R5,R6 for the three-bus organization in Figure 7.8.

Explanation

(b) Make use of figure & explain the generation of End control signal given by

$$End = T_7 \cdot ADD + T_5 \cdot BR + (T_5 \cdot N + T_4 \cdot \overline{N}) \cdot BRN + \cdots \quad [7.2]$$

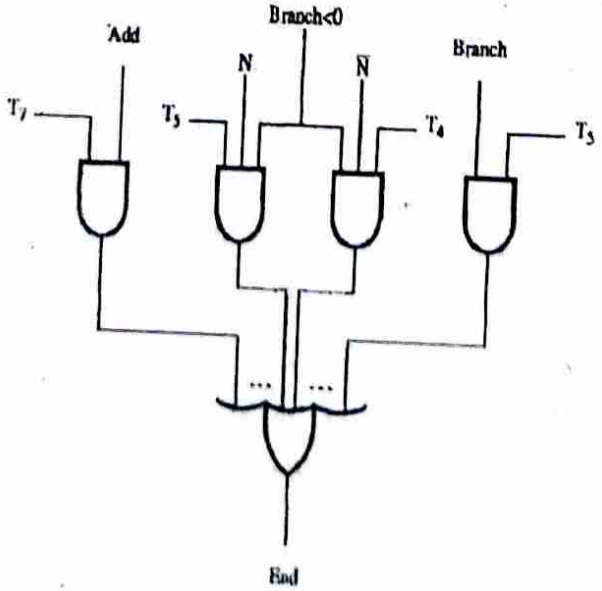
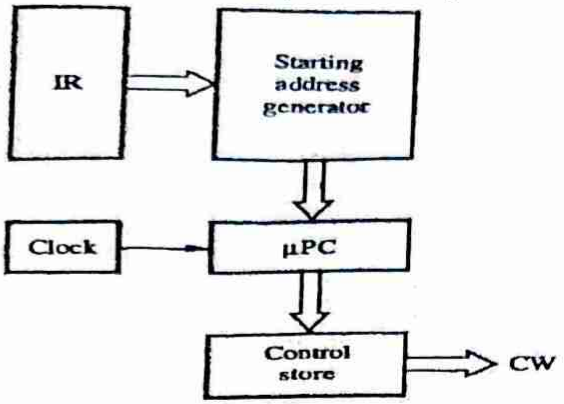
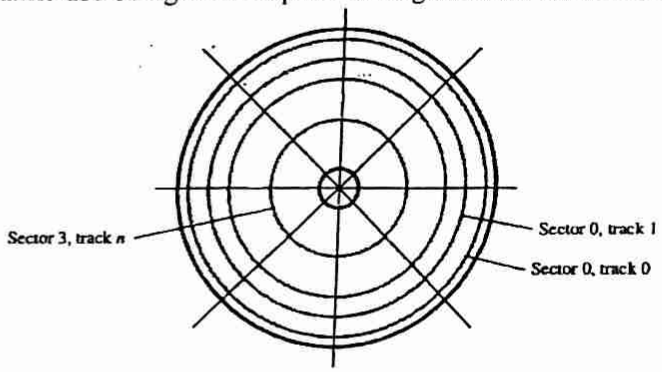
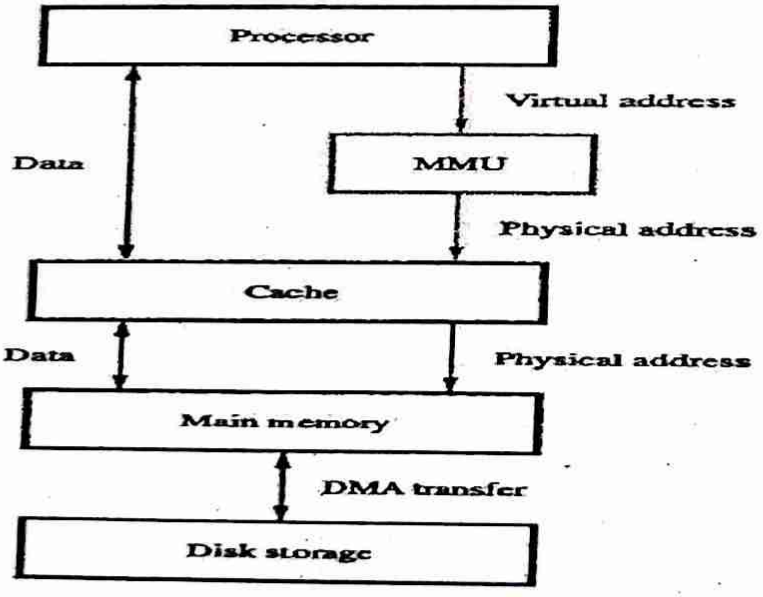


Figure 7.13 Generation of the End control signal.

Explanation and meaning of T1, T2 ..

3+3

(c)	**Make use of** figure & explain Basic organization of a mictoprogrammed control Unit.  **Figure 7.16** Basic organization of a microprogrammed control unit. Explanation	3+3
3(a)	**Make use of** figure & explain the organization and accessing of data on a DISK.  **Figure 5.30** Organization of one surface of a disk. Explanation	3+3
(b)	**Make use** of a figure and explain the organization of Virtual memory.  **Figure 5.26** Virtual memory organization. Explanation	3+3
4(a)	**Make use of** figure and explain Cache Memory.	3+3

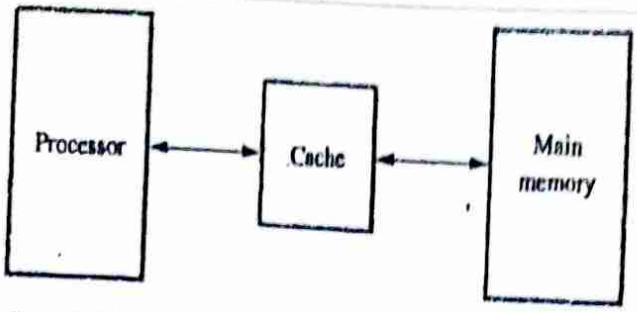


Figure 5.14 Use of a cache memory.

Explanation

(b) **Compare** flash drives and Hard disk drives. — 6

Flash Drives	Hard Disk Drives
They are solid state electronic devices	They are magnetic devices
They don't have movable parts and hence insensitive to vibration.	They have movable parts and hence sensitive to vibrations
They have shorter seek and access times which results in faster response.	They have comparatively larger seek and access times.
They have lower power consumption and hence suitable for battery driven applications	They have comparatively large power consumption.
They are available in smaller storage capacity.	Storage capacity is larger
Higher cost per bit	Lower cost per bit.
Deterioration rate is high	Deterioration rate is low

Course in charge **Module Coordinator** **HOD**

III Semester COA Assignment & Internal Marks
2020-21

Sl No	USN	Name	A1	A2	A3	A	IA1	IA2	IA3	IA Average	Total
1	1KS19EC001	ABHILASH A S	10	10	6	9	22	26	14	21	30
2	1KS19EC002	ABHISHEK CHANDRESH	5	10	10	9	30	30	9	23	32
3	1KS19EC003	AISHWARYA BASAVARAJ KEMBAVI	10	10	10	10	30	29	21	27	37
4	1KS19EC004	AISHWARYA M G	10	10	10	10	30	27	16	25	35
5	1KS19EC005	AKSHAY KUMAR D	10	10	10	10	29	26	12	23	33
6	1KS19EC006	AKSHITHA	10	10	10	10	26	30	18	25	35
7	1KS19EC007	AMRUTA	10	10	10	10	30	29	14	25	35
8	1KS19EC008	AMULYA R	10	10	10	10	27	30	21	26	36
9	1KS19EC009	ANITHA S	10	10	10	10	28	29	30	29	39
10	1KS19EC010	ANJALI Y J	10	10	6	9	30	28	13	24	33
11	1KS19EC011	ARCHANA YADAV M	10	10	6	9	30	29	13	24	33
12	1KS19EC012	ASHRITHA R	10	8	10	10	29	27	8	22	32
13	1KS19EC013	BHARATH KUMAR R	6	0	0	2	21	28	0	17	19
14	1KS19EC014	BHAVANA S	10	10	8	10	29	27	23	27	37
15	1KS19EC015	CHAITRA P	10	10	6	9	29	26	21	26	35
16	1KS19EC016	CHANDAN RAJ Y	10	10	10	10	29	26	22	26	36
17	1KS19EC017	CHANDANA.L	10	10	6	9	30	24	17	24	33
18	1KS19EC018	CHENNREDDY RAJASEKHAR	10	10	7	9	22	26	29	26	35
19	1KS19EC019	CHIRANTHANA YOGANANDA K	10	10	8	10	18	24	5	16	26
20	1KS19EC020	D NAYAN	9	10	6	9	26	23	18	23	32
21	1KS19EC021	DANESH RAJU V	10	10	8	10	29	25	20	25	35
22	1KS19EC022	DAVINO JOSEPH	10	10	10	10	25	23	12	20	30
23	1KS19EC023	DHANYA SUKANTH B K	10	10	10	10	30	26	18	25	35
24	1KS19EC024	DHEEMANTH K N	10	8	8	9	29	30	10	23	32
25	1KS19EC025	DISHA SHIVANI	10	8	10	10	29	30	23	28	38
26	1KS19EC026	ERAM FATHIMA	9	6	10	9	30	29	0	20	29
27	1KS19EC027	GAYATHRI P K	10	10	6	9	29	30	24	28	37
28	1KS19EC028	GAYATHRI R WARRIER	10	10	10	10	29	30	26	29	39
29	1KS19EC029	GONUGUNTLA SAI SIDDARTHA	9	10	8	9	27	23	11	21	30

30	1KS19EC030	GOWRI S NADIGER	10	10	10	10	30	24	15	23	33
31	1KS19EC031	HARSHA R	8	6	8	8	22	23	0	15	23
32	1KS19EC032	HARSHITHA B Y	10	10	8	10	27	26	18	24	34
33	1KS19EC033	HEMANTH.R.PATIL	10	7	6	8	29	27	15	24	32
34	1KS19EC034	HIMA SWETHA S	9	10	6	9	30	28	AB	20	29
35	1KS19EC035	JAGRUTI PAI	10	10	10	10	30	30	27	29	39
36	1KS19EC036	JAYANTH M B	6	10	8	8	29	29	11	23	31
37	1KS19EC037	KAMMA MANUBOLU MANOGNA	9	10	9	10	27	29	27	28	38
38	1KS19EC038	KARTHIK K	9	10	10	10	22	22	9	18	28
39	1KS19EC039	KASHYAP.P	10	8	6	8	25	28	11	22	30
40	1KS19EC040	KRUPA.A	10	10	6	9	30	28	22	27	36
41	1KS19EC041	KRUTHI K S	10	10	10	10	30	28	25	28	38
42	1KS19EC042	LAKSHMAN KUMARA B	10	6	6	8	29	28	10	23	31
43	1KS19EC043	LIKITHA.H	10	10	8	10	30	29	8	23	33
44	1KS19EC044	M LOKESHWARI	10	10	8	10	27	27	19	25	35
45	1KS19EC045	MANU N KANDRA	10	10	10	10	27	28	21	26	36
46	1KS19EC046	MEGHANA H P	10	10	8	10	27	30	17	25	35
47	1KS19EC047	MOHAMMAD RAKHEEB M	5	5	5	5	28	20	8	19	24
48	1KS19EC048	MOHITH KUMAR G	8	10	10	10	27	25	4	19	29
49	1KS19EC049	MONIKA V ARYA	10	10	6	9	30	28	12	24	33
50	1KS19EC050	MONISHA.B.K	10	10	8	10	30	30	20	27	37
51	1KS19EC051	N ANILA	10	10	8	10	30	28	16	25	35
52	1KS19EC052	NIDHI S	10	7	6	8	30	28	AB	20	28
53	1KS19EC053	NISARGA K	10	10	6	9	28	26	AB	18	27
54	1KS19EC054	NITHIN D	10	7	8	9	28	29	12	23	32
55	1KS19EC055	PAVAN KUMAR G R	10	10	10	10	26	29	15	24	34
56	1KS19EC056	POKURI MOUNIKA	10	10	8	10	25	23	16	22	32
57	1KS19EC057	POOJA S P	10	10	10	10	30	27	16	25	35
58	1KS19EC058	PRADEEP GADED	6	6	6	6	25	27	8	20	26
59	1KS19EC059	PRAKASH CHEGORE	8	7	6	7	24	27	12	21	28
60	1KS19EC061	PRASHANTH.S.K	10	10	8	10	29	28	15	24	34
61	1KS19EC062	PRAVEEN KUMAR N	9	6	6	7	29	27	9	22	29
62	1KS19EC063	PREETHAM G H	7	7	6	7	29	24	19	24	31
63	1KS19EC064	PRIYANKA K	10	7	6	8	29	28	20	26	34
64	1KS19EC065	RADHA KRISHNA L	10	10	6	9	26	22	17	22	31

65	1KS19EC066	RAJALAKSHMI S	10	10	10	10	25	27	17	23	33
66	1KS19EC067	RAMYASREE R	10	10	8	10	26	27	7	20	30
67	1KS19EC068	RANGASWAMY.U	10	7	6	8	28	27	18	25	33
68	1KS19EC069	ROHAN K R	10	10	6	9	30	27	9	22	31
69	1KS19EC070	S K BHARATESH	10	10	10	10	22	26	15	21	31
70	1KS19EC071	SABARISH I J	7	5	6	6	27	26	8	21	27
71	1KS19EC072	SAHANA K S	10	10	8	10	28	29	22	27	37
72	1KS19EC073	SAHANA S	10	10	6	9	29	27	13	23	32
73	1KS19EC074	SAI PRIYA T S	10	10	10	10	29	27	23	27	37
74	1KS19EC075	SAMIKSHA S	10	5	5	7	29	26	11	22	29
75	1KS19EC076	SANTOSH HEGDE	10	7	6	8	30	27	15	24	32
76	1KS19EC077	SATHVIK U M	10	5	6	7	19	25	10	18	25
77	1KS19EC078	SHAMITHA BIJOOR	10	10	8	10	29	29	16	25	35
78	1KS19EC079	SHASHANK	10	10	8	10	30	30	17	26	36
79	1KS19EC080	SHIVARAMA KRISHNA.K.P	6	0	0	2	26	26	AB	18	20
80	1KS19EC081	SHREYAMS D K	7	10	6	8	26	24	16	22	30
81	1KS19EC082	SHREYAS B ARADHYA	10	10	10	10	29	26	19	25	35
82	1KS19EC083	SHREYAS GOWDA	5	10	5	7	0	23	14	13	20
83	1KS19EC084	SHREYAS V BHARADWAJ	10	10	10	10	29	27	6	21	31
84	1KS19EC085	SHUBHAM KUMAR	9	7	6	8	29	25	14	23	31
85	1KS19EC086	SINCHANA M N	10	10	8	10	30	25	11	22	32
86	1KS19EC087	SRINIVAS S	10	10	6	9	28	25	11	22	31
87	1KS19EC088	SRINIVASAN M	8	10	8	9	28	24	16	23	32
88	1KS19EC089	SRIRAM	10	7	10	9	27	26	1	18	27
89	1KS19EC090	SUHAS.M	7	7	10	8	28	26	AB	18	26
90	1KS19EC092	SUMUKHA VASISHTA M R	10	10	6	9	30	29	15	25	34
91	1KS19EC093	SUSHMITHA S	10	10	8	10	29	28	14	24	34
92	1KS19EC094	SWAGATH AITHAL P G	10	10	10	10	28	26	18	24	34
93	1KS19EC095	SWATHI U	10	10	6	9	29	29	4	21	30
94	1KS19EC096	T N L RUTHVIK	10	7	6	8	16	27	14	19	27
95	1KS19EC097	TEJASHWINI P V	10	10	10	10	23	28	25	26	36
96	1KS19EC098	THEERTHANA S R	10	10	8	10	28	30	15	25	35
97	1KS19EC099	TUSHAR R VASISHTA	10	10	10	10	30	29	15	25	35
98	1KS19EC100	VAISHNAVI K	10	10	10	10	24	28	5	19	29

99	1KS19EC101	VANDANA G	10	10	8	10	28	29	27	28	38
100	1KS19EC102	VANDANA S	10	10	6	9	30	29	7	22	31
101	1KS19EC103	VIGNESH MUTHAIAH R	10	10	10	10	20	27	17	22	32
102	1KS19EC104	VIKAS S	8	8	10	9	24	27	7	20	29
103	1KS19EC105	VINUTH S REDDY	6	6	6	6	25	26	7	20	26
104	1KS19EC106	VISHAL SANJAY RAJU	10	10	5	9	29	26	13	23	32
105	1KS19EC107	VISHNU RAATA YADUNANDAN	8	8	8	8	28	25	7	20	28
106	1KS19EC108	YASHASWINI N	10	10	6	9	24	24	3	17	26
107	1KS18EC089	SNEHA N	7	7	6	7	24	26	AB	17	24
108	1KS19ET002	CHAITRA C	10	10	10	10	30	26	21	26	36
109	1KS19ET003	LITCHITHA M GOWDA	8	7	10	9	28	26	AB	18	27
110	1KS19ET004	MAHADEV A C	6	8	6	7	20	27	12	20	27
111	1KS19ET005	MRUTHYUNJAYA GUDIBANDE	4	10	10	8	24	25	12	21	29
112	1KS19ET006	N NELBIN	7	10	8	9	14	24	0	13	22
113	1KS19ET007	NIRANJAN S RAO	10	7	10	9	19	29	20	23	32
114	1KS19ET008	RISHI KUMAR S	10	5	5	7	17	28	1	16	23
115	1KS19ET009	ROHIT KUMAR	10	8	8	9	24	24	12	20	29
116	1KS19ET010	SHREYAS C R	10	8	6	8	25	23	13	21	29
117	1KS19ET011	SHWETHA K	3	10	8	7	29	20	22	24	31
118	1KS19ET012	VAISHNAVI S	7	7	6	7	24	23	1	16	23
119	1KS18TE005	ANKITHA . N	10	5	5	7	22	26	3	17	24
120	1KS20EC400	MADALA VIVEK KUMAR	9	9	8	9	11	10	10	11	20
121	1KS20EC400	RANJANA P	9	9	8	9	16	20	4	14	23
122	1KS20EC400	SINDHU J	9	9	8	9	18	18	5	14	23

K S INSTITUTE OF TECHNOLOGY

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

2020 – 21 ODD

List of students who are identified as slow learners and their marks in every internal assessment.

Subject with Code: : Computer Organization & Architecture-18EC35
Semester and Section: : 3rd A & B

Sl.No	USN Number	Name of the student	Remedial class Attendance			First Test Marks(30)	Second Test Marks(30)	Third Test Marks (30)
			19/10	21/10	22/10			
1	1KS19EC083	SHREYAS GOWDA	P	P	A	0	23	13
2	1KS19ET006	N NELBIN	A	P	P	14	24	13

Name and Signature of the Faculty

Signature of the HOD

K S INSTITUTE OF TECHNOLOGY BANGALORE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

18EC35 -COMPUTER ORGANIZATION & ARCHITECTURE

CHALLENGING QUESTIONS

1. The subroutine call instruction of a computer saves the return address in a processor register called the link register, RL. What would you do to allow subroutine nesting? Would your scheme allow the subroutine to call itself?

2. Register R5 is used in a program to point to the top of a stack. Write a sequence of instructions using the Index, Autoincrement, and Autodecrement addressing modes to perform each of the following tasks:

   (*a*) Pop the top two items off the stack, add them, and then push the result onto the stack.

3. Three devices, A, B, and C, are connected to the bus of a computer. I/O transfers for all three devices use interrupt control. Interrupt nesting for devices A and B is not allowed, but interrupt requests from C may be accepted while either A or B is being serviced. Suggest different ways in which this can be accomplished in each of the following cases:

   (*a*) The computer has one interrupt-request line.

   (*b*) Two interrupt-request lines, INTR1 and INTR2, are available, with INTR1 having higher priority.

4. The application program in a computer system with cache uses 1400 instruction acquisition bus cycle from cache memory and 100 from main memory. What is the hit rate? If cache memory operates with zero wait state and main memory bus cycles use three wait states, what is the average number of wait states experienced during the program execution?

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE - 560109

DEARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

TEACHING AND LEARNING

PEDAGOGY REPORT

Academic Year	**2020-21**
Name of the Faculty	**Dr.B Sudarshan**
Course Name /Code	**Computer Organization and Architecture**
Semester/Section	**3/A & B**
Activity Name	**KAHOOT QUIZ**
Topic Covered	**Basics of Computer organization and Architecture**
Date	**12.10.2020 – A section** **25.9.2020 – B section**
No. of Participants	**48-B section** **55-A section**
Objectives/Goals	**To make students understand the basics of Computer organization and architecture**
ICT Used	**KAHOOT on Mobiile**

Sample question

Following are different computer types

20scc

1. PC, Desktops, Workstations, Enterprise systems
2. PC, Laptop, Workstations, Enterprise systems
3. PC, Laptop, Workstations, Desktop
4. PC, Laptop, Desktop, Enterprise systems

Relevant PO's	**PO1, PO9, PO12**
Significance of Results/Outcomes	**Able to actively participate in the quiz and answer to the questions.**
Reflective Critique	**Students were actively participating in the quiz and requested for more such quiz.**

Proofs (Photographs/Videos/Reports/Charts/Models

https://drive.google.com/file/d/1d98p9oklvfBJDa3mYUx1MCAR5YjpzE5g/view?usp=sharing

Signature of Course In charge

Signature of HOD ECE

K S INSTITUTE OF TECHNOLOGY BANGALORE

DEPARTMENT OF ELECRONICS & COMMUNICATION ENGINEERING

NAME OF THE STAFF : Dr. B Sudarshan

SUBJECT CODE/NAME : 18EC35/COMPUTER ORGANIZATION AND ARCHITECTURE

SEMESTER/YEAR/SEC : III / II/ A & B

ACADEMIC YEAR : 2020-2021

Question Bank

Module – 1

1. Define computer program and computer memory.
2. Explain different types of computer.
3. With a neat block diagram, explain the basic functional blocks of a computer.
4. Explain the steps needed to execute an instruction ADD LOCA,R0
5. Explain connection between processor and main memory.
6. Differentiate between single and multiple bus structure.
7. Explain the use of Buffer Register.
8. Define (i) System Software (ii) Compiler (iii) Text Editor (iv) File (v) OS
9. With the help of a time line diagram show how user program and OS share the processor.
10. Define processor time and elapsed time.
11. Explain performance equation of a processor and mention how the performance can be improved.
12. How are signed numbers and characters stored in memory?
13. Differentiate between Big-Endian and Little-Endian Assignments with examples.
14. Explain Register Transfer Notation.
15. Explain basic Instruction Types.
16. Compare Straight Line and Branch Sequencing.
17. Explain some commonly used condition code flags. Why are condition codes used?.

Module – 2

1. Explain different addressing modes with examples.
2. Write an assembly language program to add N numbers using indirect addressing mode.
3. Write an assembly language program to add the scores of N students in each test and store it in memory location SUM1, SUM2, SUM3.
4. Differentiate between auto increment and auto decrement addressing modes with examples.
5. Define (i) Assembly language (ii) Assembler (iii) Source program (iv) Object program (v) mnemonics

6. Explain Assembler Directives with examples.
7. Explain the general format of an assembly language statement.
8. What is an assembler? Why is it used? What is 2 pass assembler?
9. What is Loader program? Why is it used?
10. What is debugger program? Why is it used?
11. How are decimal, binary and hexadecimal numbers represented?
12. Explain the bus connection for processor, keyboard and display.
13. Differentiate between memory mapped and program mapped I/O with examples.
14. Write an assembly language program to read a line of character and display it.
15. Compare Stack and Queue with examples.
16. Write and explain the instructions for SAFE PUSH and SAFE POP operations in a stack.
17. Define subroutines. Explain the procedures for passing parameters to subroutines.
18. Explain subroutine nesting.
19. Explain different types of shift and rotate operations with examples.
20. Registers R1 and R2 of a computer contain the decimal values 1200 and 4600. What is the effective address of the memory operand in each of the following instructions
21. Explain PUSH and POP operations of a stack with relevant instructions.
22. Register R5 is used to point to the top of a stack. Perform the following tasks using Index, Autoincrement and Autodecrement addressing modes.
a. Pop top 2 items off the stack, add them and then push the result to the top of the stack.
b. Pop the 5th item from the stack.

Module – 3

1. Explain the registers in keyboard and display interfaces.
2.Define
(a) Interrupt request signal (b) Interrupt acknowledge signal (c) Interrupt latency (d) Real Time Processing
3. Compare Subroutine and Interrupt Service Routine.
4. With a neat diagram explain the equivalent circuit used to implement common Interrupt request line.
5. Define Interrupts. Point out the methods of enabling and disabling interrupts.
6. List the sequence of events involved in handling an interrupt request from a single device.
7. Explain two methods of handling multiple devices during interrupts.
8. What is interrupt nesting? With a neat block diagram explain the implementation of Interrupt priority using individual request and acknowledge lines.
9. Compare Daisy Chain scheme with the scheme which is the combination of Daisy Chain and individual request and acknowledge lines for handling multiple interrupt requests.
10. Write an Assembly language program to read an input line from the keyboard and store the characters in successive byte locations in the memory starting at location LINE.
11. With a neat diagram explain DMA controller.
12. Explain the process of execution of ISR.
13. Explain with a neat diagram the hardware required to implement a common interrupt request line.
14. Explain different methods of handling multiple devices.
15. Explain the methods of handling simultaneous requests.

Module – 4

1. Apply the concept of virtual memory to make a program fit into the main memory.
2. Compare SRAM and DRAM memory cell.
3. Draw and explain the internal organization of 2Mx8 asynchronous DRAM chip.
4. Explain the principle of working of magnetic disk.
5. Illustrate the organization of 1Kx1 memory chip.
6. Make use of a neat diagram explain the working of CMOS memory cell.
7. Compare Flash Drives and Hard Disk Drives.
8. Define ROM and Analyze different types of ROMs.
9. Explain the connection of memory to the processor.
10. Define memory access time and memory cycle time.
11. Explain the internal organization of bit cells in the memory chip.
12. Explain the working of static RAM cell;
13. Explain the working of Asynchronous DRAM cell.
14. Explain (i) Disk buffer (ii) Disk controller
15. Define Booting and explain the process of booting.
16. What are floppy disks. List the advantages and disadvantages of the same.
17. Explain RAID Disk Arrays.

Module – 5

1. Write the sequence of operations and timing diagram for the instruction **MOV (R1), R2**
2. Write the sequence of control steps required for the following instruction using single bus architecture. **Add (R3), R1**
3. Write the sequence of control steps required for single bus structure for the following Instruction: Add the contents of memory location NUM to register R1.
4. Explain the single bus organization of data path inside a processor.
5. Explain the implementation of 1 bit register.
6. Write the sequence of operations to add the contents of register R1 with the contents of register R2 and store the result in R3.
7. Write the sequence of steps for the following instructions

   MOV (R1), R2

   MOV R2, (R1)
8. Draw and explain multiple bus organization of CPU. Write the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization.
9. Explain with neat diagram micro-programmed control method for design of control unit and write the micro-routine for the instruction Branch < 0.
10. Write the micro-routine for the instruction Branch < 0 and explain the organization of control unit for unconditional branch instruction.
11. Explain Hardwired Control Unit Organization in a processing unit.
12. Explain with a neat block diagram the structure of a complete processor.
13. Write and explain the control equations to generate Z and End control signals and represent the same using gates.

14. Explain Single Bus Organization of data path inside a processor.
15. Explain input and output gating for registers.
16. Define multiphase clocking.
17. Explain the implementation of 1 bit register.
18. Explain the process of Fetching a word from memory with the help of timing diagram and write the control sequence for the same.
19. Explain the process of storing a word into memory and write the control sequence for the same.
20. Write and explain the control sequences for unconditional and conditional branch nstructions.
21. Explain with a neat diagram, three bus organization of the data path.

CBCS SCHEME

USN 1 K S 1 9 E C 0 1 F

18EC35

Third Semester B.E. Degree Examination, July/August 2021
Computer Organization and Architecture

Time: 3 hrs. Max. Marks: 100

Note: Answer any FIVE full questions.

1 a. Explain following registers: (i) PC (ii) IR (iii) MAR (06 Marks)
b. Explain how user program and OS routine are sharing processor with printer. (08 Marks)
c. Explain basic performance equation. (06 Marks)

2 a. Perform using 2's complement arithmetic: (i) −5 + (−2) (ii) Subtract −5 from −7 (06 Marks)
b. Explain BIG-ENDIAN and LITTLE-ENDIAN assignment. (06 Marks)
c. Illustrate instruction execution and straight line sequencing for the program C ←[A] + [B]. [Assume that each instruction is 4 byte]. (08 Marks)

3 a. List the generic addressing modes with assembler syntax and addressing function. (10 Marks)
b. Explain shift and any two rotate instructions with relevant diagrams. (10 Marks)

4 a. Write assembly language program to add 'N' numbers and store the result in 'SUM'. Assume the following address:
(i) Program should start from '100'.
(ii) 'N' is stored at 204
(iii) Numbers are stored in memory from the address 208. Each number is 4 bytes.
(iv) 'SUM' is stored at 200
(v) Assume each instruction is 4 byte (08 Marks)
b. Explain stack concept with relevant diagrams. (08 Marks)
c. List the steps involved in 'CALL' and 'RETURN' instructions. (04 Marks)

5 a. Explain I/O interface for input device and also write the assembly program that reads the one LINE from the keyboard and echoes it back to the display. (10 Marks)
b. Explain methods used for enabling and disabling interrupts. (10 Marks)

6 a. Explain daisy chain method used for handling simultaneous interrupt request. (06 Marks)
b. Explain memory mapped I/O access. (06 Marks)
c. Explain use of DMA controller in computer system. (08 Marks)

7 a. Calculate number of address lines required to access following memory:
(i) 64 KB (ii) 512 MB (iii) 256 KB (iv) 8 GB (04 Marks)
b. Explain internal organization of 2M × 8 dynamic memory chip. (08 Marks)
c. Explain different types of nonvolatile memory. (08 Marks)

8 a. Explain cache memory and its relevant terms. (08 Marks)
b. Explain virtual memory organization. (06 Marks)
c. Explain magnetic disk principles. (06 Marks)

9 a. Explain single bus organization of the data path inside a processor. (10 Marks)
b. List the steps involved in memory read operation and also draw corresponding timing diagram. (10 Marks)

10 a. Write the control sequence for execution of the instruction Add (R_3), R_1. (06 Marks)
b. Explain block diagram of a complete processor. (06 Marks)
c. Explain micro programmed control concept. (08 Marks)

* * * * *

CBCS SCHEME

USN | 1 | K | S | 1 | 9 | E | T | 0 | 0 | 7 |

18EC35

Third Semester B.E. Degree Examination, July/August 2021
Computer Organization and Architecture

Time: 3 hrs. Max. Marks: 100

Note: Answer any FIVE full questions.

1 a. Explain following registers: (i) PC (ii) IR (iii) MAR (06 Marks)
b. Explain how user program and OS routine are sharing processor with printer. (08 Marks)
c. Explain basic performance equation. (06 Marks)

2 a. Perform using 2's complement arithmetic: (i) −5 + (−2) (ii) Subtract −5 from −7 (06 Marks)
b. Explain BIG-ENDIAN and LITTLE-ENDIAN assignment. (06 Marks)
c. Illustrate instruction execution and straight line sequencing for the program C ← [A] + [B] [Assume that each instruction is 4 byte]. (08 Marks)

3 a. List the generic addressing modes with assembler syntax and addressing function. (10 Marks)
b. Explain shift and any two rotate instructions with relevant diagrams. (10 Marks)

4 a. Write assembly language program to add 'N' numbers and store the result in 'SUM' Assume the following address:
(i) Program should start from '100'.
(ii) 'N' is stored at 204
(iii) Numbers are stored in memory from the address 208. Each number is 4 bytes.
(iv) 'SUM' is stored at 200
(v) Assume each instruction is 4 byte (08 Marks)
b. Explain stack concept with relevant diagrams. (08 Marks)
c. List the steps involved in 'CALL' and 'RETURN' instructions. (04 Marks)

5 a. Explain I/O interface for input device and also write the assembly program that reads the one LINE from the keyboard and echoes it back to the display. (10 Marks)
b. Explain methods used for enabling and disabling interrupts. (10 Marks)

6 a. Explain daisy chain method used for handling simultaneous interrupt request. (06 Marks)
b. Explain memory mapped I/O access. (06 Marks)
c. Explain use of DMA controller in computer system. (08 Marks)

7 a. Calculate number of address lines required to access following memory:
(i) 64 KB (ii) 512 MB (iii) 256 KB (iv) 8 GB (04 Marks)
b. Explain internal organization of 2M × 8 dynamic memory chip. (08 Marks)
c. Explain different types of nonvolatile memory. (08 Marks)

8 a. Explain cache memory and its relevant terms. (08 Marks)
b. Explain virtual memory organization. (06 Marks)
c. Explain magnetic disk principles. (06 Marks)

9 a. Explain single bus organization of the data path inside a processor. (10 Marks)
b. List the steps involved in memory read operation and also draw corresponding timing diagram. (10 Marks)

10 a. Write the control sequence for execution of the instruction Add (R_3), R_1. (06 Marks)
b. Explain block diagram of a complete processor. (06 Marks)
c. Explain micro programmed control concept. (08 Marks)

* * * * *

GBCS SCHEME

USN 1 K S 1 9 E C 1 0 5

18EC35

Third Semester B.E. Degree Examination, Jan./Feb. 2021
Computer Organization and Architecture

Time: 3 hrs. Max. Marks: 100

Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.
2. Write neat diagrams wherever necessary.

Module-1

1 a. With a neat diagram, describe the functional units of a computer. Give few examples for I/O devices. **(06 Marks)**

b. Discuss IEEE standard for single-precision and double-precision floating point numbers, with standard notations. **(06 Marks)**

c. Develop an Assembly Language Program (ALP) for the expression $Y = Ax^2 + BCx + D$ using 3-address, 2-address and 1-address instruction formats. Assume A, B, C, D, Y as memory locations and x as immediate data. **(08 Marks)**

OR

2 a. With a neat diagram, discuss the operational concepts in a computer highlighting the role of PC, MAR, MDR and IR. **(08 Marks)**

b. Perform subtraction on the following pairs of numbers using 5-bit signed 2's-complement format. Indicate about overflow in each case:
i) +10 and -8 ii) +12 and +9 iii) -15 and -9 iv) -14 and +5 **(08 Marks)**

c. Distinguish between Big-endian and little-endian memory assignment. With a neat sketch, show how the number 26789435 is stored using these methods. **(04 Marks)**

Module-2

3 a. Define addressing mode. Explain any four basic addressing modes with syntax and examples. **(08 Marks)**

b. What is subroutine? With a pseudocode or program segment, illustrate parameter passing using registers. **(06 Marks)**

c. Consider a database of marks scored by students in 3 tests, stored in memory starting at address LIST. Each student record consists of studentID followed by marks in 3 tests. Assume each of these to be 4 bytes in size. There are 50 students in the class and this value is stored at location NUM.
i) Sketch the memory map showing all details
ii) Develop an ALP using Indexed Addressing mode, to compute the sum of scores by all the students in Test2 and store the result in location SUM. Write appropriate comments. **(06 Marks)**

OR

4 a. Discuss Auto-increment and Auto-decrement addressing modes with syntax. Consider a set of numbers (each 4 bytes in size) stored in memory starting at address TABLE. Total numbers are N and this value is stored at location LOCN.
i) Sketch the memory map showing all details
ii) Develop an ALP using Auto-increment addressing mode, to compute the sum of all numbers and store the result at memory address RESUTL. Write appropriate comments. **(08 Marks)**

b. Define stack. Explain PUSH and POP operations on stack with neat sketches and examples. (06 Marks)

c. Consider a register R1 to size 16-bits with initial data 5867_d. With neat sketches, depict the output in each case, after performing the following operations:
i) LshiftL #2, R1 ii) AshiftR #1, R1 iii) RotateR #1, R1
Note: For each operation, R1 value is to be taken as 5867_d and carry flag is indicated cleared. (06 Marks)

Module-3

5 a. Distinguish between memory mapped I/O and standard I/O. Write a program segment to read a line of text from keyboard and display it. (08 Marks)

b. What is interrupt priority? Why is it necessary? With relevant diagram, discuss daisy-chain method of handling multiple interrupt requests. (06 Marks)

c. Explain distributed arbitration mechanism in DMA with a neat diagram. (06 Marks)

OR

6 a. With a neat diagram, discuss implementation of interrupt priority using individual request and acknowledge lines. (06 Marks)

b. Briefly explain: i) Vectored interrupts and ii) Registers in a DMA interface. (06 Marks)

c. Explain centralized arbitration mechanism in DMA with a neat sketch and timing diagram. (08 Marks)

Module-4

7 a. Classify memory in a computer. With a neat diagram, describe the organization of 2M × 8 DRAM chip. (08 Marks)

b. What is cache memory? Explain direct mapping technique with a neat diagram. (08 Marks)

c. Briefly discuss the concept of virtual memory with a diagram. (04 Marks)

OR

8 a. Briefly explain the working of 1-bit CMOS SRAM cell with a schematic. (06 Marks)

b. What is mapping function? Explain set-associative cache mapping technique with a relevant diagram. (08 Marks)

c. With a neat diagram, explain the principle of working of magnetic disk. (06 Marks)

Module-5

9 a. Explain single-bus organization of data path in a processor with a neat diagram. Highlight the importance of gating signals. (08 Marks)

b. Develop the complete control signal sequence for the instruction Add (R1), R3 with appropriate remarks. (06 Marks)

c. Discuss micro programmed control unit design with relevant diagrams. (06 Marks)

OR

10 a. List different ways of improving CPU performance. With a neat diagram, discuss three-bus organization of CPU. Compare the performance with single-bus organization. (08 Marks)

b. Discuss Hardwired control unit organization with relevant diagrams and illustrate the logic to generate Z_{in} control signal. (08 Marks)

c. Define the following:
i) Gating signal ii) Control word iii) Microroutine iv) Control store. (04 Marks)

* * * * *

CBCS SCHEME

USN | | | | | | | | | | 18EC35

Third Semester B.E. Degree Examination, Dec.2019/Jan.2020
Computer Organization and Architecture

Time: 3 hrs. Max. Marks: 100

Note: Answer any FIVE full questions, choosing ONE full question from each module.

Module-1

1 a. With a neat diagram, explain basic operational concept of computer. (10 Marks)
b. Explain in brief different types of key parameters that affect the processor performance. (05 Marks)
c. Explain the Bus Structures. (05 Marks)

OR

2 a. Illustrate Instruction and Instruction sequencing with an example. (10 Marks)
b. Define Byte Addressability, Big-endian and Little-endian assignment. (06 Marks)
c. Represent 85.125 in IEEE floating point using single precision. (04 Marks)

Module-2

3 a. What is an addressing mode? Explain any five types of addressing modes with example. (10 Marks)
b. Write a program to add 'n' number using indirect addressing mode. (06 Marks)
c. Explain various assembler directives used in assembly language program. (04 Marks)

OR

4 a. Explain stack operation with an example. (10 Marks)
b. Explain subroutine linkage with an example using linkage register. (06 Marks)
c. Explain the shift and rotate operations with example. (04 Marks)

Module-3

5 a. Showing the possible register configuration in I/O interface, explain program controlled input/output. (10 Marks)
b. What is an interrupt? With an example illustrate the concept of interrupt. (10 Marks)

OR

6 a. Explain in detail, the situations where a number of devices capable of initiating interrupts are connected to processor. How to resolve the problems? (10 Marks)
b. Explain the registers involved in a DMA interface, to illustrate DMA. (06 Marks)
c. Explain the concept of Vectored Interrupt. (04 Marks)

Module-4

7 a. With figure, explain Internal Organization of 2M×8 dynamic memory chip. (10 Marks)
b. Illustrate Internal structure of static memories. (10 Marks)

OR

8 a. With a neat diagram, explain virtual memory organization. (10 Marks)
b. Briefly explain any four non-volatile memory concepts. (05 Marks)
c. Briefly explain secondary storage devices. (05 Marks)

Module-5

9 a. Explain the three-bus organization of the processor and its advantages. (10 marks)
b. Discuss the organization of hardwired control unit. (05 marks)
c. Discuss the control sequence for execution of instruction ADD(R_3), R_1 (05 marks)

OR

10 a. With a block diagram, describe the organization of a micro programmed control unit. (10 marks)

b. Describe the sequence of control signals to be generated to fetch an instruction from memory in a single bus organization. (10 marks)

* * * * *

CBCS SCHEME

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

18EC35

Third Semester B.E. Degree Examination, Aug./Sept.2020
Computer Organization and Architecture

Time: 3 hrs. Max. Marks: 100

Note: *Answer any FIVE full questions, choosing ONE full question from each module.*

Module-1

1 a. Explain the operation of computer with neat block diagram. (10 Marks)
b. Explain computer basic performance equation. (04 Marks)
c. Explain following with an example : i) Three – address instruction
ii) Two – address instruction iii) One – address instruction. (06 Marks)

OR

2 a. Explain Single –BUS structure in computer. (06 Marks)
b. Explain system software functions in computer. (06 Marks)
c. What is Operating system? Explain user program and OS routine sharing the processor. (08 Marks)

Module-2

3 a. Explain Big–Endian and Little–Endian with neat diagram. (08 Marks)
b. Explain memory operations with examples. (04 Marks)
c. Explain condition codes with examples. (08 Marks

OR

4 a. Discuss following addressing modes with example :
i) Immediate ii) Register iii) Direct iv) Indirect v) Index. (10 Marks)
b. What are assembler directive? Explain any five assembler directives. (10 Marks)

Module-3

5 a. With a neat diagram, explain how to interface printer to the processor. (10 Marks)
b. Define Interrupt. Point out and explain the various ways of enabling and disabling interrupts. (10 Marks)

OR

6 a. Explain the following method of handling interrupts from multiple devices.
i) Daisy chain method ii) Priority structure. (10 Marks)
b. Explain operation of DMA with neat diagram. (10 Marks)

Module-4

7 a. Explain internal organization of 16 × 8 memory chip. (10 Marks)
b. Discuss a single–transistor dynamic memory cell. (06 Marks)
c. Write a note on Virtual Memory. (04 Marks)

OR

8 a. Draw and explain the internal organization of 2M × 8 asyndronous DRAM Chip. (08 Marks)
b. Describe the principles of magnetic disk. (06 Marks)
c. What is mapping? Explain set associative cache mapping techniques. (06 Marks)

Module-5

9 a. Discuss with neat diagram, the single bus organization of data path inside a processor. (10 Marks)
b. What are the action required to execute a complete instruction ADD(R3), R1. (10 Marks)

OR

10 a. Draw and explain multiple bus organization of CPU. (10 Marks)
b. Draw and explain organization of the control unit to allow conditional branching in the microprogram. (10 Marks)

* * * * *

CBCS SCHEME

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

15CS34

Third Semester B.E. Degree Examination, June/July 2018
Computer Organization

Time: 3 hrs. Max. Marks: 80

Note: ***Answer any FIVE full questions, choosing one full question from each module.***

Module-1

1 a. Define Addressing Mode. Give the details of different addressing modes. **(08 Marks)**

b. Describe the basic operational concepts between the processor and memory. **(08 Marks)**

OR

2 a. What is Subroutine? How to pass parameters to subroutines? Illustrate with an example. **(08 Marks)**

b. How to encode assembly instructions into 32-bit words? Explain with examples. **(08 Marks)**

Module-2

3 a. Define Bus Arbitration. With diagrams, explain the centralized bus arbitration mechanism. **(08 Marks)**

b. With the help of timing diagram, briefly discuss the main phases of SCSI bus involved in its operation. **(08 Marks)**

OR

4 a. With neat diagrams, explain how to interface printer to the processor. **(08 Marks)**

b. Explain the following methods of handling interrupts from multiple devices.
i) Interrupt nesting/priority structure ii) Daisy chain method. **(08 Marks)**

Module-3

5 a. Describe how to translate virtual address into physical address with diagram. **(08 Marks)**

b. Draw and explain the internal organisation of 2M × 8 asynchronous DRAM chip. **(08 Marks)**

OR

6 a. Describe any two mapping functions in cache. **(08 Marks)**

b. Describe the principles of magnetic disk. **(08 Marks)**

Module-4

7 a. Perform the operations on 5 – bit signed numbers using 2's complement system. Also indicate whether overflow has occurred. **(06 Marks)**
i) (-10) + (-13) ii) (-10) – (-13) iii) (-2) + (-9).

b. Perform the multiplication of 13 and -6 using Booth algorithm and Bit – pair recoding method. **(10 Marks)**

OR

8 a. Perform the restoring division for 8 ÷ 3 by showing all the steps. **(06 Marks)**

b. Explain the logic diagram of 4 – bit carry look ahead adder and its operations. **(10 Marks)**

Module-5

9 a. Draw and explain multiple bus organization along with its advantages. **(10 Marks)**

b. Write down the control sequence for the instruction Add (R_3) , R_1 for single bus organization. **(06 Marks)**

OR

10 a. With block diagram, explain the general requirements and working of digital camera. **(10 Marks)**

b. Write the control sequence for an unconditional branch instruction. **(06 Marks)**

* * * * *

CBCS Scheme

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

15CS34

Third Semester B.E. Degree Examination, June/July 2017

Computer Organization

Time: 3 hrs. Max. Marks: 80

Note: Answer any FIVE full questions, choosing one full question from each module.

Module-1

1 a. With a neat block diagram discuss the basic operational concept of a computer. **(06 Marks)**

b. Explain the methods to improve the performance of computer. **(04 Marks)**

c. Explain Big-Endian, little Endian and assignment byte addressability. **(06 Marks)**

OR

2 a. What are addressing modes? Explain the different 4 types addressing modes with example. **(08 Marks)**

b. Write the use of Rotate and shift instruction with example. **(04 Marks)**

c. What is stack and queue? Write the line of code to implement the same. **(04 Marks)**

Module-2

3 a. Define bus arbitration? Explain detail any one approach of bus arbitaration. **(08 Marks)**

b. What are priority interrupts? Explain any one interrupt priority scheme. **(04 Marks)**

c. Write a note on register in DMA interface. **(04 Marks)**

OR

4 a. With a block diagram explain how the printer interfaced to processor. **(08 Marks)**

b. Explain the following with respect to U.S.B

i) U.S.B Architecture

ii) U.S.B protocols. **(08 Marks)**

Module-3

5 a. Define :

i) Memory Latency

ii) Memory bandwidth

iii) Hit-rate

iv) Miss-penality. **(04 Marks)**

b. With a neat diagram explain the internal organization of a 2M×8 dynamic memory chip. **(06 Marks)**

c. Explain Associative mapping technique and set Associative mapping technique. **(06 Marks)**

OR

6 a. What is virtual memory? With a diagram explain how virtual memory address is translated. **(08 Marks)**

b. Write a note on :

i) Magnetic tape system

ii) Flash memory. **(08 Marks)**

Module-4

7 a. Perform following operations on the 5-bit signed numbers using 2's complement representation system. Also indicate whether overflow has occurred.
i) $(-9)+(-7)$ ii) $(+7)-(-8)$. **(04 Marks)**

b. Explain with a neat block diagram, 4 bit carry lookahead adder. **(05 Marks)**

c. Explain the concept of carry save addition for the multiplication operation, $M \times Q = P$ for 4-bit operands with diagram and suitable example. **(07 Marks)**

OR

8 a. Multiply the following signed 2's complement numbers using Booth's algorithm multiplicand = $(010111)_2$, multiplier = $(110110)_2$. **(05 Marks)**

b. Perform division operation on the following unsigned numbers using the restoring method. Dividend = $(10101)_2$ Divisor = $(00100)_2$. **(05 Marks)**

c. With a neat diagram, explain the floating point addition/subtraction unit. **(06 Marks)**

Module-5

9 a. Draw and explain multiple bus organization of CPU. And write the control sequence for the instruction Add R4, R5, R6 for the multiple bus organization. **(08 Marks)**

b. Explain with neat diagram, micro-programmed control method for design of control unit and write the micro-routine for the instruction Branch < 0. **(08 Marks)**

OR

10 a. With block diagram, explain the working of microwave oven in an embedded system. **(08 Marks)**

b. With block diagram, explain parallel I/O interface. **(08 Marks)**

* * * * *

K.S.Institute of Technology,Bangalore -109
Department of Electronics and Communication Engg
3RD sem Course End Survey 2020-21

COURSE : Computer Organization & Architecture COURSE CODE:18EC35

Q1.Rate your understanding on basic structure of a computer

Q2.Rate your level of understanding of I/O devices and their handling

Q3.Rate your ability to apply the knowledge gained in interrupts

Q4.Rate your ability to analyze the operations of control unit

Q5.Rate your level of analyzing simple instructions and programs

Sl No	Date	USN	Student Name	Section	Faculty Name	Q1	Q2	Q3	Q4	Q5
1	3/12/2021	1KS18EC085	Shubham Kumar singh A	III-B	Dr.Sudarshan B	3	3	3	3	3
2	3/12/2021	1KS18EC089	Sneha	III-B	Dr.Sudarshan B	1	3	3	3	3
3	3/12/2021	1KS19EC001	Abhilash A S	III-A	Dr.Sudarshan B	2	2	2	2	2
4	2/26/2021	1KS19EC002	ABHISHEK C	III-A	Dr.Sudarshan B	3	2	2	2	2
5	3/12/2021	1KS19EC003	Aishwarya basavaraja kembavi	III-A	Dr.Sudarshan B	3	2	3	3	2
6	2/26/2021	1KS19EC004	Aishwarya MG	III-A	Dr.Sudarshan B	3	3	3	2	2
7	3/12/2021	1KS19EC004	Aishwarya MG	III-A	Dr.Sudarshan B	2	2	2	2	2
8	3/12/2021	1KS19EC005	AKSHAY KUMAR D	III-A	Dr.Sudarshan B	3	3	3	3	3
9	3/12/2021	1KS19EC006	Akshitha	III-A	Dr.Sudarshan B	3	3	3	3	3
10	3/12/2021	1KS19EC007	AMRUTA	III-A	Dr.Sudarshan B	2	2	2	2	2
11	3/12/2021	1KS19EC008	Amulya R	III-A	Dr.Sudarshan B	3	3	2	3	3
12	2/26/2021	1KS19EC009	Anitha.S	III-A	Dr.Sudarshan B	3	3	3	3	2
13	2/26/2021	1KS19EC010	Anjali Y J	III-A	Dr.Sudarshan B	2	2	2	2	2
14	3/12/2021	1ks19ec011	Archana Yadav M	III-A	Dr.Sudarshan B	3	3	3	3	3
15	3/12/2021	1KS19EC012	Ashritha.R	III-A	Dr.Sudarshan B	2	2	2	2	2
16	3/12/2021	1KS19EC014	Bhavana S	III-A	Dr.Sudarshan B	3	3	3	3	3
17	3/12/2021	1KS19EC015	Chaitra p	III-A	Dr.Sudarshan B	3	3	3	2	2
18	2/26/2021	1KS19EC016	Chandan RAJ Y	III-A	Dr.Sudarshan B	3	3	3	3	3
19	2/26/2021	1KS19EC017	Rajasekhar	III-A	Dr.Sudarshan B	3	2	2	2	3
20	3/12/2021	1KS19EC017	Chandana	III-A	Dr.Sudarshan B	2	2	2	2	2
21	2/26/2021	1KS19EC019	Chiranthana Yogananda.K	III-A	Dr.Sudarshan B	2	2	2	2	2
22	3/12/2021	1KS19EC020	D Nayan	III-A	Dr.Sudarshan B	2	2	2	2	3

Sl No	Date	USN	Student Name	Section	Faculty Name	Q1	Q2	Q3	Q4	Q5
23	2/26/2021	1KS19EC021	Danesh Raju v	III-A	Dr.Sudarshan B	2	2	2	2	2
24	3/12/2021	1KS19EC022	Davino Joseph	III-A	Dr.Sudarshan B	3	3	3	3	3
25	3/12/2021	1KS19EC023	DHANYA SUKANTH B.K.	III-A	Dr.Sudarshan B	3	3	3	3	3
26	3/12/2021	1KS19EC024	Dheemanth kn	III-A	Dr.Sudarshan B	3	3	3	3	3
27	2/26/2021	1KS19EC025	Disha Shivani	III-A	Dr.Sudarshan B	3	3	2	3	2
28	2/26/2021	1KS19EC026	Eram fathima	III-A	Dr.Sudarshan B	3	3	3	2	2
29	3/12/2021	1KS19EC027	Gayathri.P.K	III-A	Dr.Sudarshan B	2	2	2	2	2
30	3/12/2021	1KS19EC028	Gayathri R Warrier	III-A	Dr.Sudarshan B	3	3	3	3	3
31	3/12/2021	1KS19EC029	GONUGUNTLA SAI SIDDARTHA	III-A	Dr.Sudarshan B	3	3	3	3	3
32	3/12/2021	1KS19EC030	Gowri	III-A	Dr.Sudarshan B	3	3	3	3	3
33	3/14/2021	1KS19EC032	Harshitha by	III-A	Dr.Sudarshan B	2	3	2	2	3
34	3/12/2021	1KS19EC033	Hemanth	III-A	Dr.Sudarshan B	3	2	2	3	3
35	2/26/2021	1KS19EC035	Jagruti pai	III-A	Dr.Sudarshan B	3	3	3	3	3
36	2/26/2021	1KS19EC036	Jayanth. MB	III-A	Dr.Sudarshan B	3	2	2	2	3
37	2/26/2021	1KS19EC037	Manogna K M	III-A	Dr.Sudarshan B	3	2	2	3	2
38	2/26/2021	1KS19EC038	Karthik.k	III-A	Dr.Sudarshan B	3	3	2	3	2
39	3/12/2021	1KS19EC039	Kashyap p	III-A	Dr.Sudarshan B	3	2	2	2	2
40	3/12/2021	1KS19EC040	Krupa A	III-A	Dr.Sudarshan B	3	3	2	2	3
41	2/26/2021	1KS19EC041	Kruthik s	III-A	Dr.Sudarshan B	2	3	1	3	3
42	3/12/2021	1KS19EC042	LAKSHMAN KUMARA. B	III-A	Dr.Sudarshan B	2	2	2	2	2
43	3/12/2021	1KS19EC043	Likitha H	III-A	Dr.Sudarshan B	3	3	3	3	3
44	3/12/2021	1KS19EC044	M.Lokeshwari	III-A	Dr.Sudarshan B	3	3	3	3	3
45	3/12/2021	1KS19EC045	Manu N Kandra	III-A	Dr.Sudarshan B	3	3	3	3	3
46	3/12/2021	1KS19EC046	Meghana H P	III-A	Dr.Sudarshan B	3	3	3	3	3
47	3/12/2021	1KS19EC047	MOHAMMAD RAKHEEB M R	III-A	Dr.Sudarshan B	2	2	2	2	2
48	3/12/2021	1KS19EC048	Mohith Kumar G	III-A	Dr.Sudarshan B	2	2	2	2	2
49	2/26/2021	1KS19EC049	MONIKA V ARYA	III-A	Dr.Sudarshan B	3	3	2	2	3
50	2/26/2021	1KS19EC050	Monisha B K	III-A	Dr.Sudarshan B	3	3	2	3	2
51	3/12/2021	1KS19EC051	N.Anila	III-A	Dr.Sudarshan B	2	2	2	2	2
52	3/12/2021	1KS19EC053	Nisargak	III-A	Dr.Sudarshan B	1	1	1	1	1
53	3/12/2021	1KS19EC054	Nithin D	III-A	Dr.Sudarshan B	3	3	2	2	3

Sl No	Date	USN	Student Name	Section	Faculty Name	Q1	Q2	Q3	Q4	Q5
54	2/26/2021	1KS19EC055	Pavan Kumar G R	III-A	Dr.Sudarshan B	3	2	2	2	2
55	3/1/2021	1KS19EC056	Pokuri Mounika	III-A	Dr.Sudarshan B	2	2	3	3	3
56	3/12/2021	1KS19EC057	Pooja Sp	III-A	Dr.Sudarshan B	3	2	2	2	2
57	3/12/2021	1ks19ec058	Pradeep gaded	III-A	Dr.Sudarshan B	3	3	3	3	3
58	3/12/2021	1KS19EC059	Prakash Chegore	III-A	Dr.Sudarshan B	2	2	2	2	2
59	2/26/2021	1KS19EC061	Prashanth SK	III-A	Dr.Sudarshan B	3	2	3	2	3
60	3/12/2021	1KS19EC062	Praveen Kumar.N	III-A	Dr.Sudarshan B	2	2	2	2	2
61	3/12/2021	1KS19EC063	PREETHAM G H	III-A	Dr.Sudarshan B	3	2	2	2	3
62	3/12/2021	1KS19EC064	Priyanka K	III-A	Dr.Sudarshan B	3	2	2	3	3
63	3/12/2021	1KS19EC065	Radhakrishna L	III-A	Dr.Sudarshan B	3	3	3	3	3
64	3/12/2021	1KS19EC066	Rajalakshmi S	III-A	Dr.Sudarshan B	2	2	2	2	2
65	3/12/2021	1KS19EC067	Ramya sree	III-B	Dr.Sudarshan B	2	2	2	2	2
66	3/12/2021	1KS19EC068	Rangaswamy u	III-B	Dr.Sudarshan B	2	3	2	3	3
67	3/12/2021	1KS19EC069	Rohan K R	III-B	Dr.Sudarshan B	3	3	3	3	3
68	2/26/2021	1KS19EC070	SK Bharatesh	III-B	Dr.Sudarshan B	3	2	2	2	2
69	3/12/2021	1KS19EC071	Sabarish IJ	III-B	Dr.Sudarshan B	2	2	3	2	3
70	2/26/2021	1KS19EC072	Sahana.K.S	III-B	Dr.Sudarshan B	3	2	2	2	2
71	3/12/2021	1KS19EC073	Sahana.S	III-B	Dr.Sudarshan B	2	2	2	2	2
72	3/1/2021	1KS19EC074	Sai Priya TS	III-B	Dr.Sudarshan B	2	2	2	2	2
73	3/12/2021	1KS19EC075	SAMIKSHA S	III-B	Dr.Sudarshan B	3	3	3	3	3
74	3/12/2021	1KS19EC076	Santosh Hegde	III-B	Dr.Sudarshan B	2	2	2	2	2
75	2/27/2021	1KS19EC077	SATHVIK U.M	III-B	Dr.Sudarshan B	2	2	3	3	3
76	3/12/2021	1KS19EC078	Shamitha Bijoor	III-B	Dr.Sudarshan B	2	2	2	2	2
77	3/11/2021	1KS19EC079	Shashank Kashyap hr	III-B	Dr.Sudarshan B	2	2	2	2	2
78	2/26/2021	1KS19EC081	Shreyams D.K	III-B	Dr.Sudarshan B	2	2	2	2	2
79	3/12/2021	1KS19EC082	Shreyas B Aradhya	III-B	Dr.Sudarshan B	3	3	2	2	3
80	3/12/2021	1KS19EC083	Shreyas Gowda	III-B	Dr.Sudarshan B	3	3	3	3	3
81	2/26/2021	1KS19EC084	Shreyas V Bharadwaj	III-B	Dr.Sudarshan B	2	2	2	2	2
82	3/12/2021	1KS19EC086	Sinchana mn	III-B	Dr.Sudarshan B	2	2	2	2	2
83	3/12/2021	1KS19EC087	Srinivas S	III-B	Dr.Sudarshan B	2	3	3	2	2
84	2/26/2021	1KS19EC088	Srinivasan M	III-B	Dr.Sudarshan B	3	3	2	2	2

Sl No	Date	USN	Student Name	Section	Faculty Name	Q1	Q2	Q3	Q4	Q5
85	3/12/2021	1ks19ec089	Sriram	III-B	Dr.Sudarshan B	3	2	1	1	1
86	3/12/2021	1KS19EC090	Suhas m	III-B	Dr.Sudarshan B	1	1	1	1	1
87	3/15/2021	1KS19EC093	SUSHMITHA S	III-B	Dr.Sudarshan B	3	3	3	3	3
88	3/12/2021	1ks19ec094	Swagathaithal	III-B	Dr.Sudarshan B	2	2	3	2	2
89	3/12/2021	1KS19EC095	Swathi.U	III-B	Dr.Sudarshan B	2	2	2	2	2
90	2/26/2021	1KS19EC096	Ruthvik T	III-B	Dr.Sudarshan B	2	3	3	2	2
91	3/12/2021	1ks19ec097	Tejashwini pv	III-B	Dr.Sudarshan B	3	3	2	2	3
92	2/27/2021	1KS19EC098	Theerthana s r	III-B	Dr.Sudarshan B	2	2	2	2	2
93	2/26/2021	1KS19EC099	Tushar R Vasishta	III-B	Dr.Sudarshan B	3	3	3	3	3
94	3/12/2021	1KS19EC100	Vaishnavi k	III-B	Dr.Sudarshan B	2	2	2	2	2
95	3/12/2021	1KS19EC101	Vandana.G	III-B	Dr.Sudarshan B	3	2	2	2	2
96	3/12/2021	1KS19EC102	Vandana S	III-B	Dr.Sudarshan B	2	2	2	2	2
97	2/26/2021	1KS19EC103	Vignesh muthaiah r	III-B	Dr.Sudarshan B	3	3	2	3	3
98	3/15/2021	1KS19EC104	Vikas S	III-B	Dr.Sudarshan B	2	2	2	2	2
99	3/15/2021	1KS19EC105	VINUTH S REDDY	III-B	Dr.Sudarshan B	3	2	3	3	3
100	3/15/2021	1KS19EC106	VISHAL SANJAY RAJU	III-B	Dr.Sudarshan B	2	2	2	2	2
101	3/12/2021	1KS19EC107	Vishnuraata Yadunandan	III-B	Dr.Sudarshan B	2	2	2	2	2
102	3/12/2021	1KS19EC108	Yashaswini N	III-B	Dr.Sudarshan B	3	3	3	3	3
103	2/28/2021	1KS19ET002	Chaitra C	III-B	Dr.Sudarshan B	3	3	3	3	3
104	2/26/2021	1KS19ET003	Litchitha Gowda	III-B	Dr.Sudarshan B	2	2	2	2	2
105	3/12/2021	1KS19ET004	MAHADEV AC	III-B	Dr.Sudarshan B	3	3	3	3	2
106	2/27/2021	1KS19ET006	NELBIN N	III-B	Dr.Sudarshan B	1	1	1	1	1
107	2/26/2021	1KS19ET007	Niranjan S Rao	III-B	Dr.Sudarshan B	3	3	3	3	3
108	3/12/2021	1KS19ET009	ROHIT KUMAR	III-B	Dr.Sudarshan B	3	3	3	3	3
109	3/12/2021	1ks19et011	Shwethak	III-B	Dr.Sudarshan B	2	2	2	2	2
110	3/12/2021	1KS19ET012	Vaishnavi.s	III-B	Dr.Sudarshan B	2	2	3	3	3
111	2/27/2021	1KS1ET010	Shreyas C R	III-B	Dr.Sudarshan B	2	2	2	2	2
					NO. OF 1S	**4**	**3**	**5**	**4**	**4**
					Total count	**111**	**111**	**111**	**111**	**111**
					Percentage	**96.40**	**97.30**	**95.50**	**96.40**	**96.40**
					Average	**96.40**				

K.S. INSTITUTE OF TECHNOLOGY, BANGALORE
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGG

YEAR / SEMESTER	**2ND YEAR/ 3RD SEMESTER**
COURSE TITLE	**COMPUTER ORGANIZATION & ARCHITECTURE**
COURSE CODE	**18EC35**
ACADEMIC YEAR	**2020-21**
BATCH	**2019-23**

CO Attai	Significance
Level 3	60% and above students should have scored >= 60% of Total marks
Level 2	55% to 59% of students should have scored >= 60% of Total marks
Level 1	50% to 54% of students should have scored >= 60% of Total marks

For Direct attainment , 50% of CIE and 50% of SEE marks are considered.
For indirect attainment, Course end survey is considered.
CO attainment is 90%of direct attainment + 10% of Indirect atttainment.
PO attainment = CO-PO mapping strength/3 * CO attainment .

SI. NO.	USN	NAME	IA1							Assignment 1							IA2										Assignment 2										IA3								Assignment 3							EXTERNA		
			IA1	CO1	core	Target 60%	CO2	score	Target 60%	A1	CO1	score	Target 60%	CO2	score	Target 60%	IA2	CO2	score	Target 60%	CO3	score	Target 60%	CO4	score	Target 60%	A2	CO2	score	Target 60%	CO3	score	Target 60%	CO4	score	Target 60%	IA3	CO4	core	Target 60%	CO5	score	Target 60%	A3	CO4	core	Target 60%	CO5	score	Target 60%	SEE	Scores	Target 60%
	Maximum Marks		**30**	**18**			**12**			**10**	**6**			**4**			**30**	**6**			**18**			**6**			**10**	**2**			**6**			**2**			**30**	**12**			**18**			**10**	**6**			**4**			**60**		
1	1KS19EC001	ABHILASH	22	15	3	Y	7	2	N	10	6	3	Y	4	3	Y	26	**6**	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	14	10	3	Y	4	0	N	6	2	0	N	4	3	Y	**21**	0	N
2	1KS19EC002	ABHISHEK CHANDRESH	30	18	3	Y	12	3	Y	5	3	1	N	2	1	N	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	9	4	0	N	5	0	N	10	4	3	Y	6	3	Y	**17**	0	N
3	1KS19EC003	AISHWARYA BASAVARAJ KEMBAVI	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	21	10	3	Y	11	3	Y	10	4	3	Y	6	3	Y	**28**	0	N
4	1KS19EC004	AISHWARYA M G	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	5	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	8	3	Y	8	0	N	10	4	3	Y	6	3	Y	**31**	1	N
5	1KS19EC005	AKSHAY KUMAR D	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	5	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	12	1	0	N	11	3	Y	10	4	3	Y	6	3	Y	**37**	3	Y
6	1KS19EC006	AKSHITHA	26	17	3	Y	9	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	18	6	1	N	12	3	Y	10	4	3	Y	6	3	Y	**42**	3	Y
7	1KS19EC007	AMRUTA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	14	11	3	Y	3	0	N	10	4	3	Y	6	3	Y	**24**	0	N
8	1KS19EC008	AMULYA R	27	15	3	Y	12	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	21	10	3	Y	11	3	Y	10	4	3	Y	6	3	Y	**32**	1	N
9	1KS19EC009	ANITHA S	28	17	3	Y	11	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	30	12	3	Y	18	3	Y	10	4	3	Y	6	3	Y	**29**	0	N
10	1KS19EC010	ANJALI Y J	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	13	11	3	Y	2	0	N	6	2	0	N	4	3	Y	**30**	1	N
11	1KS19EC011	ARCHANA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	13	9	3	Y	4	0	N	6	2	0	N	4	3	Y	**32**	1	N
12	1KS19EC012	ASHRITHA	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	8	2	3	Y	4	3	Y	2	3	Y	8	4	0	N	4	0	N	10	4	3	Y	6	3	Y	**21**	0	N
13	1KS19EC013	BHARATH	21	14	3	Y	7	2	N	6	4	3	Y	2	1	N	28	6	3	Y	17	3	Y	5	3	Y	0	0	0	N	0	0	N	0	0	N	0	0	0	N	0	0	N	0	0	0	N	0	0	N	**A**	3	Y
14	1KS19EC014	BHAVANA	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	23	11	3	Y	12	3	Y	8	4	3	Y	4	3	Y	**42**	3	Y
15	1KS19EC015	CHAITRA P	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	21	10	3	Y	11	3	Y	6	2	0	N	4	3	Y	**35**	2	N
16	1KS19EC016	CHANDAN RALY	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	22	8	3	Y	14	3	Y	10	4	3	Y	6	3	Y	**30**	1	N
17	1KS19EC017	CHANDANA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	24	6	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	7	2	N	10	2	N	6	2	0	N	4	3	Y	**15**	0	N

18	1KS19EC018	CHENNREDDY RAJASEKHAR	22	12	3	Y	10	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	29	11	3	Y	18	3	Y	7	3	1	N	4	3	Y	**37**	3	Y
19	1KS19EC019	CHIRANTHANA YOGANANDA K	18	10	2	N	8	3	Y	10	6	3	Y	4	3	Y	24	5	3	Y	15	3	Y	4	3	Y	10	2	3	Y	6	3	Y	2	3	Y	5	5	0	N	0	0	N	8	3	1	N	5	3	Y	**22**	0	N
20	1KS19EC020	D NAYAN	26	16	3	Y	10	3	Y	9	5.5	3	Y	3.5	3	Y	23	5	3	Y	14	3	Y	4	3	Y	8	2	3	Y	1	0	N	5	3	Y	18	9	3	Y	9	0	N	6	2	0	N	4	3	Y	**30**	1	N
21	1KS19EC021	DANESH RAJU V	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	25	6	3	Y	14	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	20	10	3	Y	10	2	N	8	3	1	N	5	3	Y	**32**	1	N
22	1KS19EC022	DAVINO JO	25	15	3	Y	10	3	Y	10	6	3	Y	4	3	Y	23	6	3	Y	12	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	12	5	0	N	7	0	N	10	4	3	Y	6	3	Y	**24**	0	N
23	1KS19EC023	DHANYA SUKANTH B K	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	18	10	3	Y	8	0	N	10	4	3	Y	6	3	Y	**34**	2	N
24	1KS19EC024	DHEEMANTH K N	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	8	2	3	Y	4	3	Y	2	3	Y	10	10	3	Y	0	0	N	8	3	1	N	5	3	Y	**33**	2	N
25	1KS19EC025	DISHA SHIVANI	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	8	2	3	Y	4	3	Y	2	3	Y	23	10	3	Y	13	3	Y	10	4	3	Y	6	3	Y	**35**	2	N
26	1KS19EC027	GAYATHRI	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	24	10	3	Y	14	3	Y	6	2	0	N	4	3	Y	**26**	0	N
27	1KS19EC028	GAYATHRI	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	26	11	3	Y	15	3	Y	10	4	3	Y	6	3	Y	**24**	0	N
28	1KS19EC029	GONUGUNTLA SAI SIDDARTHA	27	18	3	Y	9	3	Y	9	5.5	3	Y	3.5	3	Y	23	5	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	11	8	3	Y	3	0	N	8	3	1	N	5	3	Y	**21**	0	N
29	1KS19EC030	GOWRI S NADIGER	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	24	6	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	9	3	Y	6	0	N	10	4	3	Y	6	3	Y	**27**	0	N
30	1KS19EC031	HARSHA R	22	13	3	Y	9	3	Y	8	5	3	Y	3	3	Y	23	6	3	Y	13	3	Y	4	3	Y	6	1	1	N	4	3	Y	1	1	N	0	0	0	N	0	0	N	10	4	3	Y	6	3	Y	**23**	0	N
31	1KS19EC032	HARSHITH A B Y	27	18	3	Y	9	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	18	2	0	N	16	3	Y	8	3	1	N	5	3	Y	**29**	0	N
32	1KS19EC033	HEMANTH.	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	15	3	Y	6	3	Y	7	1	1	N	4	3	Y	2	3	Y	15	5	0	N	10	2	N	6	2	0	N	4	3	Y	**30**	1	N
33	1KS19EC034	HIMA SWETHA S	30	18	3	Y	12	3	Y	9	5.5	3	Y	3.5	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	0	0	0	N	0	0	N	6	2	0	N	4	3	Y	**26**	0	N
34	1KS19EC035	JAGRUTI P.	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	27	11	3	Y	16	3	Y	10	4	3	Y	6	3	Y	**41**	3	Y
35	1KS19EC036	JAYANTH M B	29	17	3	Y	12	3	Y	6	4	3	Y	2	1	N	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	11	7	2	N	4	0	N	8	3	1	N	5	3	Y	**22**	0	N
36	1KS19EC037	KAMMA MANUBOL	27	18	3	Y	9	3	Y	9	5	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	27	10	3	Y	17	3	Y	9	4	3	Y	5	3	Y	**32**	1	N
37	1KS19EC038	KARTHIK K	22	17	3	Y	5	0	N	9	5.5	3	Y	3.5	3	Y	22	6	3	Y	12	3	Y	4	3	Y	10	2	3	Y	6	3	Y	2	3	Y	9	3	0	N	6	0	N	10	4	3	Y	6	3	Y	**37**	3	Y
38	1KS19EC039	KASHYAP. P	25	15	3	Y	10	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	8	2	3	Y	4	3	Y	2	3	Y	11	9	3	Y	2	0	N	6	2	0	N	4	3	Y	**21**	0	N
39	1KS19EC040	KRUPA.A	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	22	9	3	Y	13	3	Y	6	2	0	N	4	3	Y	**37**	3	Y
40	1KS19EC041	KRUTHI K S	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	25	11	3	Y	14	3	Y	10	4	3	Y	6	3	Y	**35**	2	N
41	1KS19EC042	LAKSHMAN	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	6	1	1	N	4	3	Y	1	1	N	10	3	0	N	7	0	N	6	2	0	N	4	3	Y	**23**	0	N
42	1KS19EC043	LIKITHA.H	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	8	6	1	N	2	0	N	8	3	1	N	5	3	Y	**26**	0	N
43	1KS19EC044	M LOKESH	27	17	3	Y	10	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	19	10	3	Y	9	0	N	8	3	1	N	5	3	Y	**41**	3	Y
44	1KS19EC045	MANU N KANDRA	27	18	3	Y	9	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	21	9	3	Y	12	3	Y	10	4	3	Y	6	3	Y	**38**	3	Y
45	1KS19EC046	MEGHANA	27	18	3	Y	9	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	8	3	Y	9	0	N	8	3	1	N	5	3	Y	**26**	0	N
46	1KS19EC047	MOHAMMA	28	17	3	Y	11	3	Y	5	3	1	N	2	1	N	20	4	3	Y	12	3	Y	4	3	Y	5	1	1	N	3	1	N	1	1	N	8	8	3	Y	0	0	N	5	2	0	N	3	3	Y	**22**	0	N
47	1KS19EC048	MOHITH KUMAR G	27	16	3	Y	11	3	Y	8	5	3	Y	3	3	Y	25	6	3	Y	14	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	4	0	0	N	4	0	N	10	4	3	Y	6	3	Y	**24**	0	N
48	1KS19EC049	MONIKA V	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	12	10	3	Y	2	0	N	6	2	0	N	4	3	Y	**21**	0	N
49	1KS19EC050	MONISHA. B K	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	20	7	2	N	13	3	Y	8	3	1	N	5	3	Y	**37**	3	Y
50	1KS19EC051	N ANILA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	4	0	N	12	3	Y	8	3	1	N	5	3	Y	**42**	3	Y

51	1KS19EC052	NIDHI S	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	0	0	0	N	0	0	N	6	2	0	N	4	3	Y	**27**	0	N
52	1KS19EC053	NISARGA K	28	17	3	Y	11	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	0	0	0	N	0	0	N	6	2	0	N	4	3	Y	**21**	0	N
53	1KS19EC054	NITHIN D	28	17	3	Y	11	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	12	5	0	N	7	0	N	8	3	1	N	5	3	Y	**25**	0	N
54	1KS19EC055	PAVAN KUMAR G R	26	17	3	Y	9	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	5	0	N	10	2	N	10	4	3	Y	6	3	Y	**40**	3	Y
55	1KS19EC056	POKURI MOUNIKA	25	17	3	Y	8	3	Y	10	6	3	Y	4	3	Y	23	6	3	Y	13	3	Y	4	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	10	3	Y	6	0	N	8	3	1	N	5	3	Y	**36**	3	Y
56	1KS19EC057	POOJA S P	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	9	3	Y	7	0	N	10	4	3	Y	6	3	Y	**23**	0	N
57	1KS19EC058	PRADEEP C	25	16	3	Y	9	3	Y	6	3.5	2	N	2.5	3	Y	27	6	3	Y	16	3	Y	5	3	Y	6	1	1	N	4	3	Y	1	1	N	8	0	0	N	8	0	N	6	2	0	N	4	3	Y	**24**	0	N
58	1KS19EC059	PRAKASH CHEGORE	24	16	3	Y	8	3	Y	8	5	3	Y	3	3	Y	27	6	3	Y	16	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	12	4	0	N	8	0	N	6	2	0	N	4	3	Y	**21**	0	N
59	1KS19EC061	PRASHANT	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	2	0	N	13	3	Y	8	3	1	N	5	3	Y	**36**	3	Y
60	1KS19EC062	PRAVEEN KUMAR N	29	18	3	Y	11	3	Y	9	5.5	3	Y	3.5	3	Y	27	6	3	Y	16	3	Y	5	3	Y	6	1	1	N	4	3	Y	1	1	N	9	6	1	N	3	0	N	6	2	0	N	4	3	Y	**18**	0	N
61	1KS19EC063	PREETHAM	29	17	3	Y	12	3	Y	7	4.5	3	Y	2.5	3	Y	24	6	3	Y	13	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	19	10	3	Y	9	0	N	6	2	0	N	4	3	Y	**31**	1	N
62	1KS19EC064	PRIYANKA	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	7	1	1	N	4	3	Y	2	3	Y	20	9	3	Y	11	3	Y	6	2	0	N	4	3	Y	**25**	0	N
63	1KS19EC065	RADHA KRISHNA L	26	17	3	Y	9	3	Y	10	6	3	Y	4	3	Y	22	6	3	Y	12	3	Y	4	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	8	3	Y	9	0	N	6	2	0	N	4	3	Y	**22**	0	N
64	1KS19EC066	RAJALAKS HMI S	25	18	3	Y	7	2	N	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	5	0	N	12	3	Y	10	4	3	Y	6	3	Y	**38**	3	Y
65	1KS19EC067	RAMYASR EE R	26	17	3	Y	9	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	7	2	0	N	5	0	N	8	3	1	N	5	3	Y	**21**	0	N
66	1KS19EC068	RANGASW AMY U	28	17	3	Y	11	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	18	9	3	Y	9	0	N	6	2	0	N	4	3	Y	**33**	2	N
67	1KS19EC069	ROHAN K R	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	9	9	3	Y	0	0	N	6	2	0	N	4	3	Y	**39**	3	Y
68	1KS19EC070	S K BHARA	22	14	3	Y	8	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	7	2	N	8	0	N	10	4	3	Y	6	3	Y	**24**	0	N
69	1KS19EC071	SABARISH	27	16	3	Y	11	3	Y	7	4.5	3	Y	2.5	3	Y	26	6	3	Y	15	3	Y	5	3	Y	8	1	1	N	6	3	Y	1	1	N	8	7	2	N	1	0	N	6	2	0	N	4	3	Y	**5**	0	N
70	1KS19EC072	SAHANA K	28	18	3	Y	10	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	22	10	3	Y	12	3	Y	8	3	1	N	5	3	Y	**35**	2	N
71	1KS19EC073	SAHANA S	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	13	5	0	N	8	0	N	6	2	0	N	4	3	Y	**35**	2	N
72	1KS19EC074	SAI PRIYA T S	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	23	11	3	Y	12	3	Y	10	4	3	Y	6	3	Y	**24**	0	N
73	1KS19EC075	SAMIKSHA	29	18	3	Y	11	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	6	1	1	N	4	3	Y	1	1	N	11	4	0	N	7	0	N	5	2	0	N	3	3	Y	**21**	0	N
74	1KS19EC076	SANTOSH I	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	15	10	3	Y	5	0	N	6	2	0	N	4	3	Y	**23**	0	N
75	1KS19EC077	SATHVIK U	19	12	3	Y	7	2	N	10	6	3	Y	4	3	Y	25	6	3	Y	14	3	Y	5	3	Y	5	1	1	N	3	1	N	1	1	N	10	5	0	N	5	0	N	6	2	0	N	4	3	Y	**27**	0	N
76	1KS19EC078	SHAMITH A BIJOOR	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	1	0	N	15	3	Y	8	3	1	N	5	3	Y	**29**	0	N
77	1KS19EC079	SHASHAN K KASHYAP. H.R	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	7	2	N	10	2	N	8	3	1	N	5	3	Y	**29**	0	N
78	1KS19EC081	SHREYAM S D K	26	16	3	Y	10	3	Y	7	4.5	3	Y	2.5	3	Y	24	6	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	3	0	N	13	3	Y	6	2	0	N	4	3	Y	**24**	0	N
79	1KS19EC082	SHREYAS I	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	19	7	2	N	12	3	Y	10	4	3	Y	6	3	Y	**25**	0	N
80	1KS19EC083	SHREYAS GOWDA	0	0	0	N	0	0	N	5	3	1	N	2	1	N	23	6	3	Y	12	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	14	9	3	Y	5	0	N	5	2	0	N	3	3	Y	**21**	0	N
81	1KS19EC084	SHREYAS V	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	27	6	3	Y	15	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	6	4	0	N	2	0	N	10	4	3	Y	6	3	Y	**26**	0	N

82	1KS19EC085	SHUBHAM KUMAR SINGH A	29	17	3	Y	12	3	Y	9	5.5	3	Y	3.5	3	Y	25	6	3	Y	14	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	14	9	3	Y	5	0	N	6	2	0	N	4	3	Y	**36**	3	Y
83	1KS19EC086	SINCHAN A M N	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	25	6	3	Y	14	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	11	1	0	N	10	2	N	8	3	1	N	5	3	Y	**34**	2	N
84	1KS19EC087	SRINIVAS S	28	17	3	Y	11	3	Y	10	6	3	Y	4	3	Y	25	6	3	Y	14	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	11	9	3	Y	2	0	N	6	2	0	N	4	3	Y	**33**	2	N
85	1KS19EC088	SRINIVASAN M	28	17	3	Y	11	3	Y	8	5	3	Y	3	3	Y	24	6	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	16	5	0	N	11	3	Y	8	3	1	N	5	3	Y	**38**	3	Y
86	1KS19EC089	SRIRAM	27	16	3	Y	11	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	1	0	0	N	1	0	N	10	4	3	Y	6	3	Y	**31**	1	N
87	1KS19EC090	SUHAS.M	28	17	3	Y	11	3	Y	7	4.5	3	Y	2.5	3	Y	26	6	3	Y	15	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	0	0	0	N	0	0	N	10	4	3	Y	6	3	Y	**23**	0	N
88	1KS19EC092	SUMUKHA VASISHTA M R	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	8	3	Y	7	0	N	6	2	0	N	4	3	Y	**21**	0	N
89	1KS19EC093	SUSHMITH	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	28	6	3	Y	16	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	14	5	0	N	9	0	N	8	3	1	N	5	3	Y	**36**	3	Y
90	1KS19EC094	SWAGATH	28	16	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	18	9	3	Y	9	0	N	10	4	3	Y	6	3	Y	**24**	0	N
91	1KS19EC095	SWATHI U	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	4	4	0	N	0	0	N	6	2	0	N	4	3	Y	**33**	2	N
92	1KS19EC096	T N L RUTHVIK	16	10	2	N	6	1	N	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	7	1	1	N	5	3	Y	1	1	N	14	10	3	Y	4	0	N	6	2	0	N	4	3	Y	**30**	1	N
93	1KS19EC097	TEJASHWINI P V	23	17	3	Y	6	1	N	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	25	8	3	Y	17	3	Y	10	4	3	Y	6	3	Y	**22**	0	N
94	1KS19EC098	THEERTHANA S R	28	16	3	Y	12	3	Y	10	6	3	Y	4	3	Y	30	6	3	Y	18	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	5	0	N	10	2	N	8	3	1	N	5	3	Y	**27**	0	N
95	1KS19EC099	TUSHAR R VASISHTA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	17	3	Y	6	3	Y	10	2	3	Y	6	3	Y	2	3	Y	15	5	0	N	10	2	N	10	4	3	Y	6	3	Y	**24**	0	N
96	1KS19EC100	VAISHNAVI K	24	17	3	Y	7	2	N	10	6	3	Y	4	3	Y	28	6	3	Y	17	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	5	5	0	N	0	0	N	10	4	3	Y	6	3	Y	**21**	0	N
97	1KS19EC101	VANDANA	28	18	3	Y	10	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	27	11	3	Y	16	3	Y	8	3	1	N	5	3	Y	**36**	3	Y
98	1KS19EC102	VANDANA	30	18	3	Y	12	3	Y	10	6	3	Y	4	3	Y	29	6	3	Y	18	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	7	7	2	N	0	0	N	6	2	0	N	4	3	Y	**27**	0	N
99	1KS19EC103	VIGNESH M	20	14	3	Y	6	1	N	10	6	3	Y	4	3	Y	27	6	3	Y	16	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	17	8	3	Y	9	0	N	8	3	1	N	5	3	Y	**24**	0	N
100	1KS19EC104	VIKAS S	24	16	3	Y	8	3	Y	8	5	3	Y	3	3	Y	27	6	3	Y	16	3	Y	5	3	Y	8	2	3	Y	4	3	Y	2	3	Y	7	1	0	N	6	0	N	10	4	3	Y	6	3	Y	**15**	0	N
101	1KS19EC105	VINUTH S I	25	16	3	Y	9	3	Y	6	4.5	3	Y	1.5	0	N	26	6	3	Y	15	3	Y	5	3	Y	6	1	1	N	4	3	Y	1	1	N	7	5	0	N	2	0	N	6	2	0	N	4	3	Y	**5**	0	N
102	1KS19EC106	VISHAL SANJAY RAJU	29	17	3	Y	12	3	Y	10	6	3	Y	4	3	Y	26	6	3	Y	15	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	13	10	3	Y	3	0	N	5	2	0	N	3	3	Y	**21**	0	N
103	1KS19EC107	VISHNU RAATA YADUNANDAN	28	17	3	Y	11	3	Y	8	5	3	Y	3	3	Y	25	6	3	Y	14	3	Y	5	3	Y	8	2	3	Y	4	3	Y	2	3	Y	7	3	0	N	4	0	N	8	3	1	N	5	3	Y	**30**	1	N
104	1KS19EC108	YASHASWINI N	24	15	3	Y	9	3	Y	10	6	3	Y	4	3	Y	24	6	3	Y	13	3	Y	5	3	Y	10	2	3	Y	6	3	Y	2	3	Y	3	3	0	N	0	0	N	6	2	0	N	4	3	Y	**29**	0	N
105	1KS20EC400	MADALA VIVEK KUMAR	0	0	0	N	0	0	N	9	5.5	3	Y	3.5	3	Y	10	2	0	N	7	0	N	1	0	N	9	2	3	Y	6	3	Y	1	1	N	0	0	0	N	0	0	N	8	3	1	N	5	3	Y	**21**	0	N
106	1KS20EC401	RANJANA P	0	0	0	N	0	0	N	9	5.5	3	Y	3.5	3	Y	20	4	3	Y	12	3	Y	4	3	Y	9	2	3	Y	6	3	Y	1	1	N	0	0	0	N	0	0	N	9	3	1	N	6	3	Y	**21**	0	N
107	1KS20EC402	SINDHU J	0	0	0	N	0	0	N	9	5.5	3	Y	3.5	3	Y	18	4	3	Y	11	3	Y	3	1	N	9	2	3	Y	6	3	Y	1	1	N	0	0	0	N	0	0	N	8	3	1	N	5	3	Y	**24**	0	N
	CO				**CO1**			**CO2**				**CO1**			**CO2**				**CO2**			**CO3**			**CO4**				**CO2**			**CO3**			**CO4**				**CO4**			**CO5**				**CO4**			**CO5**			**SEE**	
	umber of Not Attempted(N					**0**			**0**				**0**			**0**				**0**			**0**			**0**				**0**			**0**			**0**				**0**			**0**				**0**			**0**			**0**
	Score index & No of Y's				**3**	**101**		**3**	**94**			**3**	**103**		**3**	**101**			**3**	**78**		**3**	**106**			**105**			**3**	**85**		**3**	**103**		**2.6**	**84**			**1.6**	**51**		**1.0**	**31**			**1.3**	**37**		**3**	**106**		**1**	**21**
	No. of N's					**6**			**13**				**4**			**6**				**0**			**1**			**2**				**22**			**4**			**23**				**56**			**76**				**70**			**1**			**86**
	CO Attainment					**94.4**			**88**				**96.26**			**94.4**				**100**			**99.1**			**98.1**				**79**			**96.3**			**79**				**48**			**29**				**35**			**99.1**			**20**
	Level					**3**			**3**				**3**			**3**				**3**			**3**							**3**			**3**							**0**			**0**				**0**			**3**			**0**

CO	CIE	SEE	Direct Attainment	Level	INDIRECT ATTAIN	Final Att
CO1	95.33	20	57.48	2.0	3	2.1
CO2	90.42	20	55.02	2.0	3	2.1
CO3	97.66	20	58.64	2.0	3	2.1
CO4	64.72	20	42.17	0.0	3	0.3
CO5	64.02	20	41.82	0.0	3	0.3
AVERAGE						1.4

CO	Score index out of 3
CO1	1.89
CO2	2.21
CO3	1.94
CO4	1.26
CO5	0.97

					Co-Po Mapping Table									
CO'S	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	_	_	_	_	_	_	_	_	1	2	_
CO2	3	2	1	_	_	_	_	_	_	_	_	1	2	1
CO3	3	2	1	_	_	_	_	_	_	_	_	1	_	_
CO4	3	2	1	_	_	_	_	_	_	_	_	1	_	_
CO5	3	2	1	_	_	_	_	_	_	_	_	1	_	_
AVG	3.00	2.0	1.0	_	_	_	_	_	_	_	_	1.0	2.0	1.0

PO Attainment

CO'S	CO Attainment	CO RESULT	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2.1	Y	2.1	1.4	0.7	_	_	_	_	_	_	_	_	0.7	1.4	_
2	2.1	Y	2.1	1.4	0.7	_	_	_	_	_	_	_	_	0.7	1.4	0.7
3	2.1	Y	2.1	1.4	0.7	_	_	_	_	_	_	_	_	0.7	_	_
4	0.3	N	0.3	0.2	0.1	_	_	_	_	_	_	_	_	0.1	_	_
5	0.3	N	0.3	0.2	0.1	_	_	_	_	_	_	_	_	0.1	_	_
Avera			**1.4**	**0.9**	**0.5**	_	_	_	_	_	_	_	_	**0.5**	**1.4**	**0.7**

Teaching Reference Notes

18EC35-COMPUTER ORGANIZATION & ARCHITECTURE

Third Semester E & CE

By

Dr. B Sudarshan

18EC

Chapter 1 BASIC STRUCTURE OF COMPUTERS

This chapter introduces basics of computer hardware and software, presents some common terminology and gives broad overview of the computer organization.

Learning objective of this chapter.

- Explain the basic sub systems of a computer, their organization, structure and operation.

At the end of this chapter, you will be able to learn following.

- Computer types & functional units of computer.
- Basic operational concepts of computer & performance parameters.

1.1 COMPUTER TYPES

Definition of a Computer: It is a fast electronic calculating machine that accepts digitized input information, processes it according to a list of internally stored instructions and produces the resulting output information.

The list of instructions is called a **computer program**. The internal storage is called a **computer memory**.
Following are various types of computers based on their size, cost, performance and applications.

1. **Personal Computers**: It is the most common form of Desktop Computers These computers are commonly used in homes, schools and business offices. They have
    a. Processing and storage units.
    b. Visual display and audio output units
    c. Keyboard

    All these units can easily be placed on a home/office desk. Storage media includes hard disks, CD-ROMs, diskettes.

2. **Notebook Computers**: They are of the size of the personal computer with all the above mentioned components packaged into a single unit which is of the size of a thin briefcase. These are popularly known as **Laptops.**

3. **Workstations**: They have the dimensions of desktop computers but have more computational power than personal computers. They have high resolution graphics input/output capability. They are widely used in engineering applications involving

interactive design works like automotive design using CAD/CAM etc.

4. **Enterprise Systems**: They are also called as **Mainframes**. They are used for business data processing in medium to large corporations that require much more computing power and storage capacity than workstations. Enterprise systems are referred to as **servers** at low end and **supercomputers** at high end.
   a. **Servers**: These are used for business data processing in medium to large corporations that require much more computing power and storage capacity than workstations can provide. These contain sizeable database storage units and capable of handling large volumes of requests to access the data. In many cases, servers are widely accessible to the education, business and personal user communities. The servers use internet facility for communication to happen for servicing the requests.
   b. **Supercomputers**: These contain multiple processors with parallel processing capabilities with high speed computation capabilities. They are used in areas like weather forecasting, avionics, aircraft design and testing and military applications which are used for large scale numerical calculations.

1.2 FUNCTIONAL UNITS

The computer consists of five main parts which are functionally independent:

1. **Input Unit**: Computer accepts coded information through input unit. The most commonly used input device is keyboard. When a key is pressed, corresponding digit/letter is translated into binary (ASCII) code and transmitted over the cable to either memory or to the processor. Other input devices are joysticks, mouse, microphones etc.
2. **Memory Unit**: The function of the memory unit is to store programs and data. There are two classes of storage::
   a. **Primary Storage**: It is a fast memory that operates at electronic speeds. The memory consists of large number of semiconductor storage cells capable of storing one bit of information. Group of 8 bits is called a **byte**, group of 16 bits is called a **word**, and group of 32 bits is called a **double word**. To provide easy access to any word in the memory, a distinct address is associated with each word locations. A given word is accessed by specifying its address and issuing a control command that starts the storage and retrieval process. The number of bits in each word is referred to as the **word length**. Small machines have only a few tens of millions of words whereas a medium or large machine has many tens or hundreds of millions of words. When the memory is accessed, only one word of data is read or written. Program must reside in memory during execution. Memory in which any location can be accessed in a short and fixed amount of time after specifying its address is called **Random Access Memory (RAM)**. The time required to access one word is called **memory access time**. The memory of a computer is implemented as a memory hierarchy of 3 or 4 levels of semiconductor RAM units with different speeds and sizes. The small, fast RAM units are called **caches**. The largest and slowest unit is referred to as **main memory**.

b. **Secondary Storage**: These include magnetic disks and tapes, optical disks (CD-ROMs). These are used when large amounts of data have to be stored particularly for information that is accessed infrequently.

3. **Arithmetic and Logic Unit**: Most computer operations are executed in ALU. The operands required to perform arithmetic and logical operations are brought into the processor and stored in high speed storage elements called **registers**.
4. **Output Unit**: The function of output unit is to send processed results to the outside world. Some examples of output units are printers, graphic displays, monitor, plotter etc. Printers employ mechanical impact heads, ink jet streams or photocopying techniques to perform printing. Graphic displays provide both input and output functions. Hence sometimes they are referred to as I/O units.
5. **Control Unit**: It is a unit that co-ordinates the operations of all the other units. It is the nerve centre that sends control signals to other units and senses their states. I/O transfers are controlled by programs that identify the devices involved in information to be transferred.
    a. Control unit governs the transfer using **timing signal**. Timing signal determines when a given action is to take place.
    b. Data transfer between processor and memory are also controlled by control unit through timing signals.

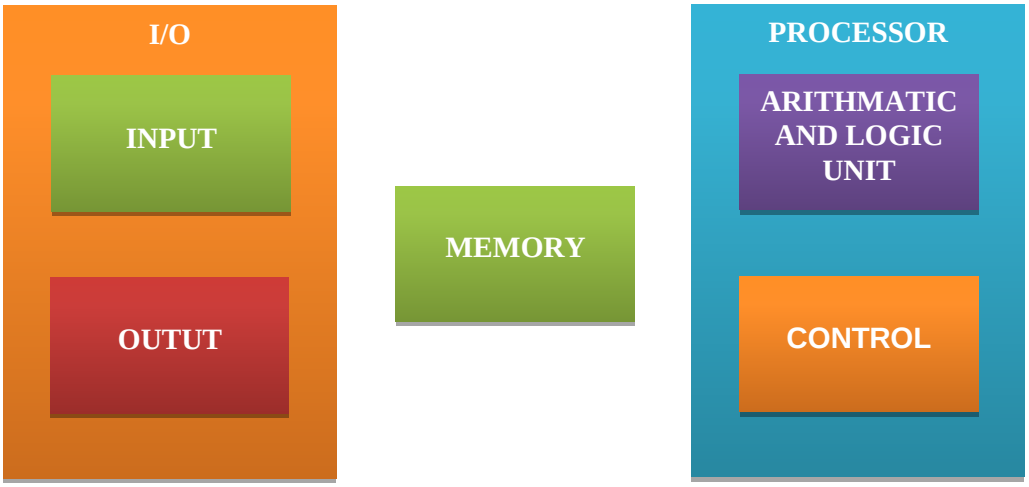


Figure 1.1 Basic functional units of a computer

1.3 BASIC OPERATIONAL CONCEPTS

Computer performs a particular task governed by list of instructions written by the user. This list of instructions (a program) is stored in memory. Individual instructions are brought from memory in to the processor which executes specified instructions. Each instruction generally consists of two parts; first part as the **opcode,** which specifies the operation to be performed by the processor and second as the **operand** on which the operation needs to be performed. Operands are also stored in memory.

A typical instruction may look like

ADD LOCA, R0 Where **ADD** is the operation code which performs addition, **LOCA** is the location of the first operand in memory, **R0** is a register which holds the second operand.

This instruction adds contents of memory location **LOCA** to the contents of register **R0** and stores the result into register **R0**. The original contents of the memory location-**LOCA** are preserved while that of register **R0** is overwritten with the result of addition.

Execution of above instruction may require several steps as below.

1. Fetch the instruction from the main memory into the processor.
2. Operand at memory location **LOCA** is fetched
3. This operand is added to the contents of the register **R0**
4. The result of addition is stored in the register **R0**.

The ADD instruction above combines memory access operation with an ALU operation. To improve the performance of the processor, these two types of operations can be performed using two separate instructions below.

LOAD LOCA, R1 // loads the operand located at LOCA into register **R1**

ADD R1,R0 // Adds the contents of register **R1** to register **R0** and stores the result into register **R0.**

The original content of **LOCA** is preserved.

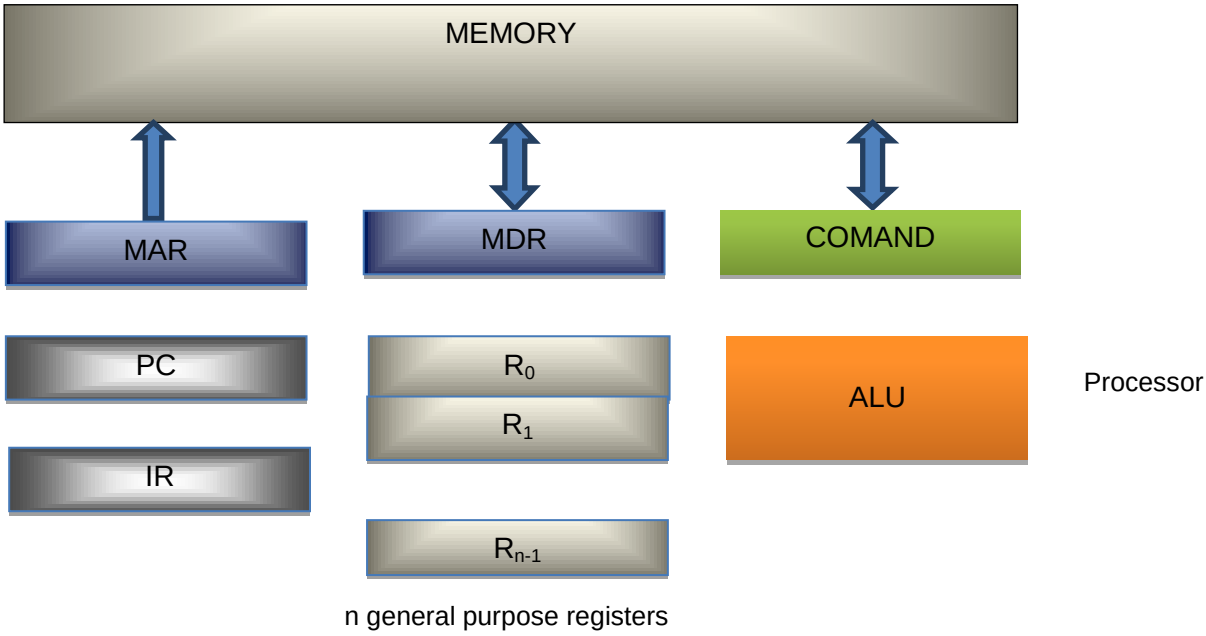


Figure 1.2 Memory and processor

Figure 1.2 shows connection between memory and the processor. Transfers between memory and the processor are started by sending the address of the memory location to be accessed to the memory unit and issuing the appropriate control signals. The data are then transferred to or from the memory.

The processor contains **ALU**, control circuitry and a number of registers used for general purpose operations. The Instruction register (**IR**) holds the instruction that is currently being executed. Its output is available to the control circuitry which generates required control signals that control the various processing elements involved in the execution of instruction.

The Program Counter (**PC**) is another special register which holds the memory address of the next instruction to be executed. This register keeps track on execution of program. During program execution, PC always points to the next instruction to be fetched from memory. There are several general purpose registers **R_0 to R_{n-1}**

Two registers **MAR** and **MDR** facilitate communication between processor and memory. **MAR** holds the address of the memory location that need to be accessed for read/write operation. **MDR** holds the data read from memory during read operation and data to be written to memory during write operation.

Program is entered into memory through **input** unit (e.g. Keyboard) and the program execution starts with **PC** pointing to the starting address of the program. The contents of the **PC** are transferred to the **MAR** and read control signal is sent by the **control** unit to memory. The data read from memory is loaded into **MDR**. The contents of **MDR** are transferred to the **IR** register to enable instruction decoding and execution by the processor.

If the instruction involves ALU operation, it is required to obtain the operands required by the operation. These operands may reside in memory or **GPR's**. if the operands are in memory, these have to be fetched from memory by issuing read command to memory as discussed earlier. Once all the operands are read, **ALU** performs the operation specified by the instruction on these operands and appropriately stores the result of operation into memory or GPR. If the result is to be stored into memory then, the control circuitry issues write command to memory along with the location address in **MAR** and data to be written into **MDR**.

At some point during the execution of the instruction, the **PC** should point to the next instruction to continue execution of the program.

Computers can also send data to some output devices and read data from the input devices for processing. Thus, some instructions should be there to perform these Input/output operations.

Normal execution of program may be interrupted if some device requires the use of processor immediately. For example, in an industrial temperature control application, if the temperature of some device increases beyond threshold value, the device raises an interrupt signal to processor to interrupt the currently executing program and run the interrupt service routine to turn **ON** the air conditioner. Once the service routine is executed, the processor resumes the earlier program. To do this, before starting to execute the interrupt service routine, the processor state (**PC**, **GPR** values) are stored so that when the program completes execution of the service routine, the saved process state is loaded again to resume the main program.

1.4 BUS STRUCTURES

For a computer to achieve its operation, the functional units need to communicate with each other. In order to communicate, they need to be connected. Functional units may be connected by a group of parallel wires. A group of lines that serves a connecting path for several devices is called a bus. Each wire in a bus can transfer one bit of information. The number of parallel wires in a bus is equal to the word length of a computer. In addition to the lines that carry the data, the bus must have lines for address and control purposes. The simplest way to interconnect functional units is to use a single bus, as shown below.

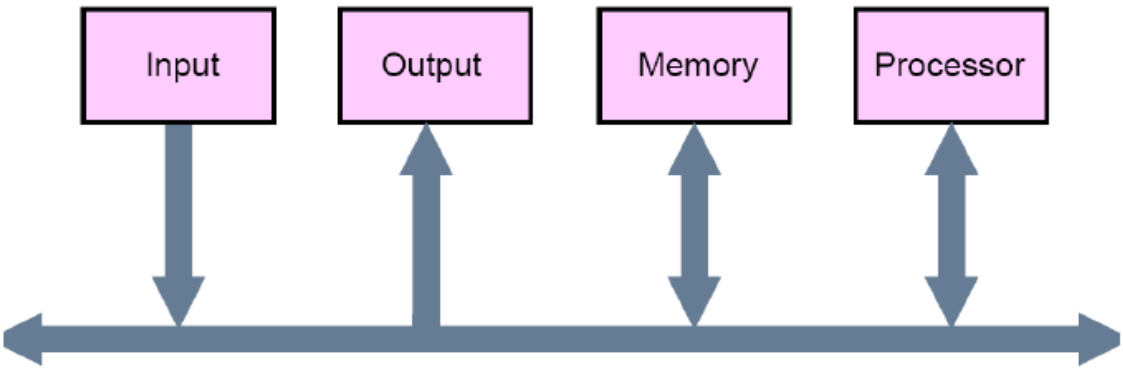


Figure 1.3 Memory and processor

The devices connected to a bus vary widely in their speed of Operation. Some devices are relatively slow, such as printer and keyboard. Some devices are considerably fast, such as optical disks, Memory and processor units operate are the fastest parts of a computer. Efficient transfer mechanism thus is needed to cope with this problem. A common approach is to include buffer registers with the devices to hold the information during transfers.

Another approach is to use two-bus structure and an additional transfer mechanism. A high-performance bus, a low-performance, and a bridge for transferring the data between the two buses. ARMA Bus belongs to this structure.

1.5 Software

System Software is a collection of programs that are executed as needed to perform functions such as

1. Receiving and interpreting user commands.
2. Entering and editing application programs, storing them as files in secondary storage devices.
3. Managing the storage and retrieval of files on secondary storage devices.
4. Running standard application programs such as word processors, spread sheets or games with data supplied by the user.
5. Controlling I/O units to receive input information and produce output results.

Compiler: Application Programs are usually written in high level programming language such as C, C++, and Java etc. Compiler is a system software program that translates high level language program into suitable machine language program.

Text Editor: It is used for entering and editing application programs.

File: It is a sequence of alphanumeric characters or binary data that is stored in memory or in secondary storage.

Operating System: An OS is a large program that is used to control the sharing of and interaction among various computer units as they execute application programs.

Let us consider an example of how an OS manages one application program. Let us consider a system with one processor, one disk and one printer. Assume that the application program has been compiled from a high level language form into a machine language form and stored in a disk.

1. First transfer this file into memory.
2. When the transfer is complete, execution of the program is started. (Assume that part of the program's task involves reading a data file from the disk into the memory, performing some computation on the data and printing the results).
3. When data file is needed, the program requests the OS to transfer the data file from disk to the memory.
4. The OS performs this task and passes execution control back to the application program.

5. When the results are ready to be printed after computation, the application program again sends a request to OS.
6. An OS routine is then executed to cause the printer to print the results.

The time line diagram shows user program and OS routine sharing of the processor.

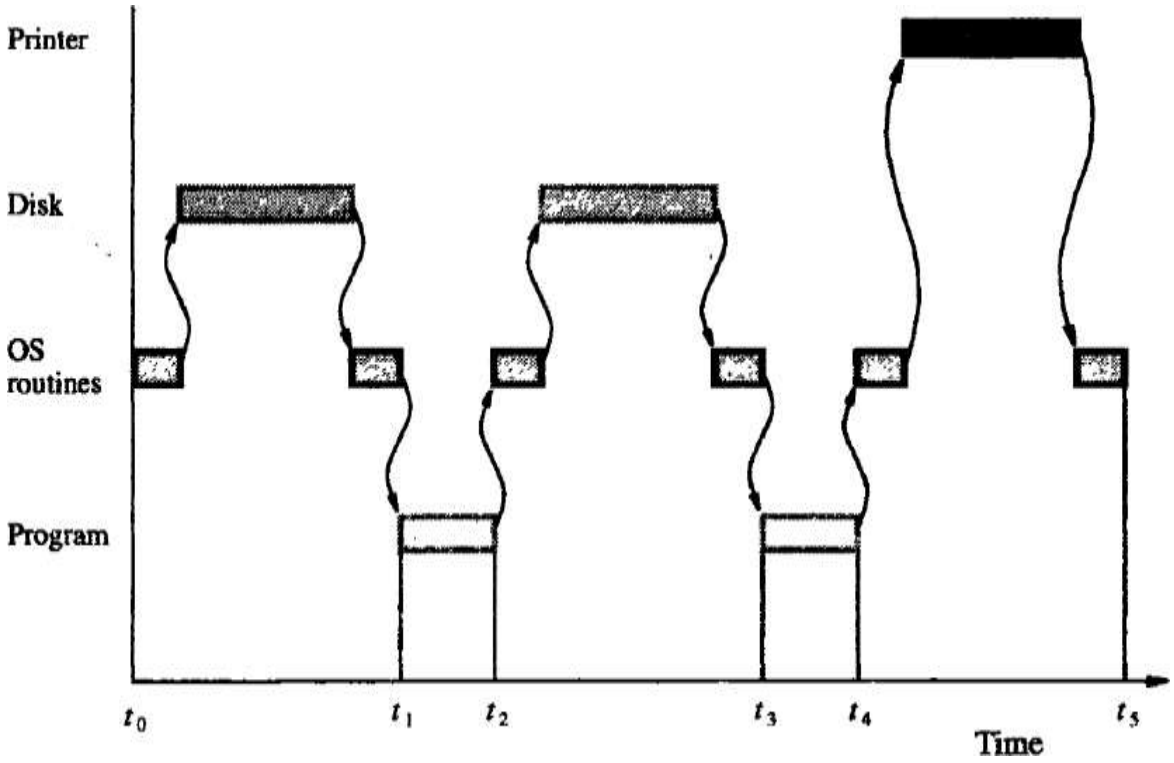


Figure 1.4 User program and OS sharing the processor

During the time period t_0 to t_1, the OS routine initiates loading the application program from disk to memory, waits until the transfer is completed and then passes execution control to the application program. A similar pattern of activity occurs during the period t_0 to t_3 and t_4 to t_5, when the OS transfers the data file from the disk and prints the results. At t_5, the OS may load and execute another application program

We can see that the disk and processor are idle between t_4 and t_5. So, next application program can be loaded during this time itself. Similarly the OS can arrange to print the previous results between t_0 and t_1. This pattern of concurrent execution is called **multiprogramming** or **multitasking**.

Processor Time: It is defined as the sum of the periods needed to execute the program (i.e. when processor is active labeled as Program and OS routine in the above figure).

Elapsed time: Total time needed to execute the program ($t_5 - t_0$).

1.6 Performance

The most important measure of the performance of a computer is how quickly it can execute

the program. The speed of the computer is affected by the design of

1. Instruction set
2. Hardware and the technology in which the hardware is implemented
3. Software including the OS

Because programs are usually written in a high-level language, performance is also affected by the compiler that translates programs into machine languages.
For best performance, it is necessary to design the compiler, Instruction set and the hardware in a coordinated way.
The processor time depends on the hardware involved in the execution of individual machine instructions. This hardware comprises a processor and the memory which are usually connected by a bus as shown in figure 1.5. This includes the cache memory as part of the processor unit. Let us examine the flow of program instructions and the data between the memory and the processor. At the start of the execution, all program instructions and the required data are stored in the main memory. As execution proceeds, instructions are fetched one by one over the bus into the processor and a copy is placed in the cache. When the execution of an instruction calls for data located in the main memory, the data are fetched and a copy of it is placed in the cache. Later, if the same instruction or the data is needed a second time, it is used directly from the cache.

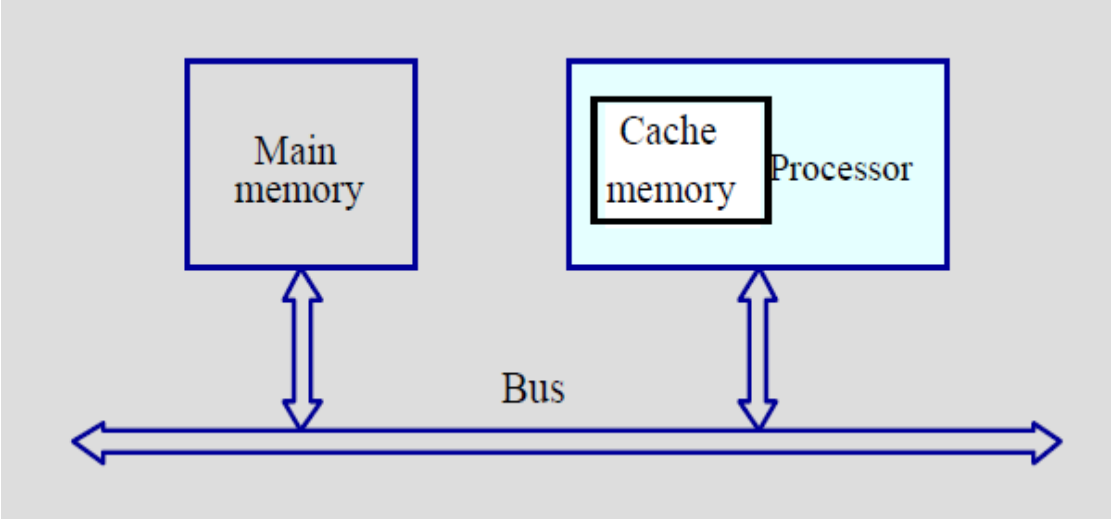


Figure 1.5 the Processor Cache

Processor circuits are controlled by a timing signal called a clock. The clock defines regular time intervals, called clock cycles. To execute a machine instruction, the processor divides the action to be performed into a sequence of basic steps, such that each step can be completed in one clock cycle. Let the clock period P of one clock cycle, its inverse is the clock rate, R=1/P.

Basic performance equation

T – Processor time required to execute a program that has been prepared in high-level language

N – Number of actual machine language instructions needed to complete the execution (note: loop)

S – Average number of basic steps needed to execute one machine instruction. Each step completes in one clock cycle

R – Clock rate

The program execution time T is given by

$$T = (NxS)/R$$

Parameters, N, S & R are dependent on each other.

To achieve high performance, the computer designer must reduce the value of T which means reducing N & S, increasing R.

The value of N is reduced if the source program is compiled into fewer machine instructions.

The value of S is reduced in instruction have a smaller number of basic steps to perform.

The value of R can be increased by using a higher frequency clock.

Care has to be taken while modifying values since changes in the parameter may affect the other.

Clock Rate

There are two possibilities for increasing the clock rate R.

1. Improving the IC technology makes logic circuits faster.

This reduces the time needed to compute a basic step and allow the clock period P to be reduced and the clock rate R increased.

2. Reducing the amount of processing done in one basic step also reduces the clock period P.

In the presence of a cache, the percentage of accesses to the main memory is small. Hence, much of the performance gain expected from the use of faster technology can be realized. The value of T will be reduced by the same factor as R is increased, i.e., S & R are not affected.