MUFFAKHAM JAH COLLEGE OF ENGINEERING AND TECHNOLOGY

Department of Electronics and communication Engineering

COURSE FILE

Course Name: Computer Organization and Architecture

Course code: PC234 EC

Semester: IV

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

COURSE FILE CONTENTS

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

CERTIFICATE

I, **J K Nag** have completed the following theory course allotted to me:

Sl.No	Semester	Name of the subject	Course ID	Total no. of units
1.	IV	**Computer Organization and Architecture**	**PC234 EC**	5

Date: 22-09-2021

Academic year: 2020-2021

Prepared by (Course coordinator): JAIDEEP KUMAR NAG
(Name and Signature)

Head of the Department:
(Verifying authority)

Head of Department
Electronics & Communication Engineering
Muffakham Jah College of Engg. & Tech.
Road No: 3, Banjara Hills Hyderabad-34

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

DEPARTMENT VISION AND MISSION

VISION

To be recognized as a premier education center providing state of art education and facilitating research and innovation in the field of Electronics and Communication Engineering.

MISSION

We are dedicated to providing high quality holistic education in Electronics and Communication Engineering that prepares the students for successful pursuit of higher education and challenging careers in industry, R&D and academics.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Course description:

This course is primarily meant to teach undergraduate students the basic operations of computing hardware and how it interfaces to software. It would provide the students the understanding of system-level programming and provide a high-level understanding of the role played by the central processing unit, instruction sets, and hardware associated with it. Students would know how to represent fixed-point and floating point numbers in computer and develop hardware algorithms using them for fixed-point and floating point arithmetic. The course would display understanding of pipelining concepts with parallelism and develop understanding of how memory is organized and managed in a modern digital computer, including cache, virtual and physical memory. It discusses input-output units and how they communicate with the processor, and how their performance is computed.

Course Overview:

This course on Computer Organization and Architecture will enable students to:

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 Pre-requisites:

ES216EC "Digital Electronics" (III SEM ECE)

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Marks Distribution:

CIE: 30 Marks (Max)
Comprising of Internal Test 1 & Internal Test 2 for 10 marks each plus 10 Marks assigned for Assignments/Quiz/Slip test

SEE: 70 Marks (Max) – O.U Pattern (Consisting of PART – A comprising of short type compulsory questions and PART – B consisting of long type questions having choice)

POs and PSOs

Establish the correlation between the courses and program outcomes (POs) and Program Specific Outcomes (PSOs)

23/07/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Program Outcomes (POs)

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

PO2: Problem analysis: Identify, formulate, research literature, and analyse complex engineering 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 and design 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 research methods 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 modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, 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 solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034

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

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

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community 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 the engineering 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.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1: VLSI Design: The ECE Graduates will acquire state of art analysis and design skills in the area of digital and analog VLSI design using modern CAD tools.

PSO2: Embedded System Design: The ECE Graduates will develop preliminary skills and capabilities necessary for embedded system design and demonstrate understanding of its societal impact.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

PSO3: Communication and Signal Processing: The ECE Graduates will obtain the knowledge of the working principles of modern communication systems and be able to develop simulation models of components of a communication system.

PSO4: Software Design Engineering: The ECE Graduates will develop soft skills, aptitude and programming skills to be employable in IT sector.

Course Outcomes (COs):

On successful completion of the above course, the student shall be able to

1. Relate the number representation of digital computers to devise the Hardware arithmetic algorithms for fixed and floating point operations.
2. Apply the knowledge of Bus structure, registers, micro-operations, control unit design using Hardwired and Micro-programmed organization.
3. Comprehend various Instructions and Addressing modes of CPU; Implement the various techniques of pipelining and parallelism in stored program structures, to analyze the processor performance.
4. Interpret different ways of processor communication with I/O devices and peripherals with their relative merits and demerits.
5. Conceptualize the memory hierarchy of a typical computer system and interpret various schemes for efficient memory utilization and performance improvements.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

CO mapping with POs and PSOs: CO - PO mapping

Courses	Program Outcomes											
	PO 1	PO 2	PO 3	Po 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
PC 234 EC – **Computer Organization and Architecture** Second Year Semester IV												
PC234EC.1: Relate the number representation of digital computer to devise the Hardware arithmetic algorithms for fixed and floating point operations.	**3**	2	-	-	-	-	-	-	-	-	-	-
PC234EC.2: Apply the knowledge of Bus Structure, Registers, Micro-operations, Control unit design using Hardwired and Micro-programmed organization.	2	2	2	-	-	-	-	-	-	-	-	-
PC234EC.3: Comprehend various Instructions, addressing modes of CPU; implement the various techniques of pipelining and parallelism in stored program structures to analyze the performance of a processor.	-	**3**	-	-	-	-	-	-	-	-	-	-
PC234EC.4: Interpret different ways of processor communication with I/O devices and peripherals with their relative merits and demerits.	-	2	**3**	-	-	-	-	-	-	-	-	-
PC234EC.5: Conceptualize the memory hierarchy of a typical computer system and interpret various schemes for efficient memory utilization and performance improvements.	-	2	-	-	-	-	-	-	-	-	-	-

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Justification of Mapping between course and Program Outcomes

The subject deals with the study of various data representations in computer and the digital algorithms required for implementing arithmetic and logical operations in a basic computer, which requires knowledge of mathematics fundamentals. They are further required in designing of digital circuit elements that form a basic part of a basic computer. Hence CO1 strongly maps with PO1.

Studying of the remaining functional blocks of a computer, processing of data in the CPU and various interfacing methods for achieving performance in the system forms a major part of the subject. This part moderately maps with PO2 and PO3.

Justification of Mapping with course outcomes and Program Outcomes:

PC234EC.1

CO1 strongly maps with PO1, as it describes and explains data representations, arithmetic and logical operations for basic functional blocks of a basic computer. CO1 moderately maps with PO2 as it deals with analyzing computer arithmetic, identify the various data inputs to computer and formulating to process the data.

PC234EC.2

CO2 moderately maps to PO1 in applying knowledge of digital circuits to design a basic computer.

It moderately maps with PO2 to identify and analyze the complex control organization and its implementation.

CO2 also moderately maps with PO3, as it deals with design of Hardwired and Micro-programmed control unit.

PC234EC.3

CO3 strongly maps with PO2, as this unit identifies the key attributes of CPU and explains various processing techniques with their merits and limitations.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P.

PC234EC.4

CO4 moderately maps with PO2 as it deals with analyzing different ways of communicating with input-output devices.

CO4 strongly maps with PO3, as it deals with communication with the external environment for its processing in the computer.

PC234EC.5

CO5 moderately maps wit PO2, as it deals with memory organization including cache and virtual memories along with their mappings and different ways of storing data in memory system.

CO-PSO Mapping

Courses	Program Specific Outcomes			
	PSO1	PSO2	PSO3	PSO4
PC 234 EC – **Computer Organization and Architecture** Second Year Semester IV				
PC234EC.1: Apply the knowledge of Number representation of digital computer to devise the Hardware arithmetic algorithms for fixed and floating point operations.	0	2	0	0
PC234EC.2: Apply the knowledge of Bus Structure, Registers, Micro-operations, Control unit design using Hardwired and Micro-programmed organization.	0	1	0	0
PC234EC.3: Comprehend various Instructions, addressing modes and implement the techniques of pipelining and parallelism in stored program structures to analyze the performance of a processor.	0	0	0	0
PC234EC.4: Interpret different ways of communicating with I/O devices and peripherals with their relative merits and demerits.	0	2	1	0
PC234EC.5: Conceptualize the memory hierarchy of a typical computer system and interpret various schemes for efficient memory utilization and performance improvements.	0	1	0	0

23/07/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Justification of Mapping between course and Program Specific Outcome:

The course on Computer Organization and Architecture deals with the design and implementation of computer arithmetic algorithms and key attributes of CPU which is prerequisite for designing of an embedded system and hence it is moderately mapped to PSO2.

The course also deals with input-output organization along with their interfaces and communication with basic computer, and hence is weakly mapped to PSO3.

CO1 deals with the design and implementation of computer arithmetic and hence is moderately mapped to PSO2. It is not relevant to other PSOs.

CO2 deals with design of basic computer and its internal operations, and hence weakly mapped to PSO2.

CO4 deals with Input-output organization and other part of this unit deals with input output interfaces and communication, and hence is moderately mapped to PS02 and weakly mapped to PS03

Syllabus, Textbooks and Reference Books

Course Code	Course Title						Core/Elective
PC234EC	**Computer Organisation and Architecture**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	-	-	-	**30**	**70**	**3**

Course Objectives

- Implement the fixed-point and floating-point addition, subtraction, multiplication & Division.

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, HYDERABAD - 500 034. A.P

- Describe the basic structure and operation of a digital computer.
- Discuss the different ways of communicating with I/O devices and standard I/O interfaces.
- Analyze the hierarchical memory system including cache memories and virtual memory.
- Understand issues affecting modern processors.

Course Outcomes

1. Perform mathematical operations on fixed and floating point digital data.
2. Illustrate the operation of a digital computer.
3. Understand I/O interfacing of a computer.
4. Interface microprocessor with memory devices.
5. Understand latest trends in microprocessors.

UNIT-I

Data representation and Computer arithmetic: Introduction to Computer Systems, Organization and architecture, evolution and computer generations; Fixed point representation of numbers, digital arithmetic algorithms for Addition, Subtraction, Multiplication using Booth's algorithm and Division using restoring and non-restoring algorithms. Floating point representation with IEEE standards and its arithmetic operations.

UNIT-II

Basic Computer organization and Design: Instruction codes, stored program organization, computer registers and common bus system, computer instructions, timing and control, instruction cycle: Fetch and Decode, Register reference instructions; Memory reference instructions. Input, output and Interrupt: configuration, instructions, Program interrupt, Interrupt cycle, Micro programmed Control organization, address sequencing, micro instruction format and micro program sequencer.

UNIT-III

Central Processing Unit: General register organization, stack organization, instruction formats, addressing modes, Data transfer and manipulation, Program control. CISC and RISC: features and comparison. Pipeline and vector Processing, Parallel Processing, Pipelining, Instruction Pipeline, Basics of vector processing and Array Processors.

UNIT-IV

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Input-output Organization: I/O interface. I/O Bus and interface modules, I/O versus

sfer: Strobe Control Methods Handshaking, Asynchronous serial

transfer. Modes of Transfer: Programmed I/O, Interrupt driven I/O, Priority interrupt; Daisy Chaining, Parallel Priority interrupt.Direct memory Access, DMA controller and transfer. Input Output Processor, CPU-IOP communication, I/O channel.

UNIT-V

Memory Organization: Memory hierarchy, Primary memory, Auxiliary memory, Associative memory,Cache memory: mapping functions, Virtual memory: address mapping using pages, Memory management.

Suggested Readings:

1. Morris Mano, M., "Computer System Architecture," 3/e, Pearson Education, 2005.
2. William Stallings, "Computer Organization and Architecture: Designing for performance," 7/e,Pearson Education, 2006.
3. John P. Hayes, "Computer Architecture and Organization," 3/e, TMH, 1998.
4. Govindarajulu, "Computer Architecture and Organization" TMH.
5. Hebbar, "Computer Architecture", Macmillan,2008

Gaps in syllabus

Mapping of Course outcomes with PO to identify Curricular Gaps

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Sl.No	Module and Course	PO	CO	Comments/Gaps
1.	Embedded Electronics Computer Organization and Architecture (PC234EC)	PO1	**CO1**,CO2,CO3,CO4	**CO1 strongly maps with PO1**, as it describes and explains data representations. Arithmetic and logical operations for basic functional blocks of a basic computer. **CO3 strongly maps with PO2**, as this unit identifies the key attributes of CPU and explains various processing techniques. **CO4 strongly maps with PO3**, as it deals with communication with the external environment for its processing in the computer. In order to design complex systems, curriculum should include Verilog code representations of various digital components of a basic computer.
		PO2	CO1,CO2,**CO3**,CO4	
		PO3	CO2,CO3,**CO4**	
		PO4	CO1	
		PO5	NOT MAPPED	
		PO6	NOT MAPPED	
		PO7	NOT MAPPED	
		PO8	CO1	
		PO9	CO2	
		PO10	CO1,CO5	
		PO11	CO2,CO5	
		PO12	CO1,CO3	
		PSO1	NOT MAPPED	Since the course of Computer Organization and Architecture is related to the study of a typical controller chip and its implementation design in particular, it strongly maps to PSO2 which is in the area of embedded system design.
		PSO2	CO1,CO4	
		PSO3	NOT MAPPED	
		PSO4	NOT MAPPED	

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Course Plan/Lesson Plan

MJCET

Banjara Hills, Hyderabad – 500034

Department of ECE

LESSON PLAN

Faculty Name: Mr. J.K.Nag **Dept: ECE**

Subject Name: COMPUTER ORGANIZATION AND ARCHITECTURE ***Code: PC234EC***

Year: II ***Semester: IV***

Degree & Branch: BE (ECE A) ***Academic Year: 2020-2021***

Lecture No.	Topics to be covered	PPT/BB/ OHP/ e-material	No. of Hrs	Relevant COs

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

1.	**Data representation and Computer arithmetic:** Introduction to Computer Systems, Organization and Architecture, evolution and computer generations;	**PPT**	1	CO1
2.	Fixed point representation of numbers	**PPT**	1	CO1
3.	Digital arithmetic algorithms for Addition, Subtraction, Multiplication hardware algorithm for signed magnitude data	**PPT**	2	CO1
4.	Multiplication using Booth's algorithm and numerical example	**PPT**	1	CO1
5.	Division using restoring and non-restoring algorithms	**PPT**	1	CO1
6.	Floating point representation with IEEE standards and its arithmetic operations	**PPT**	3	CO1
7.	**Basic Computer Organization and Design:** Instruction codes, Stored program organization	**PPT**	1	CO2
8.	Computer registers and common bus system	**PPT**	1	CO2
9.	Computer instructions, Timing and Control	**PPT**	2	CO2
10.	Instruction cycle: Fetch and Decode	**PPT**	1	CO2
11.	Register reference instructions, Memory reference instructions	**PPT**	2	CO2
12.	Input, output and interrupt: configuration, instructions, program interrupt, interrupt cycle	**PPT**	2	CO2
13.	Hardwired Control organization, Microprogrammed Control organization, address sequencing	**PPT**	2	CO2

22/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

14.	Micro-instruction format and micro program sequencer	**PPT**	2	CO2
15.	**Central Processing Unit:** General register organization, Control Word	**PPT**	1	CO3
16.	Stack Organization: Register Stack, Memory Stack	**PPT**	2	CO3
17.	Reverse Polish Notation, Evaluation of Arithmetic Expressions	**PPT**	1	CO3
18.	Instruction formats, Single Accumulator, General Register and Stack organized CPU organizations	**PPT**	1	CO3
19.	Addressing modes: Types, Numerical Example	**PPT**	1	CO3
20.	Data Transfer and manipulation, Program control.	**PPT**	2	CO3
21.	CISC and RISC: features and comparison	**PPT**	1	CO3
22.	Pipeline and Vector processing, parallel processing, pipelining, instruction pipeline	**PPT**	2	CO3
23.	Basics of vector processing and Array processors	**PPT**	1	CO3
24.	**Input-output organization:** I/O interface, I/O Bus and interface modules, I/O versus Memory Bus: Isolated Vs. Memory-Mapped I/O	**PPT**	2	CO4
25.	Example of I/O Interface	**PPT**	1	CO4
26.	Asynchronous data transfer: Strobe control, Handshaking Timing Diagrams	**PPT**	1	CO4
27.	Asynchronous serial transfer, Asynchronous Communication Interface	**PPT**	2	CO4
28.	Modes of transfer: Programmed I/O, interrupt-initiated I/O	**PPT**	1	CO4

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

29.	Priority interrupt, Daisy Chaining Priority	PPT	1	CO4
30.	Parallel Priority interrupt, Priority Encoder, Priority interrupt Hardware, Interrupt Cycle	PPT	2	CO4
31.	Direct Memory Access, DMA controller and DMA Transfer in a computer system	PPT	1	CO4
32.	Input output processor (IOP), CPU-IOP Communication, I/O channel	PPT	1	CO4
33.	**Memory Organization:** Memory hierarchy, Primary memory,(RAM & ROM Chips) Memory Address Map, Memory connections to CPU	PPT	1	CO5
34.	Auxiliary Memory, Associative Memory, Hardware organization	PPT	1	CO5
35.	Cache memory, locality of reference property, Types of Mapping procedures	PPT	1	CO5
36.	Cache memory: mapping functions Direct Mapping, Associative Mapping – Numerical example	PPT	2	CO5

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

37.	Set-Associative Mapping – Numerical example Writing into cache, cache Initialization	**BB**	1	CO5
38.	Virtual memory: Address Space and Memory Space, Address mapping using pages	**BB**	2	CO5
39	Memory Management Hardware: Basic components, Segmented-Page Mapping	**BB**	1	CO5
40.	Solving Numerical Problems related to V Unit	**BB**	1	1

Course Coordinator **Program Coordinator** **HEAD, ECE DEPT**

22/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Lecture Notes:

Unit wise notes uploaded on Google Classroom

Unit wise Question Bank

23/07/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

COMPUTER ORGANIZATION AND ARCHITECTURE (PC234EC)

QUESTION BANK (2020-2021)

B.E. ECE IV SEM (AICTE) SECTION A

REVISED QESTION BANK

NOTE: These are sample questions only for preparation purpose.

UNIT III [Central Processing Unit]

Short Type Questions

1. What the advantages are of stack organized computer?
2. Write the need for different addressing modes.
3. Write the differences between RISC and CISC processors.
4. Determine the number of clock cycles that it takes to process 200 tasks in a six segment pipeline.
5. Mention the different types of instruction formats.
6. What is a stack and what is its role in the operation or execution of computer instructions?
7. Discuss various types of CPU organizations.
8. Write the features of a RISC processor.
9. What is meant by pipeline hazard?
10. Draw a space-time diagram for a six-segment pipeline showing the time it takes to process eight tasks.
11. What are the different addressing modes of a basic computer?
12. What is vector processing?

Essay Type Questions

1. Explain instruction formats for various types of computer organizations as single accumulator, general register and stack.
2. Explain various types of interrupts in brief.
3. What is the need for addressing mode? Explain various addressing modes of general purpose computer.
4. Explain data manipulation operations of a basic computer.
5. How is vector processing is different from Array processing. Give their application areas.

03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

6. Explain pipeline conflicts and discuss their remedies.
7. Compare CISC and RISC Architectures.
8. Classify the instructions based on the fields of a instruction format.
9. A non-pipeline takes 50 ns to process a task. The same task can be processed in a six-segment pipeline with a clock of 10 ns. Determine the speed up ratio of the pipeline for 100 tasks. What is the maximum speed up that can be achieved?
10. Draw a flow chart for a six stage CPU instruction pipeline and explain.

UNIT II [Basic Computer Organization & Design]

Short Type Questions

1. What is stored program organization?
2. Write the instruction formats for memory reference and register reference instructions of a basic computer.
3. What is the difference between a direct and an indirect address instruction? Illustrate with an example.
4. Explain the use of the following registers in basic computer:
   (i) PC (ii) DR (iii) IR
5. Compare Hardwired and Microprogrammed Control Organization.
6. What are the sub phases in an Instruction cycle of a basic computer?
7. Define the following terms: i) Micro operation ii) Micro instruction iii) Micro program.
8. Hardwired Control is faster than micro-programmed control unit. Justify this statement.
9. Draw the microinstruction format and specify each field.

Essay Type Questions

1. With a neat diagram of a common bus system, show how to execute the micro operation

   $AC \longleftarrow AC + DR$
2. Draw the block diagram of a control unit of a basic computer and explain.
3. Explain the various phases of an Instruction cycle with a flow chart.

Previous University Question papers

Code No: 2936/AICTE

FACULTY OF ENGINEERING

B.E. IV - Semester (AICTE) (ECE) (Main) Examination, December 2020

Subject: Computer Organization & Architecture

Time: 2 Hours Max. Marks: 70

Note: (Missing data if, any can be assumed suitable)

PART-A

Answer any five questions. (5 X 2= 10 Marks)

1. Define address space and memory space.
2. Draw the Basic Computer instruction formats
3. Differentiate between single precision and double precision IEEE standard floating point representations.
4. List the Memory reference instructions?
5. Write the differences between RISC and CISC processors.
6. List the applications are of stack organized computer?
7. Compare memory mapped I/O and I/O mapped I/O.
8. Mention various modes of data transfer.
9. Draw the Hardware required for Addition and Subtraction
10. Define Page in Memory management.

PART-B

Answer any four questions. (4 X 15 = 60 Marks)

11. (a) Explain Booth's algorithm with a suitable example
    (b) Explain Floating point Division.

12. (a) Explain the operation of a address sequencer in a micro programmed control unit
    (b) Explain various phases of an instruction cyclein detail.

13. Draw and explain the space time diagram for a four –segment pipeline showing the time it takes to process five tasks.

14. (a) Explain the DMA transfer with block diagram.
    (b) Explain CPU-IOP Communication.

15. Describe the different addressing mapping techniques in the cache memory?

16. (a) Explain Common Bus for Memory unit of 4096 X 16.
    (b) Draw and Explain memory table mapping with virtual Address

17. Write short notes on
    (a) Array Processors
    (b) Computer Generations

****

23/03/2020

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, HYDERABAD - 500 034. A.P

Code No: 15

FACULTY OF ENGINEERING

B.E. IV - Semester (AICTE) (ECE) (Backlog) Examination, August 2021

Subject: Computer Organization & Architecture

Time : 2 Hours Max. M

Missing data, if any, may be suitably assumed.

PART – A

Answer any Five Questions (5x2= 10

1. Differentiate between restoring and non-restoring algorithm
2. Compare Hardwired and Micro programmed control unit
3. What are the main four components of any general purpose computer?
4. What is stored program organization?
5. Write the features of a RISC processor
6. Brief about an instruction pipeline
7. Explain Interrupt driven I/O.
8. Compare between synchronous and asynchronous communication.
9. Draw the block diagram of Associative Memory.
10. Write about Memory hierarchy in a computer system

PART – B

Note: Answer any Four Questions (4x15= 60M

11. Write and explain Restoring division algorithm using a suitable example

12. (a) Explain the operation of a micro programmed control unit with block diagram
    (b) Discuss about Common bus system with a example

13. (a) Draw the space time diagram for a four – segment pipeline showing the time it takes to process five tasks.
    (b) Explain various addressing modes in the general purpose computer

14. What is Asynchronous data transfer and Explain block diagram of Asynchronous Communication interface

15. Explain Direct and Associative address mapping techniques in Cache memory with One address mapping example.

16. (a) Draw and Explain Daisy chain
    (b) Explain Vector & Array Processing

17. Write short notes on:
    (a) Array Processors
    (b) Hand Shaking

*****

Code No: 2151/O/BL

FACULTY OF ENGINEERING

B.E. 3/4 (ECE) II-Semester, (Old) Examination, November 2020

Subject: Computer Organization & Architecture

Time: 2 hours **Max. Marks: 75**

Note: Answer any seven questions from Part-A and any three questions from Part-B

PART – A (7x3=21 Marks)

1. Distinguish between micro operation and micro instruction.
2. Explain the use of following registers
   i) PC ii) DR iii) IR
3. Differentiate between single precision and double precision IEEE standard floating point format representations.
4. Show the hardware implementation for integer binary division operation.
5. What are the basic differences between a branch instruction, a call subroutine instruction and program interrupt?
6. Compare and contrast Hardwired vs. Micro programmed control unit.
7. Why handshake is required between CPU and I/O devices?
8. What is the difference between isolated I/O and memory mapped I/O.
9. What do you mean by a page fault? Which hardware is responsible for detecting the page fault?
10. Draw the space-time diagram for a six-segment pipeline showing the time it takes to process eight tasks.

PART-B (3x18=54 Marks)

11. a) Using booth's multiplication algorithm, multiply (-15) × (-13) showing all the steps. Assume 5-bit registers that hold signed numbers.
    b) What is dividend alignment and why do we use it while performing division operation of binary numbers?
12. a) With neat diagram of a common bus system, show how to execute the Micro-operation AC ← AC + DR.
    b) List the control functions and micro-operations needed for the execution of the following instructions and explain.
    i) ADD ii) BSA
13. a) Write the need for addressing modes. Explain various addressing modes supported by a general purpose CPU.
    b) What is the purpose of micro program sequencer? Explain with a block diagram, how the sequencer presents address to control memory?
14. a) Draw the block diagram of an asynchronous communication interface and explain its operation.
    b) Describe in detail how data is transferred in DMA. Draw the necessary diagrams to support your explanation.

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

15. a) A non-pipeline system takes 50ns to process a task. The same task can be processed in a six-segment pipeline with a clock cycle of 10ns. Determine the speedup ratio of the pipeline for 100 tasks. What is the maximum speedup that can be achieved?

b) Explain instruction pipeline conflicts and their remedies.

16. a) Why page-table is required in a virtual memory system? Explain different ways of organizing a page table.

b) How many 128 × 8 memory chips are needed to provide a memory capacity of 4096 × 16? Give the circuit diagram of the memory using the memory chips.

17. Write short notes on any Two of the following:

a) Cache memory

b) Accumulator based CPU organization

c) Various phases of interrupt cycle in basic computer

****

Internal Question Papers with Key

INTERNAL TEST – 1 (Online mode)

PART – A (Compulsory section)

1. The product of 1101 and 1011 is
   a. 10001111
   b. 10101010
   c. 11110000
   d. 11001111

2. In IEEE 32-bit representation, the mantissa of the fraction is said to occupy __ bits.
   a. 24
   b. 23
   c. 20
   d. 16

3. The load instruction is mostly used to designate a transfer to a processor register known as
   a. Program Counter
   b. Instruction register
   c. Accumulator
   d. Address register

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

4. The number of address lines required to access memory with 16 K words will be
   a. 16
   b. 15
   c. 14
   d. 17

5. ___ Register keep track of the instructions stored in program stored in memory.
   a. AR
   b. PC
   c. IR
   d. DR

6. Individual control words of the microroutine are called as
   a. Micro task
   b. Micro operation
   c. Micro instruction
   d. Micro code

Solution: 1 - a, 2 – b, 3 – c, 4 – c, 5 – b, 6 -c

22/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

PART – B (answer any Two)

7. Draw the hardware required for implementing Booth's multiplication algorithm and show the step-by step multiplication process for the following example (+15) x (+13). Assume 5-bit registers that hold signed numbers.

8. a) Write the micro operations needed for the fetch and decode Phases of an Instruction Cycle in Basic computer.

   b) List the control functions and micro operations needed for the execution of the following memory reference instructions:

   (i) ADD (ii) LDA

9. What is the purpose of microprogram sequencer? Explain with a block diagram, how the sequencer present addresses to control memory.

INTERNAL TEST – 2 (Online mode)

PART – A (Compulsory section)

1. A program counter contains a number 835 and address part of the instruction contains the number 24. The effective address in the relative address mode, when an instruction is read from the memory is __
   a. 859
   b. 860
   c. 811
   d. 812

2. The number of clock cycles needed to process 200 tasks in a six-segment pipeline will be __
   a. 305
   b. 210
   c. 205
   d. 310

3. In the memory-mapped I/O
   a. The I/O devices has a separate address space
   b. The I/O devices and memory share the same address space
   c. A part of the memory is specifically set aside for the I/O device
   d. None of the above

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

4. In a DMA write operation, the data is transferred
   a. From I/O to memory
   b. From memory to I/O
   c. From memory to memory
   d. From I/O to I/O

5. The average time required to reach a storage location in memory and obtain its contents is called the
   a. Seek time
   b. Turnaround time
   c. Access time
   d. Transfer time

6. How many 256 × 8 RAM chips are needed to provide a memory capacity of 4096 bytes?
   a. 8
   b.18
   c. 16
   d. 32

solution: 1 – b, 2 – c, 3 – b, 4 – a, 5 – c, 6 - c

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

PART – B (answer any Two)

7. (a) What are the basic differences between a Branch instruction, a Call subroutine instruction and Program interrupt?

   (b) Draw the space time diagram for a four-segment pipeline showing the time it takes to process five tasks.

8. Draw the block diagram of an Asynchronous Communication Interface and explain its operation.

9. (a) What is the disadvantage of transferring data through Strobe control method? How Handshaking overcomes this disadvantage? Explain.

   (b) Explain the need of Memory Hierarchy in a computer system with the help of a block diagram.

Unit wise Assignment Questions

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 1: Submission Date: 03-05-2021

1. Perform the arithmetic operations (+46) + (-15) and (-46) – (-15) in binary using signed-2's complement representation for negative numbers. Use eight bits to accommodate each number together with its sign.
2. Perform the arithmetic operations (+60) + (+70) and (-60) + (-70) with binary numbers in signed-2 complement representation. Use eight bits to accommodate each number together with its sign. Show that overflow occurs in both cases.
3. Show the hardware for signed-2's complement addition and subtraction.
4. Show the hardware for implementing Booth's algorithm. Show the step-by-step multiplication process using Booth algorithm for the following example (+15) × (-13). Assume 5-bit registers that hold signed numbers.
5. Differentiate between restoring and non-restoring algorithm. What is divide overflow condition?
6. Represent the number (+37.5) as a floating-point binary number with 24 bits. The normalized fraction mantissa has 16 bits and the exponent has 8 bit bits.

PRINCIPAL 03/2022
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

7. Differentiate between single precision and double precision IEEE standard floating point representations with their formats.

8. Explain the process of floating point number multiplication with a neat flow chart.

9. What is normalization and alignment in floating point arithmetic?

10. Explain the floating point division operation using a neat flow chart. What is meant by dividend alignment?

J.K.NAG
Assoc.Prof. (ECED)

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 2: Submission Date: 28-06-2021

1. a) What do you understand by stored program concept? Explain.

b) With a neat sketch of a common bus system, show how to execute the micro operation AC $\longleftarrow$ AC + DR.

2. a) List the control functions and micro operations needed for the execution of the following instructions and explain.
i) ADD ii) BSA iii) ISZ

b) Explain various phases of an instruction cycle in detail with flow chart.

3. Explain input-output configuration of a computer and list any three I/O instructions with their control functions and micro operations.

4. Describe the various phases of an interrupt cycle and its micro operations.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

5. a) Differentiate between Hardwired and Micro programmed control. Mention their merits and demerits.

   b) Explain the operation of a address sequencer in a micro programmed control unit.

6. What is the purpose of micro program sequencer? Explain with Block diagram, how the sequencer present address to control memory

J.K.NAG
Assoc.Prof. (ECED)

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 3: Submission Date: 23-08-2021

1. Explain instruction formats for various types of computer organizations as single accumulator, general register and stack.

3. What is the need for addressing modes? Explain about the various addressing modes in a general purpose computer.

4. An instruction is stored at location 300 with its address field at location 301. The address field has the value 400. A processor register R1 contains the number 200. Evaluate the effective address if the corresponding mode of the instruction is (a) direct; (b) immediate; (c) relative; (d) register indirect; (e) index with R1 as the index register.

5. Convert the following arithmetic expressions from infix to reverse polish notation.

23/03/2022
PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

a. A * B + C * D + E * F b. A + B *[C * D + E *(F +G)]

6. Convert the following numerical arithmetic expression into reverse polish notation and show the stack operations for evaluating the numerical result.

$$(3 + 4)[10(2 + 6) + 8]$$

7. Write the features of RISC Processor. Write the differences between RISC and CISC Processors.

8. Explain any four data manipulation instructions of a general purpose computer.

9. What are the basic differences between a branch instruction, a call subroutine instruction and program interrupt?

10. Draw the space time diagram for a four segment pipeline showing the time it takes to process five tasks.

11. Draw the flow chart for a six-stage CPU instruction pipeline and Explain.

12. A non-pipeline system takes 50 ns to process a task. The same task can be processed in a six-segment pipeline with a clock cycle of 10 ns. Determine the speed up ratio of the pipeline for 100 tasks. What is the maximum speed up that can be achieved?

13. Explain pipeline conflicts and briefly discuss the remedies for those conflicts.

14. How vector processing is different from array processing and give their application areas?

J.K.NAG
Assoc.Prof. (ECED)

23/03/2022

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 4: Submission Date: 30-08-2021

1. Explain the need for an I/O interface. Show an example of a typical I/O Interface unit.
2. Compare memory mapped I/O and I/O mapped I/O with merits and demerits.
3. What is the disadvantage of transferring data through strobe control method? How handshaking overcomes this disadvantage? Explain.
4. Design parallel priority interrupt hardware for a system with eight interrupt sources.
5. Draw the block diagram of an asynchronous communication interface and explain its operation.
6. Explain the DMA transfer with block diagram.
7. What is an Input-output processor (IOP)? Explain how CPU and IOP communicate with each other through a flow chart.

J.K.NAG
Assoc.Prof. (ECED)

22/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

M J C E T
IV-SEMESTER ECE A (2020-2021)
COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 5: Submission Date: 14-09-2021

1. How many 128 × 8 RAM chips are needed to provide a memory capacity of 4096 bytes? How many lines of the address bus be used to access 4096 bytes? How many lines must be decoded for chip select? Specify the size of the decoders.

2. Derive the Match logic for one word of associative memory and explain how the read operation is performed.

3. Explain the need of memory hierarchy with the help of a Block diagram.

4. Explain Direct and Associative address mapping techniques in Cache memory with an example.

5. Consider a set-associative cache that consists of 256 blocks divided into 4-block sets and a byte addressable main memory of size 512K bytes, with block size of 32 bytes each. How many bits will be there in the TAG, SET and WORD fields respectively?

6. Why page-table is required in a virtual memory system? Draw and explain the organization of typical memory mapping table in a paged system.

7. Using block diagram explain the mapping of logical address into physical address in a segmented-page memory management unit. What is the importance of a TLB in the memory management unit?

J.K.NAG

Assoc.Prof. (ECED)

23/07/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

Content beyond Syllabus:

N/A

Methodology used to identify weak and bright students:

Support extended to weak students –

Assessment thro Tutorial questions unit wise

Solving the question bank unit wise

Faculty instructor: J.K.NAG

Designation: Associate Professor

Dept: ECE

Class: ECE A Section

Semester: IV

Scheme: AICTE

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

CoA internal Test - 2

PART - B

Soln Key

7a) Branch Instruction:

-> when a branch instruction is encountered the content of pc is altered and pc is changed to address specified by the address in the instruction and the return address of the next instruction is stored is stack.

-> It may or may not use status bits while branching.

-> It is included by the programmer in the program.

eg: BUN, BZ, BNZ etc.

Call subroutine Instruction

-> when call subroutine instruction is encountered a sequence of instructions are executed and will return to the next instruction where the branch instruction was encountered.

-> It won't use status bits to call subroutines

-> It is included the programmer

eg:- CALL:

Program Interrupt:

→ It refers to transfer a program control from a currently running program to another service program. As a result of an external or internal generated request.

→ It uses status bits.

→ It is not included by the programmer in the program.

7b) The behaviour of a pipeline can be illustrated with a space-time diagram.

→ This is a diagram that shows the segment utilization as a function of time. The space time diagram for a four segment pipeline is demonstrated in below fig.

segments ↓ / clock cycles →	1	2	3	4	5	6	7	8
1	T_1	T_2	T_3	T_4	T_5			
2		T_1	T_2	T_3	T_4	T_5		
3			T_1	T_2	T_3	T_4	T_5	
4				T_1	T_2	T_3	T_4	T_5

Time taken $= (k+n-1)\,t_p$

where $k \rightarrow$ no. of segments

$n \rightarrow$ No. of tasks

$t_p \rightarrow$ propogation delay.

$k = 4$

$n = 5$

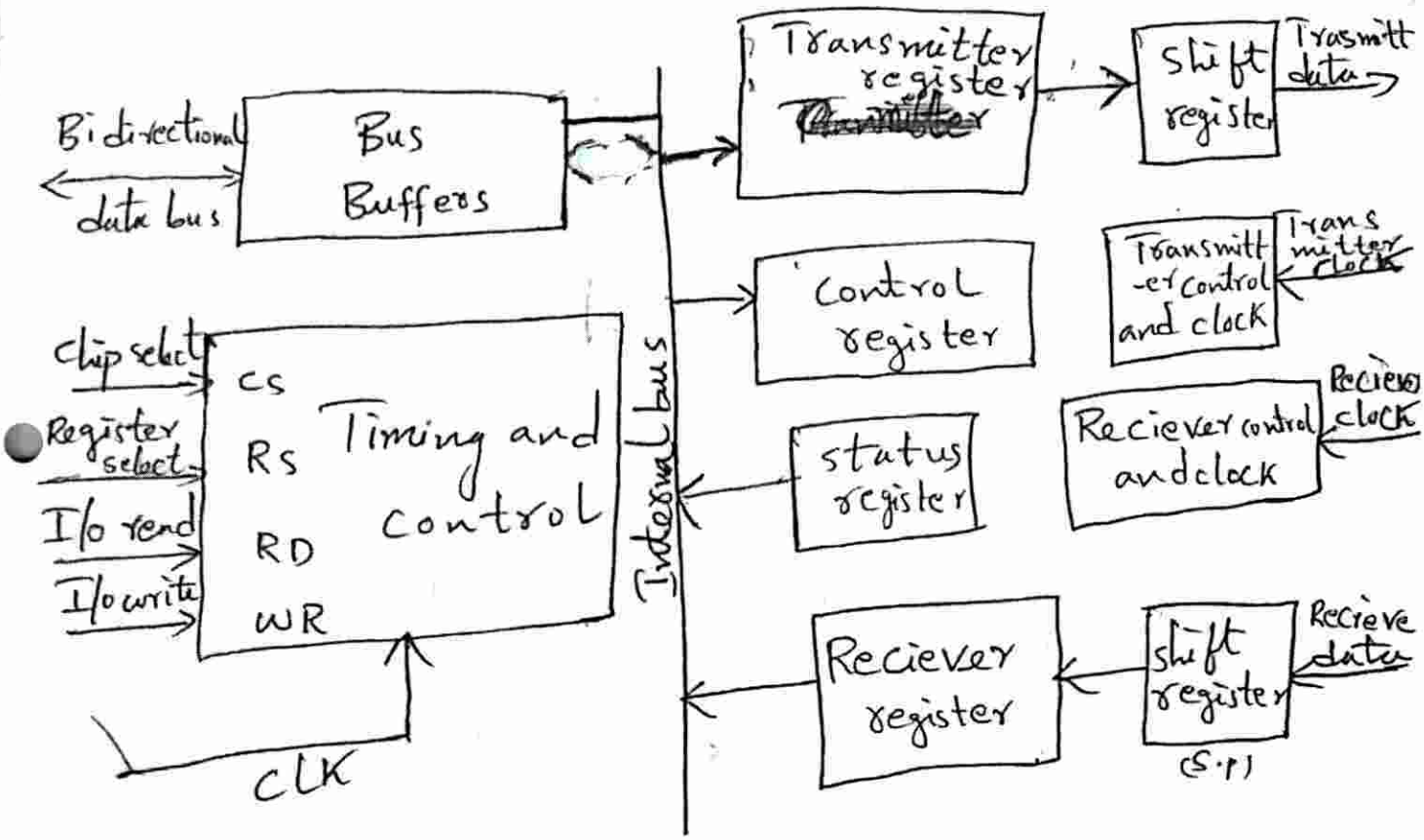
$t_n = (4+5-1)\,t_p$

$= \underline{8\,t_p}$

22/07/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

8 Ans] Block diagram of typical asynchronous communication



CS	RS	Operation	Register selected
0	X	X	None: data bus in high-impedence
1	0	WR	Transmitter register
1	1	WR	Control register
1	0	RD	Receiver register
1	1	RD	Status register

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, HYDERABAD - 500 034. A.P

→ A synchronous communication Interface functions as both transmitter and a register. The interface is initialized for a particular mode of transfer by means of a control byte that is loaded into its control register.

→ The transmitter accepts data type through data bus, it is transfered to a Shift register for serial transmission.

→ The receiver portion receives serial information into another shift register, and when a complete data byte is accumulated it is transferred to receiver register.

→ The ~~cpu~~ cpu can read the status register to check the status of the flag bits and to determine if any errors have occured.

→ The clip select (c.s) i/p is used to select the interface through the address bus. The register select (RS) is associated with the read (RD) and write (WR). Controls two registers are write only and two are read only.

Transmitter register :-

(i) Accepts a data byte (from cpu) through the data.

(ii) Transferred to a shift register for serial transmission

Reciever:

(i) Reciever serial information in to another shift register

(ii) complete data types is sent to the receiver register

status Register bits:-

used to I/O flags and for recording errors.

Control Register bits:-

Define baud rate, number of bits in each character, whether to generate and check parity, and number of stop bits.

23/03/2022

(9A)

(a). Strobe methods:

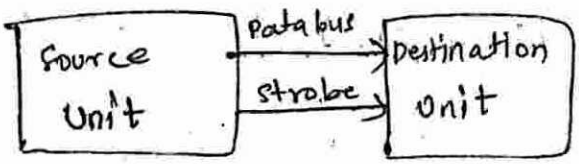
1) Employs a single control line time each transfer.

2) The strobe may be activated by either the source or the destination unit

Strobe methods:

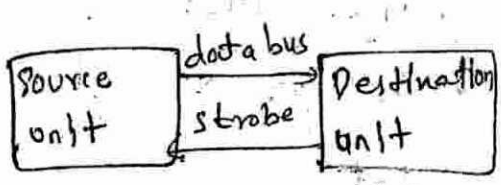
Source-initiated

Block diagram

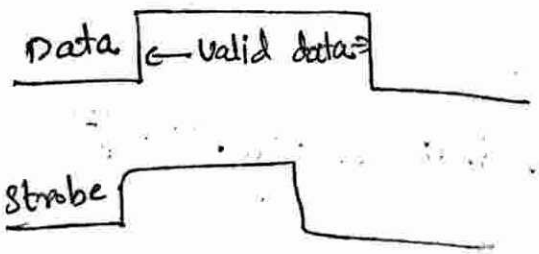


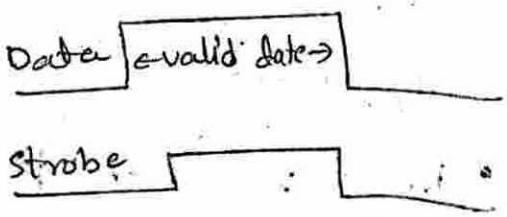
Destination-initiated

Block diagram



Timing diagram.





Source-initiated Methods:

1). The source unit that initiates the transfer has no way at knowing whether the destination unit has actully received data.

Destination-Initiated Method:

The destination unit that initiates the transfer

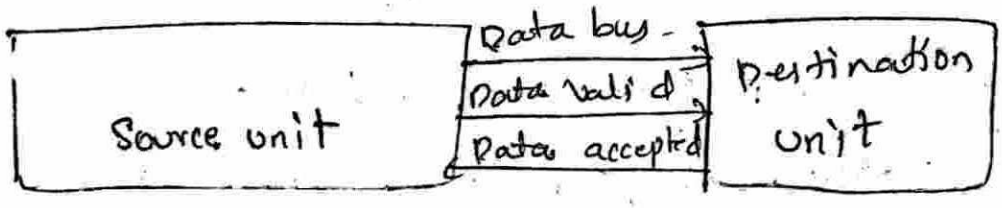
no way of knowing whether the source has actually placed the data on the bus

actually placed the data on

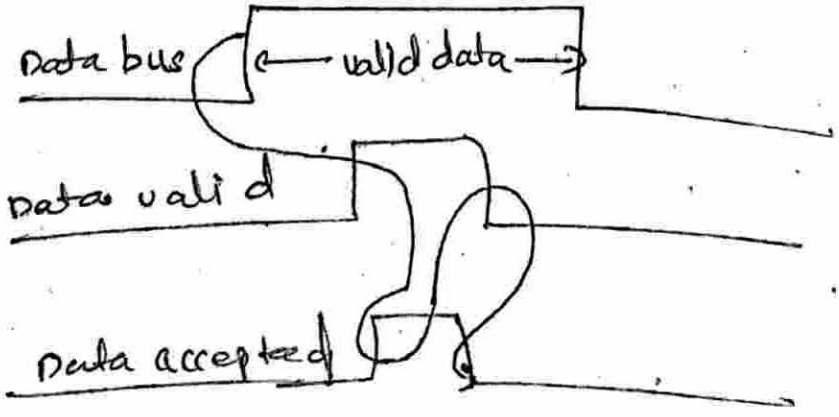
=) To solve this problem the HANDSHAKE method introduces a second control signal to provide a reply to the unit that initiates the transfer.

Source - initiated Transfer using Handshake

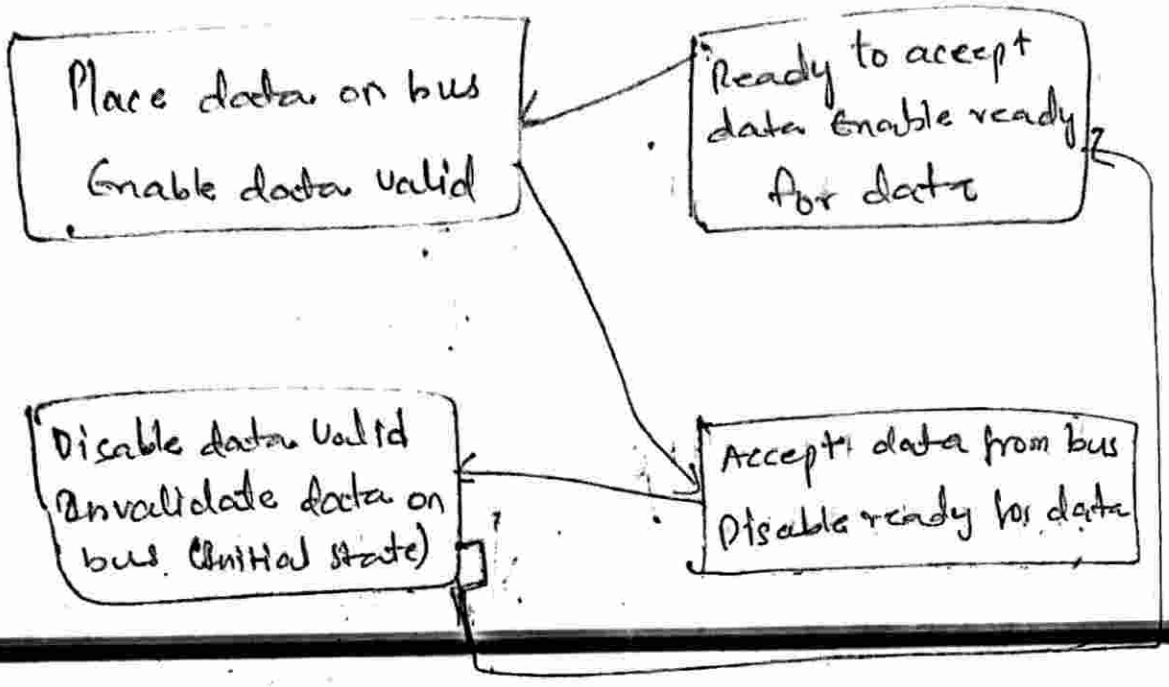
Block diagram:



Timing diagram:



Sequence of Events:

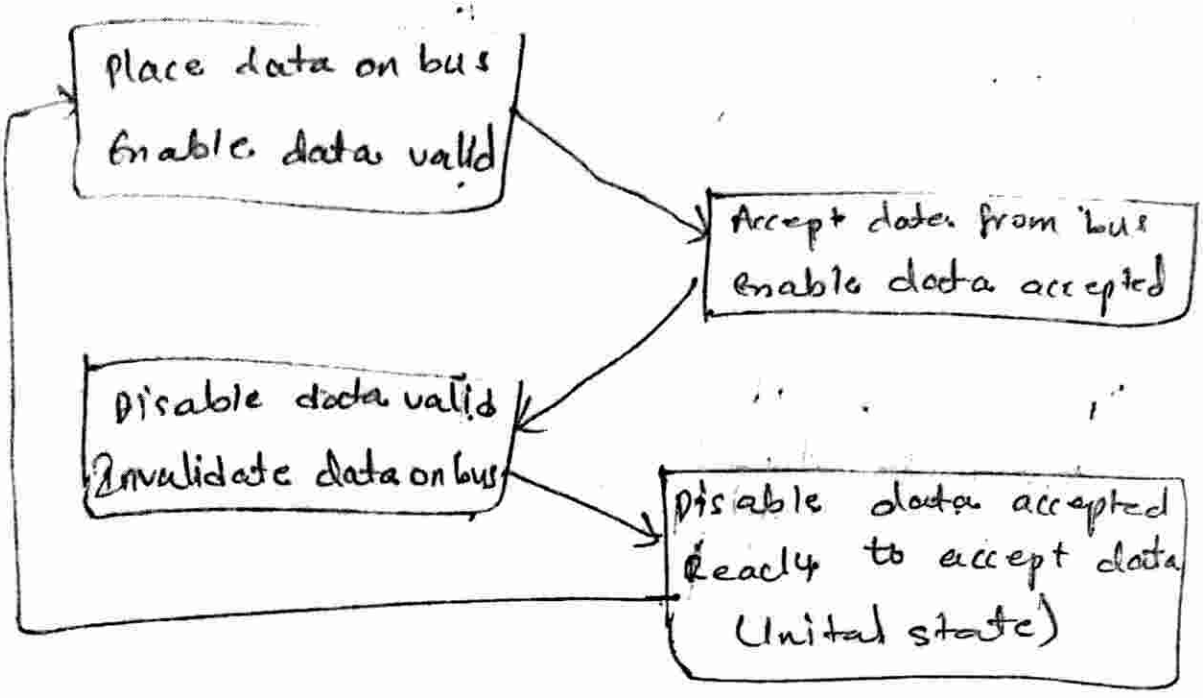


⇒ Handshaking provides a high degree of flexibility and reliability because the successful completion data transfer relies on active participation by both units.

→ If one unit is faulty data transfer will not be completed

→ can be detected by means of timeout mechanism.

Sequence of events!

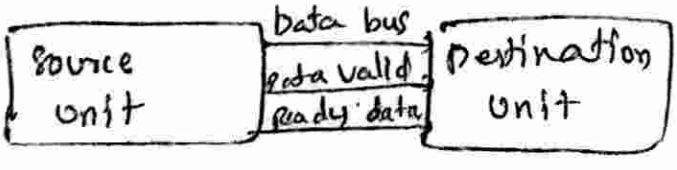


=) Allows arbitary delays from one state to the rent.

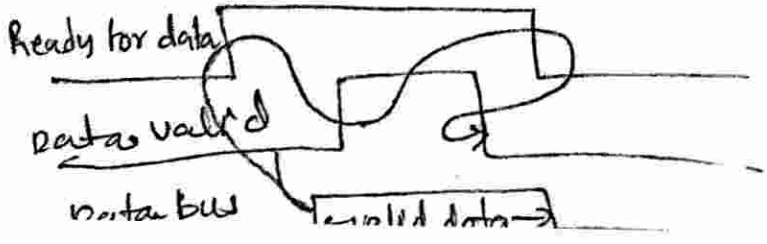
=) Deomits each unit to respond at its own data transfer date.

=) Destination - initiated Transfer using Handshake.

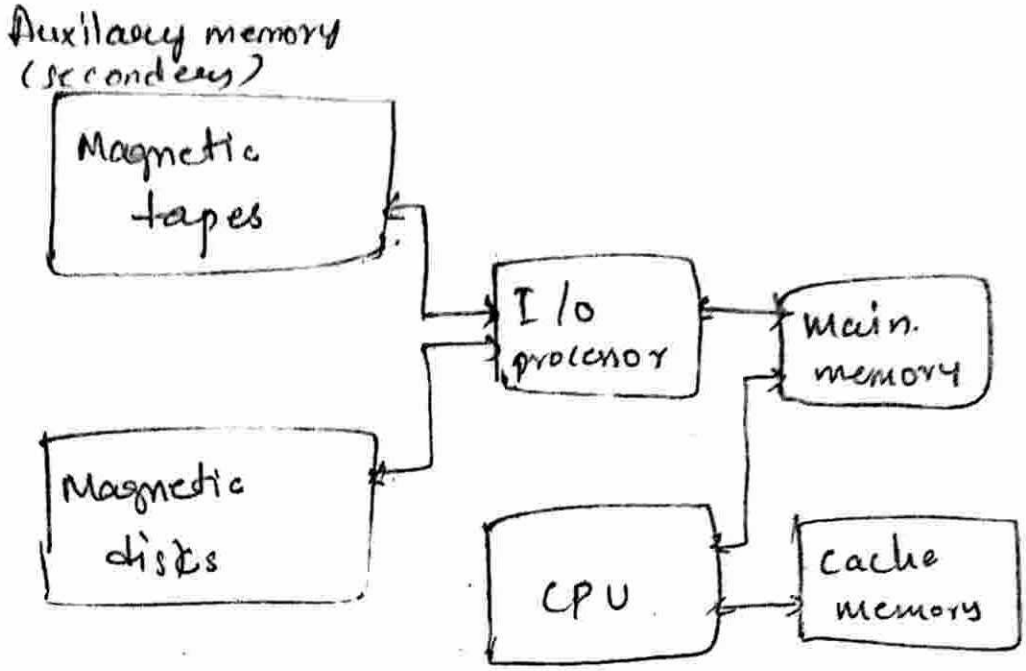
-) Block diagram.



Timing diagram



9) (b) Memory Hierarchy in a computer system.



- Magnetic tapes used to store removable files
- Next magnetic disk are used for backup storage.
- Main memory occupies a central position. and communicate with CPU & Auxilary memory devices.

CoA Internal Test – 1

PART – B

Soln Key

7A) Given $(+15) \times (+13) = +195$

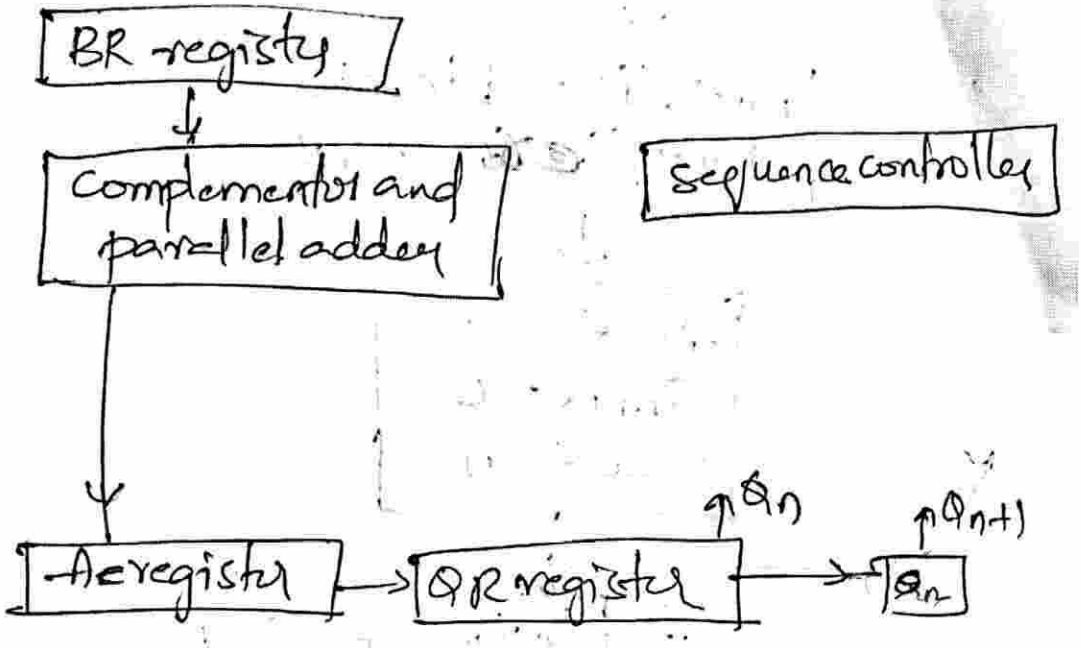
$= (0011000011)_2$

$BR = 01111\ (+15)$, $\overline{BR}+1 = 10001\ (-15)$;

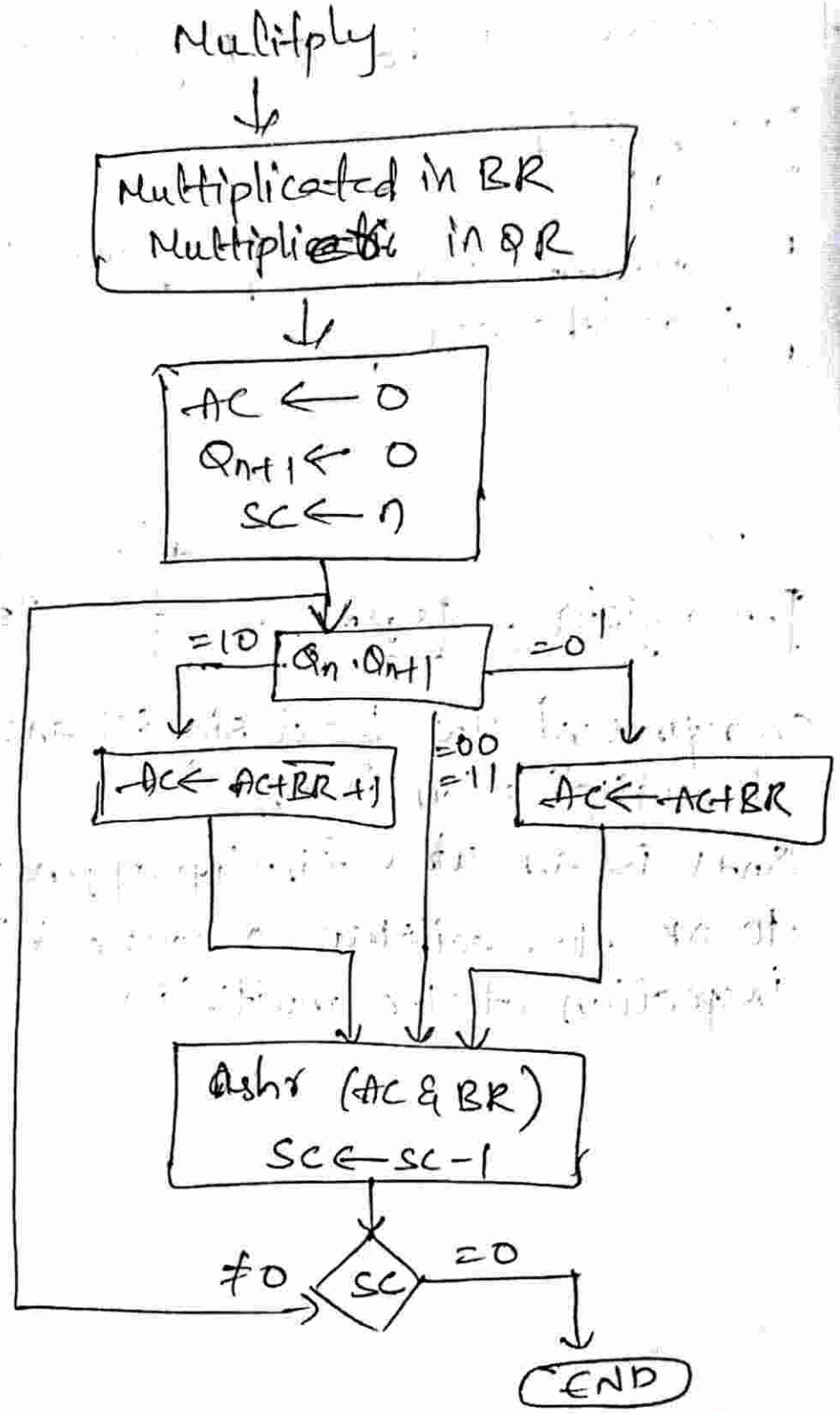
$QR = 01101\ (+13)$

Q_n	Q_{n+1}		AC	QR	Q_{n+1}	SC
1	0	Initial	00000	01101	0	101
		subtract BR	10001			
			10001			
0	1	ashr —	11000	10110	1	100
		Add BR	01111			
			00111			
1	0	ashr —	00011	11011	0	011
		sub BR	10001			
			10100			
1	1	ashr —	11010	01101	1	010
		ashr —	11101	00110	1	001
0	1	Add BR	01111			
			01100			
		ashr —	00110	00011	0	000
			+195			

Hardware for booth Algorithm



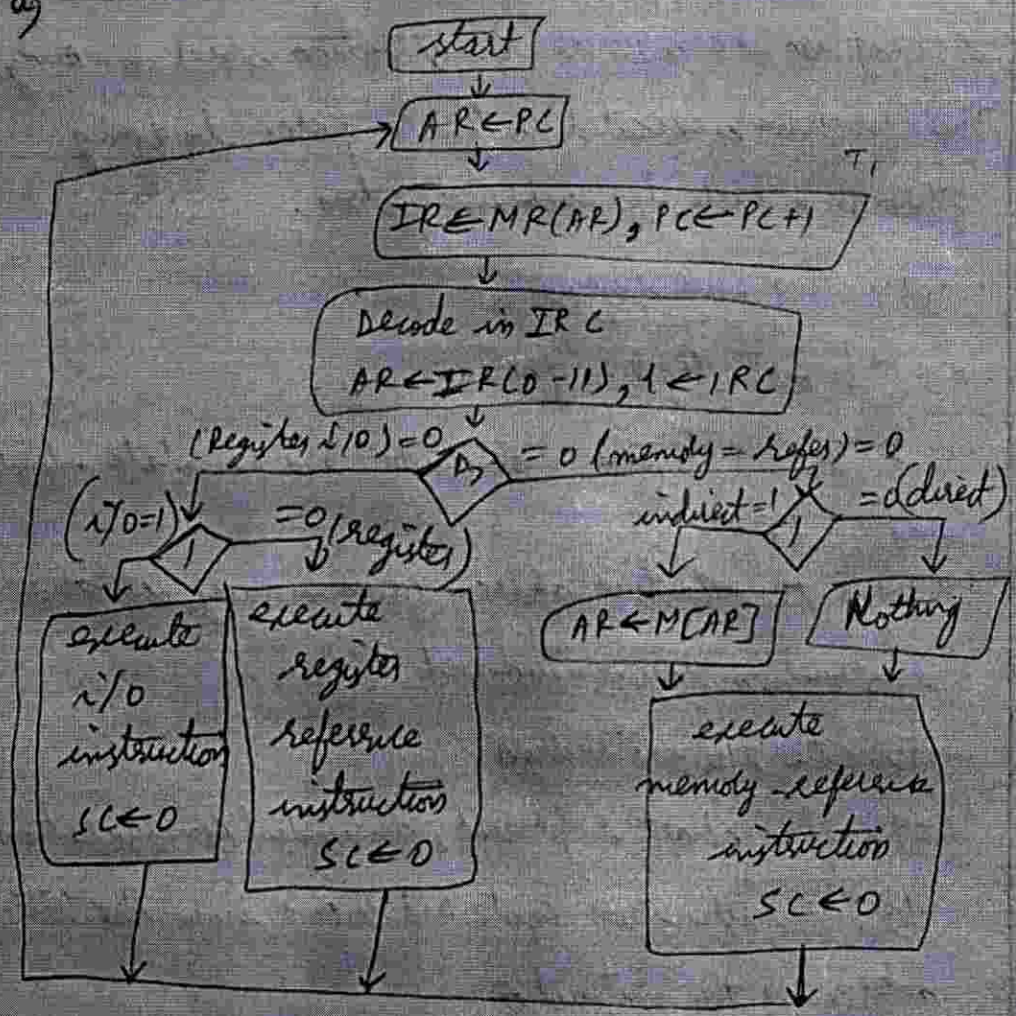
Qn represent the least significant bit of multiplication in Qn
Qn+1 is an extra flipflop appendend to QR to facilitat a double bit inspection of the multiplier



Branch: ECE-A

Subject: COA

(8) a)



D_7/T_3 : AR ← M[AR]

D_7/T_3 : Nothing

D_7/T_3 : execute a register reference instruction

D_7/T_3 : Execute an i/p - o/p instruction

PRINCIPAL

Muffakham Jah College of Engineering & Technology

HYDERABAD - A.P.

[illegible]

[illegible] Branch: ECE-A

[illegible] Subject: COA

8) a) A program is a sequence of instructions stored in memory. The program is executed in the computer by going through a cycle for each instruction.

Instruction cycle phase:

→ Fetch an instruction from memory.

→ Decode the instruction and calculate effective address (EA).

→ Read the effective address from memory if the instruction has an indirect address.

→ Execute the instruction.

The flow chart represent, an initial configuration for the instruction cycle and shows how the control determines the instruction.

(8) b) i) ADD:

D_1T_0 : $AR \leftarrow PC$

The address in PC is moved to AR.

D_1T_1 : $IR \leftarrow M[AR]$, $PC \leftarrow PC+1$

With the address in AR, from the memory the instruction is fetched into IR. PC is incremented to next instruction address is by 1.

D_1T_5: $AC \leftarrow AC + DR$, $SC \leftarrow 0$

The operand in DR is added with content of AC & the sum is stored in AC. The sequence counter is cleared.

(ii) LDA: load to AC

This instruction transfers the memory word specified by the effective address to AC.

The microoperations needed to execute this instruction are

D_2T_4 : $DR \leftarrow M[AR]$

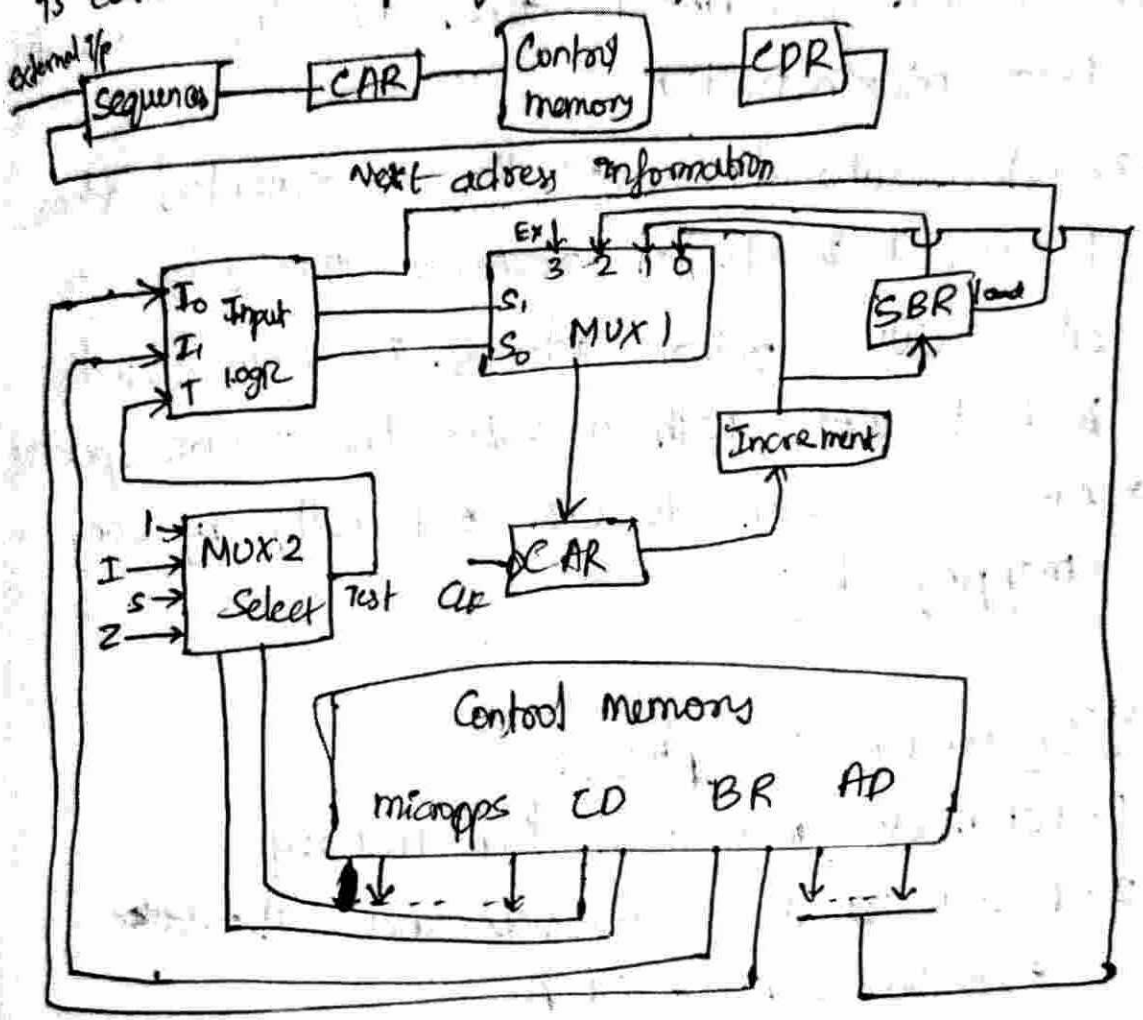
D_2T_5 : $AC \leftarrow DR$, $SC \leftarrow 0$

23/02/2021

PRINCIPAL
Muffakham Jah College Of Engineering & Technology
Banjara Hills
HYDERABAD

Internal test - 1

(5) 9) The basic components of micro program control unit are the control memory and the circuit that select the next address to the control memory is called microprogram sequence.



→ The sequencer determines the specific adress source to be loaded into the control adress register (CAR)

→ Here MUX 1 select an address from 4 sources and route it into a CAR

→ The o/p from CAR provides the adress for the control memory.

→ The content of CAR are incremented and applied to the multiplexed and subroutine

→ Branching is achieved by specifying the branch adress in one of the fields of the microoperation

→ Conditional branching is done by using status bits from microinstruction.

→ During subroutine call the incremented adress is stored in external register. this adress is also called return adress. is transferred back to CAR with a subroutine return operation.

→ Adress is given to CAR externally by using mapping logic

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 1: Submission Date: 03-05-2021

1. Perform the arithmetic operations (+46) + (-15) and (-46) – (-15) in binary using signed-2's complement representation for negative numbers. Use eight bits to accommodate each number together with its sign.
2. Perform the arithmetic operations (+60) + (+70) and (-60) + (-70) with binary numbers in signed-2 complement representation. Use eight bits to accommodate each number together with its sign. Show that overflow occurs in both cases.
3. Show the hardware for signed-2's complement addition and subtraction.
4. Show the hardware for implementing Booth's algorithm. Show the step-by-step multiplication process using Booth algorithm for the following example (+15) × (-13). Assume 5-bit registers that hold signed numbers.
5. Differentiate between restoring and non-restoring algorithm. What is divide overflow condition?
6. Represent the number (+37.5) as a floating-point binary number with 24 bits. The normalized fraction mantissa has 16 bits and the exponent has 8 bit bits.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

7. Differentiate between single precision and double precision IEEE standard floating point representations with their formats.
8. Explain the process of floating point number multiplication with a neat flow chart.
9. What is normalization and alignment in floating point arithmetic?
10. Explain the floating point division operation using a neat flow chart. What is meant by dividend alignment?

J.K.NAG
Assoc.Prof. (ECED)

COA ASSIGNMENT I Soln Key

1) $(46)_{10} = (101110)_2$

$(15)_{10} = (1111)_2$

$+46 = 00101110$

$+15 = 00001111$

$-15 = 11110001$

$(+46) + (-15) = +(46-15)$

$$\begin{array}{r} 00101110 \\ +\ 11110001 \\ \hline 00011111 \end{array}$$

$\therefore$ Neglect overflow carry.

$(00011111)_2 = (31)_{10}$

2) $(+60)_{10} = (00111100)_2$

$(+70)_{10} = (01000110)$

2	60
2	30 - 0
2	15 - 0
2	7 - 1
2	3 - 1
	1

$+60 = 00111100$

$+(+70) = 01000110$

$$\begin{array}{r} 00111100 \\ +\ 01000110 \\ \hline 10000010 \end{array}$$

overflow.

23/03/2022

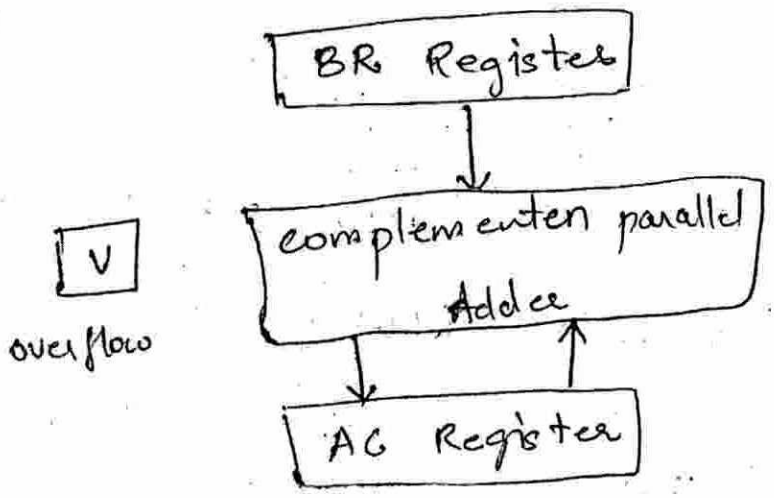
PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

Overflow is detected. MSB is treated as overflow because an overflow occured in the sign bit.

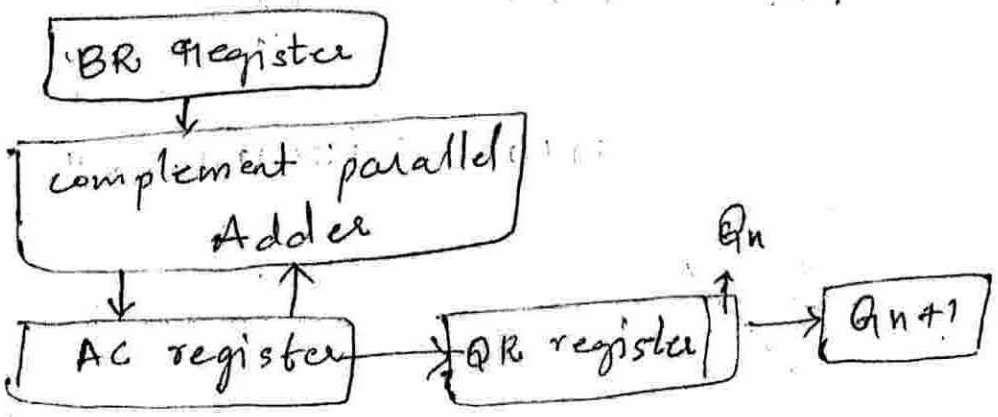
$$
\begin{array}{rcl}
(-60) & = & 11000100 \\
+(-70) & = & 01000110 \\
 & (1) & \overline{01111110}
\end{array}
$$

Overflow occured out of the sign bit so, overflow bit is activated.

(3)



(4) Booth's Algorithm Hardware.



$BR - 01111$ (+15) ; $QR = 10011$ (−13) ; $\overline{QR}+1 = 01101$

$\overline{BR}+1 = 10001$ $\qquad (+15)\times(-13) = -195$

Q_n	Q_{n+1}		AC	QR	Q_{n+1}	Sc
		Initial –	00000	10011	0	101
1	0	SUB –	10001			
		BR	10001			
		Ashr –	11000	11001	1	100
1	1	Ashr –	11100	01100	1	011
0	1	ADD –	01111			
		BR	01011			
		Ashr	00101	10110	0	010
0	0	Ashr	00010	11011	0	001
1	0	SUB	10001			
		BR	10011			
		Ashr	11001	11101	1	000

AC QR – $\underbrace{1100111101}_{-195} \rightarrow (0011000011)_2 = (195)_{10}$ verify

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

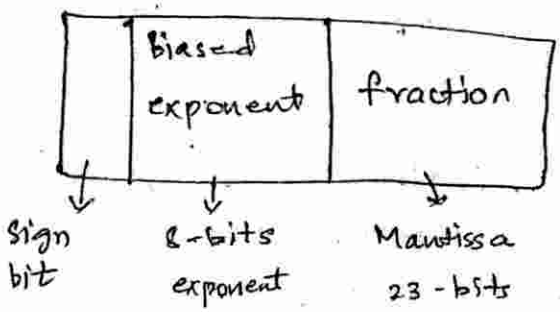
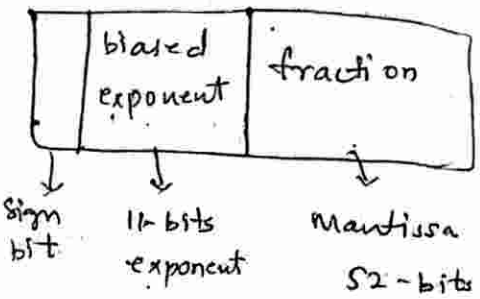
5)

Restoring	Non - Restoring.
• In restoring division algorithm partial remainder is restored by adding the divisor to negative divisor.	• In non-restoring division algorithm, the negative divisor is shifted left and the added with divisor.
• when $A < B$ $(A-B)+B$	• When $A < B$
• $(2(A-B)+B)-B = 2A-B$	• $2(A-B)+B = 2A+B$.
• Partial remainder restoring is required.	• Partial remainder restoring is not required.

(6) (+ 37.5)

$$\underbrace{0}_{\text{sign bit}} \quad \underbrace{10010100000000}_{\text{Mantise.}} \quad \underbrace{00000110}_{\text{Exponent}}$$

(7)

Single Precision	Double Precision
• It defines a 32-bit single format with 8-bit exponent and remaining bits as mantissa except sign bit.	• It defines a 64-bit Double format with 11-bit exponent and remaining bits as mantissa except sign bit.
• 	• 
• 1-bit sign +8-bit exponent +23-bit fraction	• 1-bit sign +11-bit exponent +52-bit fraction

(9) Normilzation:

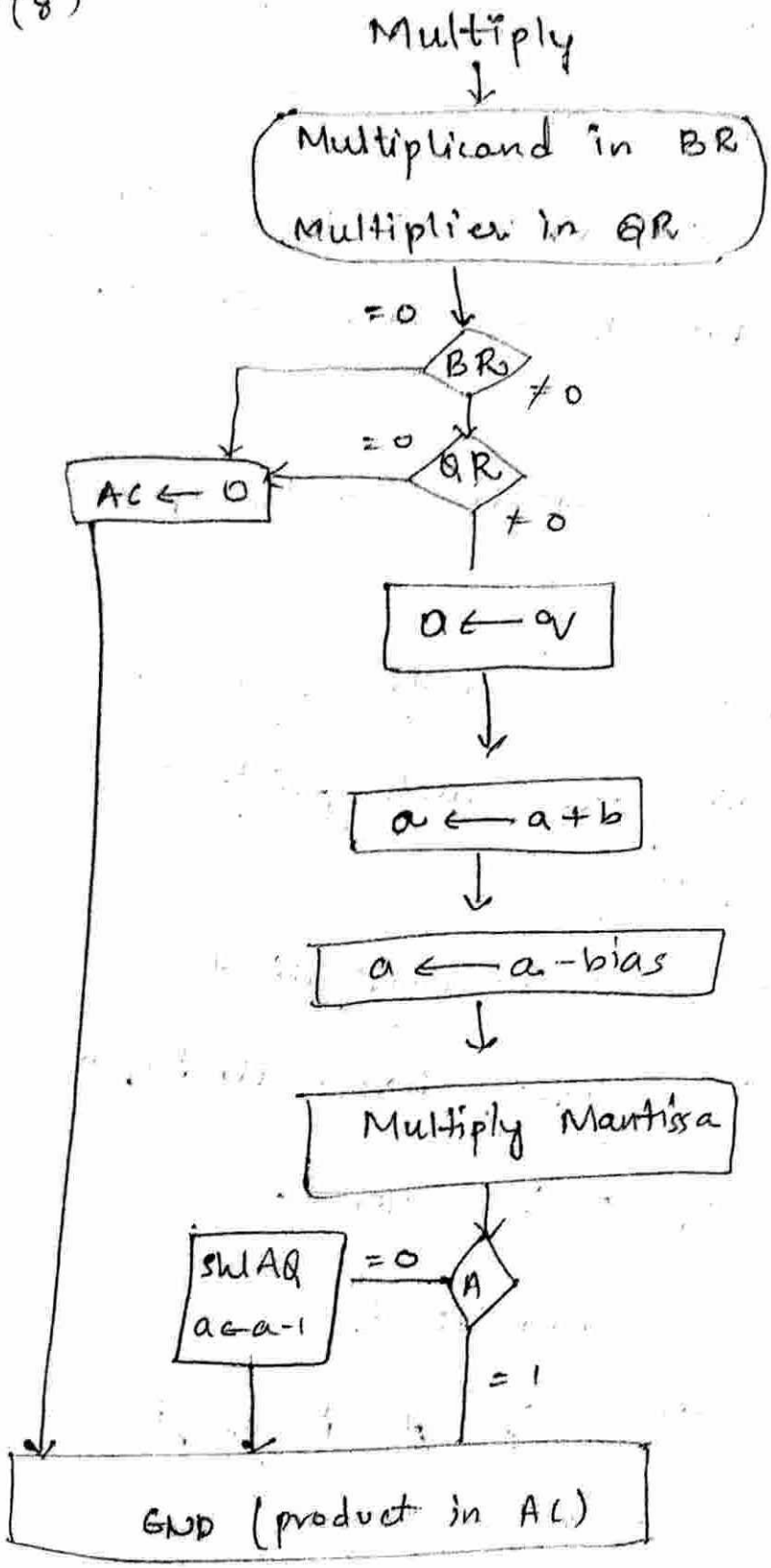
- A floating point number is said to be normalized if the MSB of mantissa is non-zero

  Ex : 11010000

- Alignment : It is the procedure of making exponent parts equal by shifting the mantissa.

b) Smaller exponent to the right and increment the exponent by 1. Repeat it untile exponents become equal.

(8)



1) Multiplicant and Multiplier are checked to determine for a zero.

2) If either operand is equal to zero the product in AC is set zero and the operation is terminated.

3) If neither of the operands is equal to zero the process continues with exponent addition.

4) The exponent of multiplier is tranferred to a.

5) The multiplication of mantissa is done as fixed point case with product term in A & Q.

6) Overflow cannot occur in multiplication. The product term may have underflow so the MSB of A is checked.

7) If A=1, It is normalized

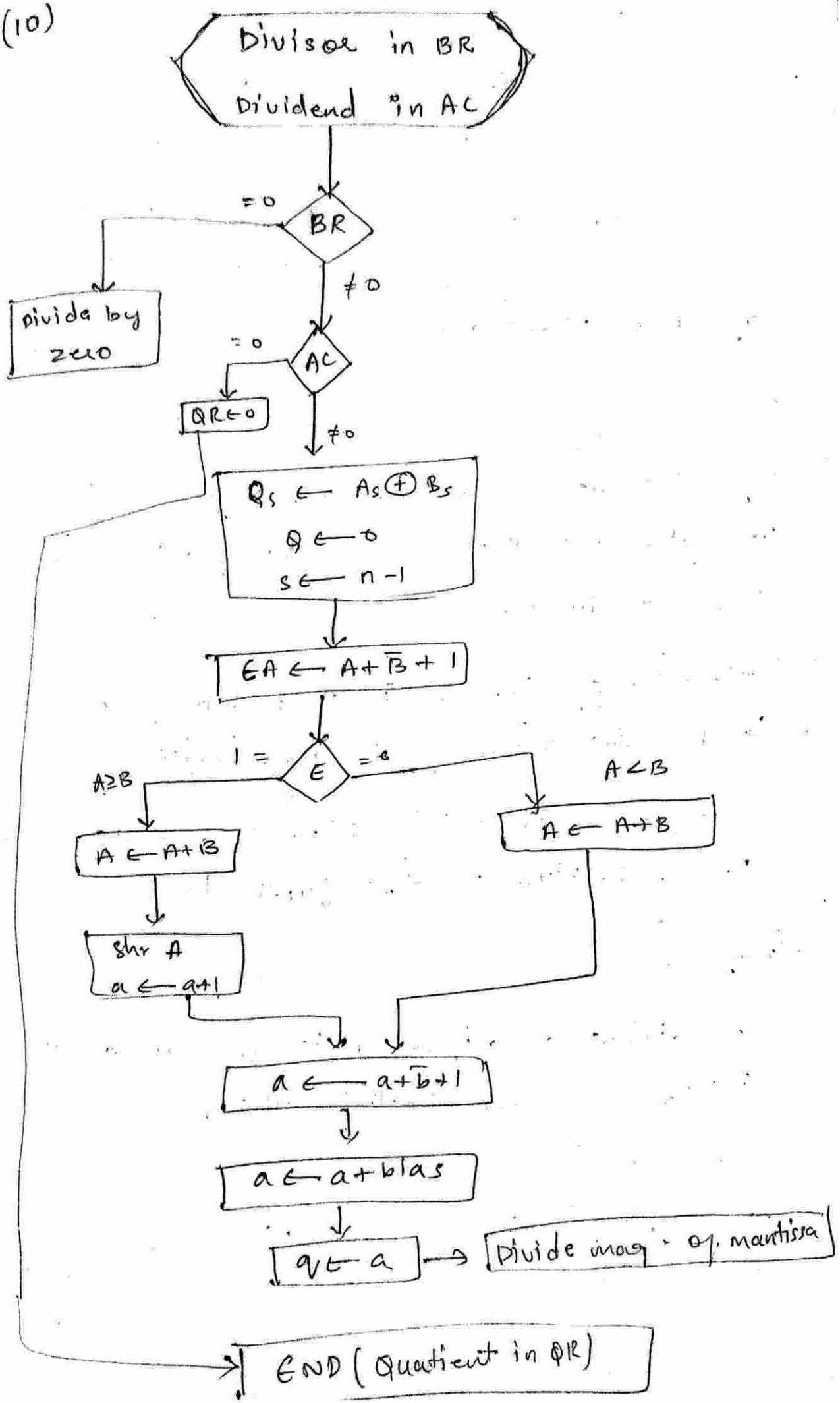
8) If A=0; mantissa in AQ is shifted left and exponent decremented.

9) Final product term is obtained (AC)

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No 5
HYDERABAD - 500 034, A.P

(10)


Divisor in BR
Dividend in AC
BR
=0
≠0
divide by zero
AC
=0
≠0
QR←0
Qs ← As ⊕ Bs
Q ← 0
s ← n-1
EA ← A+B̄+1
E
1 =
=0
A≥B
A<B
A ← A+B̄
A ← A+B
shr A
a ← a+1
a ← a+b̄+1
a ← a+bias
q ← a
Divide mag. of mantissa
END (Quotient in QR)

Step - 1 :

- The two operands are checked for zero
- If the divisor in BR is zero, it indicates an attempt to divide by zero, the quotient in QR is made zero and the operation terminals

Step - 2

- If the operands are non-zero, we proceed to determine the sign of the quotient and store it in Q.
- The sign of dividend in A is left unchanged to be the sign of the remainder
- The Q register is cleared and SC is set to n-1 bits in the quotient.

Step - 3

- The dividend alignment is similar to the divide overflow check in the fixed point representation.
- The two fractions are compared by subtraction.

23/03/2021

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P

- If $\theta = 1$, it determines $A \geq B$, firstly we restore the dividend fraction by adding A with B and then shift right the dividend (A) and increment exponent of dividend (a).

→ <u>Step - 4</u>

- Next the divisor exponent is subtracted from dividend exponent ($a \leftarrow a + \bar{b} + 1$)
- The bias is added and result transferred into a.

→ <u>Step - 5</u>

- The magnitudes of the mantissas are divided as in fixed point case.
- After the operation - The floating point quotient resides in QR.

⇒ Dividend alignment.

- If the dividend is greater than are equal to the divisor the dividend fraction is shifted to the right and the exponent of it is incremented by 1.

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 2: Submission Date: 28-06-2021

1. a) What do you understand by stored program concept? Explain.

   b) With a neat sketch of a common bus system, show how to execute the micro operation AC ← AC + DR.

2. a) List the control functions and micro operations needed for the execution of the following instructions and explain.
   i) ADD ii) BSA iii) ISZ

   b) Explain various phases of an instruction cycle in detail with flow chart.

3. Explain input-output configuration of a computer and list any three I/O instructions with their control functions and micro operations.

4. Describe the various phases of an interrupt cycle and its micro operations.

5. a) Differentiate between Hardwired and Micro programmed control. Mention their merits and demerits.

   b) Explain the operation of a address sequencer in a micro programmed control unit.

6. What is the purpose of micro program sequencer? Explain with Block diagram, how the sequencer present address to control memory

J.K.NAG
Assoc.Prof. (ECED)

23/03/2021

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034 A.P

COA Assignment - 2 1604-19-735-008

Soln Key Sameera Begum

1) (a) The term stored program concept refers to the storage of instructions in computer memory to enable it to perform a variety of tasks in sequence or intermittently.

1.) The ability to store and execute instructions, "The stored program concept", is the most important property of a general purpose computer.

2.) Such a design is reffered to as the "von Neumann architecture" where data & the instructions are stored in the same memory

3.) An instruction code is a group of bits that instruct the computer to perform a specific operation.

4) An instruction code is usually divided in operation code, operand address, addressing mode, etc.

5) The simplest way to organise a computer is to have one processor resistor & an instruction code format with two parts.

6) The 1st part specifies the operation to be performed & 2nd specifies an address.

7.) The memory address tells the control where to find an operand in memory.

8.) The operand is read from memory and used as the data to be operated on together with the data stored in the processor register.

5-1 Instruction codes - stored program organisation

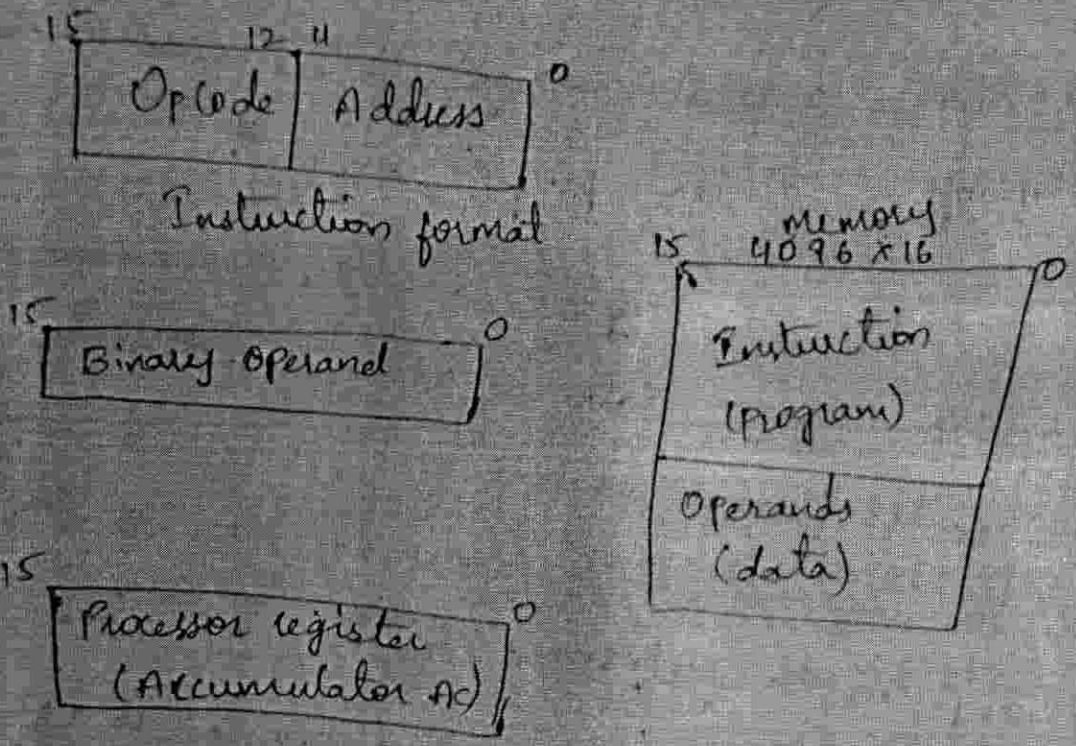


Fig. 5.1 depicts this type of organisation.

9.) Instructions are stored in one section of memory and data in another.

10.) For a memory with 4096 words, we need 12 bits to specify any address, since $12 = 4096 = 4K$ ($K = 1024$)

11.) If we store each instruction code in one 16-bit memory word we have available four bits for the operation.

12.) If then executes the operation specified by the operation code.

13) Computers that have single processor register usually assign to it the name accumulator and label it as AC.

14.) The operation is performed with the memory operated by the content of AC.

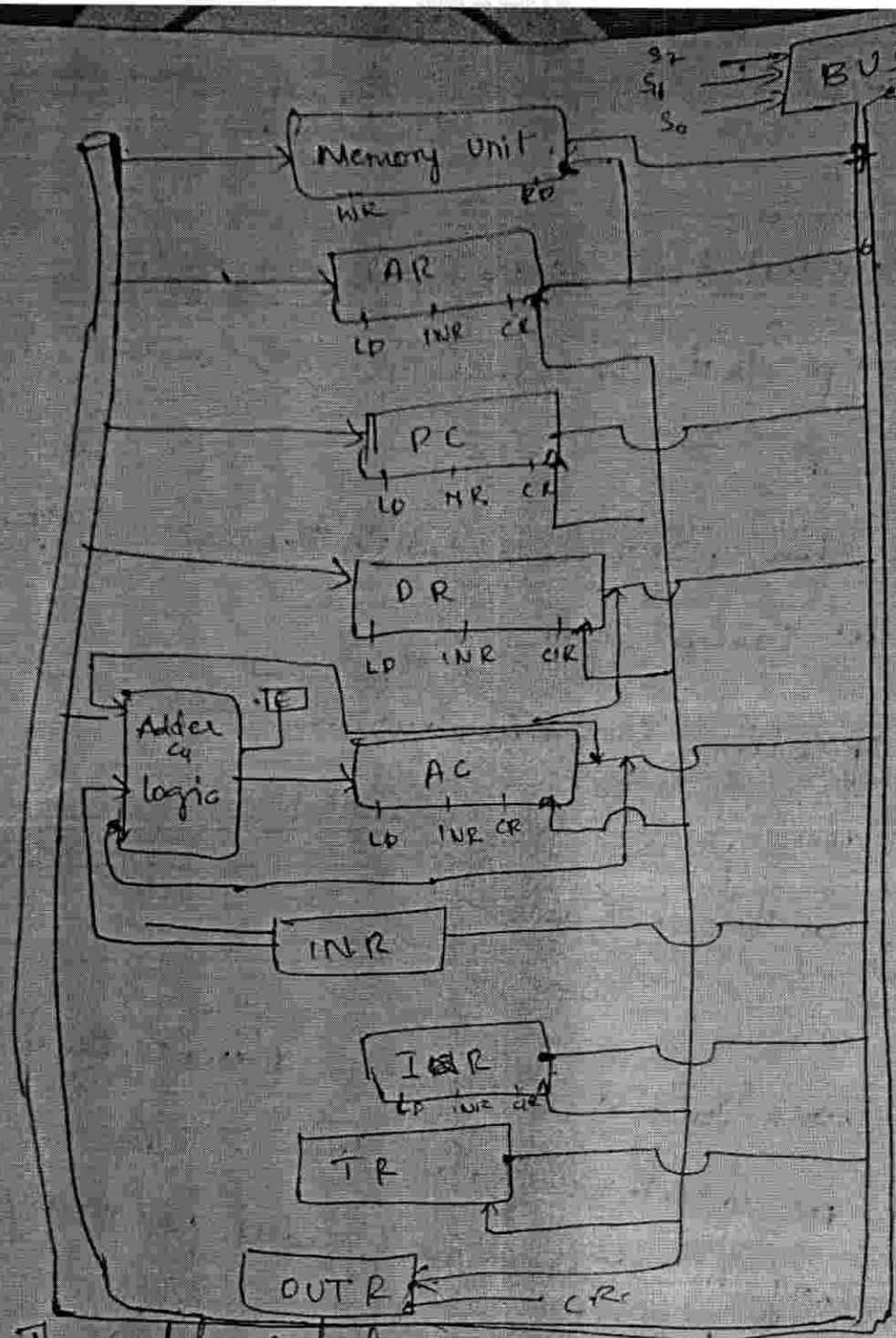
15.) If an operation in an instruction code does not need an operand from memory, the rest of the bits in the instruction can be used for other purposes.

Eg. Operations such as clear AC, complement AC & increment AC operate on data stored in the AC register. They do not need an operand from memory.

1A.) A common bus system mainly used to
(b.)
transfer data from register to register or from register to memory or from memory to register in a less expensive manner.

25/07/2022

PRINCIPAL
Muffakham Jah College of Engineering & Technology,
Banjara Hills, Road No. 3,
HYDERABAD-500 034, A.P.



1.) The adder and logic cell provides the 16 o/ps of AC.

2.) This has 3 sets of inputs.

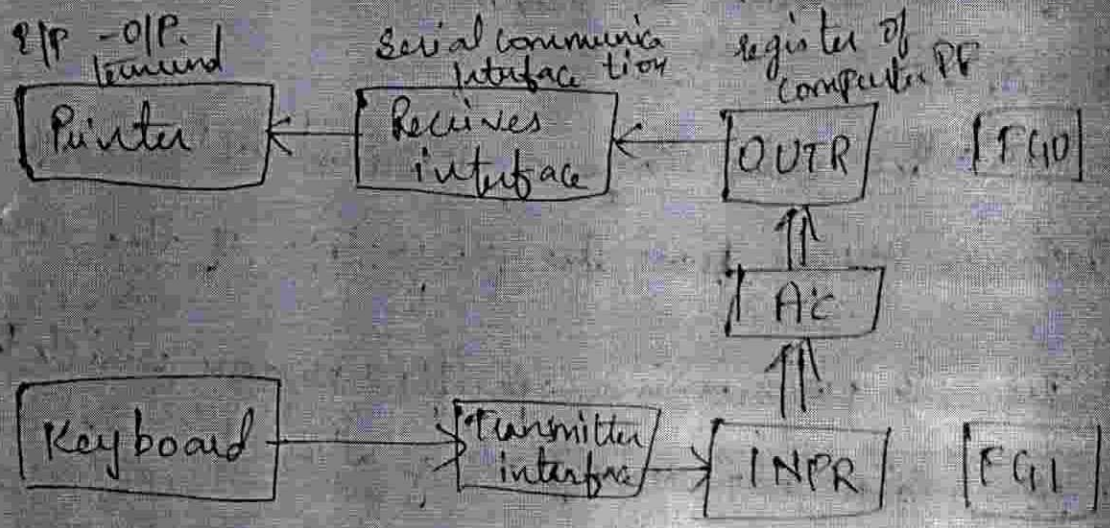
→ 1 set comes from the O/P of AC which implements register micro operations.

→ The other set comes from the DR which are used to perform arithmetic and logic micro

7.) The serial information for the printer is stored in the O/P register OUTR.

8.) They hold an 8 bit O/P numeric information

9.) The transmitter interface receives serial info from the keyboard & transmits to INPR.



→ Serial communication path

⇒ Parallel " "

10.) They receives interfacance receives information from OUTR & sends it to the printer serially.

FG1 : 1 bit i/p flag is a control flip flop aimed to control the i/p operation.

FG0 : 1 bit O/P flag used as a control FF to control the O/P operation.

→ If this an i/o instruction or a register instruction the operation is performed during the clock pulse.

8) Execute stage : The CPU sends the decoded instruction as set of control system to the corresponding computer component.

→ If the instruction involves arithmetic the ALU is utilized.

→ This is the only stage of the instruction cycle that is useful from the perspective of the user.

3A) 1) The I/O processor is a specified processor which load & store data into memory along with the execution of i/o instructions.

2) It involves a sequence of events to executing i/o & then store the results into the memory.

4) It is used to address the problems that are raised in direct memory access method.

5) To demostrate the most basic requirement for i/p & o/p communication we use as an illustration a terminal unit.

Sameera Begum

Assignment - 3

Soln Key

1604-19-735-008

1A.) Instruction format is defined as the layout of instruction bits interms of different fields

* There are 3 types:

1.) Single Accumulator 2) General register organization

3) Stack Organization.

* Single Accumulator Organization:

→ All the instructions are performed with an Implied accumulator register (AC).

→ It uses only one address field i.e the second address will be the AC.

→ The one address field specified the operand address.

Ex: ADD X

AC ← AC + M[X].

* General register Organization:

→ The instruction format in these type of computer needs 3 register address fields.

→ The reg address in these format specifies a specific register of n bits out of 2^n registers.

→ The address register can be reduced to 2, if we specify the destination as one of the

PRINCIPAL
Muffakham Jah College of Engineering & Technology
Banjara Hills, Road No. 3
HYDERABAD

operand register

→ The address fields specified may be a process register or memory word.

* Stack Organization:

→ The stack organized CPU have PUSH and POP instructions require single address field.

→ Arithmetic operations can be performed using stack organization

→ The arithmetic operation in stack organization is performed on the successive two operands in the stack

---

2A) Addressing mode will specify that the given address field is the effective address of the operand or the given address is the address of effective address.

→ Some instructions does it need address field.

→ So, we need addressing mode to know where the actual operand is present.

* Types of addressing modes:

1.) Implied Mode:

→ Operands are specified implicity in the instruction

→ All the zero address field instructions are implied mode instructions.

Ex: LDA, CMA.

2) Immediate mode:

→ The operand is specified in the instruction itself.

→ The address field of the instruction are specified by the value of the operand.

Ex: ADD # 400.

3.) Register Mode:

→ The address field is specified by a processor register.

→ The operands are in processor reg isters.

Ex: ADD R_1, R_2.

4.) Register Indirect Mode:

→ The address field is specified by a processor register which contains the effective address

Ex: ADD (R_4).

5.) Auto INC or Auto Dec Mode:

→ This is similar to reg indirect mode except

the reg is incremented or decremented after or before its value is used to access memory.

Ex: LD (R1)++

LD --(R1).

6) Direct addressing mode:

→ The address field specifies the effective address of the operand.

Ex: LD.X.

7) Relative addressing mode:

→ The address of the operand in the type of mode is the PC + address that is specified in the address field.

EA = PC + address specified.

8) Indexed addressing mode:

→ In this the effective address of the operand is given as index reg + address specified in addr field.

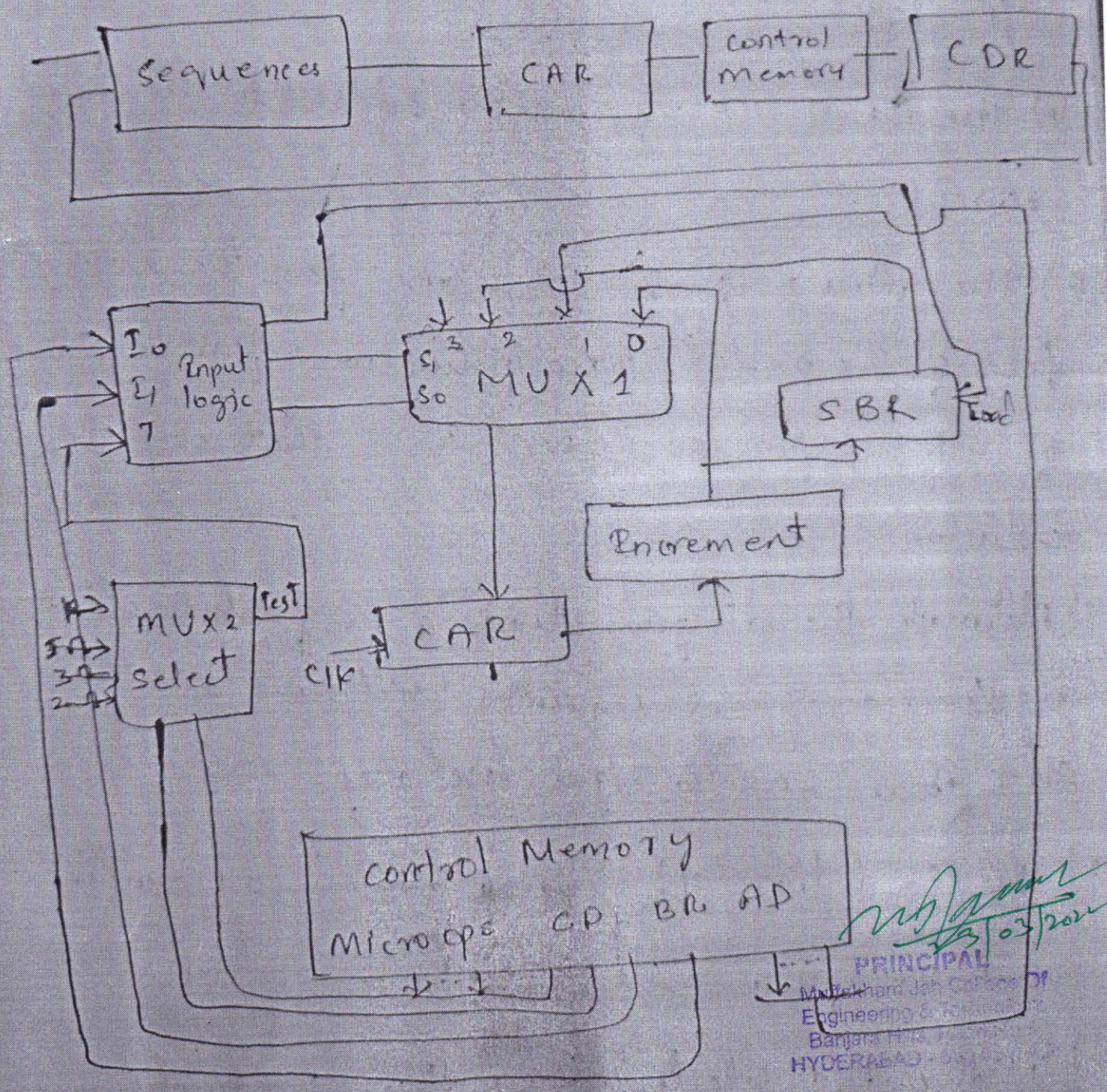
EA = Index reg + address specified.

9) Base register addressing mode:

→ The EA is obtained by adding the content of base register with address specified in the field.

6A) 1.) The basic components of a microprogrammed control unit are control memory & the circuit that select the next address.

2.) The address selection part is called a micro program sequencer.



1.) There are two multiplexers in the ckt.

2.) The 1st mux selects an address from one of 4 sources & routes it into a control address register CAR.

3.) The 2nd MUX tests the value of a selected status bit and the result of the test is applied to an i/p logic ckt.

4) The o/p of CAR provides the address for the control memory.

5) The other 3 i/p to MUX's come from the address field of the present micro instruction from the o/p of SBR & from an external source that maps the instruction.

7.) Although the diagram shows a single subroutine register, a typical sequencer will have a register stack about four to eight level deep.

8.) The Condition (CD) field of the micro instruction select one of the status bits in the 2nd MUX

9.) If the bit selected is = 1, then T variable = 1 otherwise = 0.

7.) Used in computer that makes use of reduced instruction set computer (RISC).	Used in computer that makes use of complex instruction set computer (CISC)

Merits of micro control unit

1.) Less complexity

2.) Cheaper

3.) Flexible

Demerits

1.) Slower than hardwired unit

Merits of hardwired C.U

1.) Faster than micro C.U

2.) It can be optimised to produce fast mode of operation

Demerits.

1.) Occurance of error is more

2.) Complex decoding & sequencing logic.

(b) Steps in Selection of Address

1) Conditional Branching:

→ Interacting from one routine to another depend on status bit conditions

→ Status bits provides parameter information.

→ Information in status bits are tested and actions are initialised based on their conditions 1 or 0

2.) Unconditional Branching:

→ f in value of status bit to 1

Subroutines:

→ A set of common instructions that can be used a program many times

→ Each time a subroutine is used in main program a branch is made to the beginning of subroutine.

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 3: Submission Date: 23-08-2021

1. Explain instruction formats for various types of computer organizations as single accumulator, general register and stack.

2. What is the need for addressing modes? Explain about the various addressing modes in a general purpose computer.

3. An instruction is stored at location 300 with its address field at location 301. The address field has the value 400. A processor register R1 contains the number 200. Evaluate the effective address if the corresponding mode of the instruction is (a) direct; (b) immediate; (c) relative; (d) register indirect; (e) index with R1 as the index register.

4. Convert the following arithmetic expressions from infix to reverse polish notation.

   a. A * B + C * D + E * F b. A + B *[C * D + E *(F +G)]

5. Convert the following numerical arithmetic expression into reverse polish notation and show the stack operations for evaluating the numerical result.

   (3 + 4)[10(2 + 6) + 8]

6. Write the features of RISC Processor. Write the differences between RISC and CISC Processors.

23/03/2022

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034, A.P.

7. Explain any four data manipulation instructions of a general purpose computer.

8. What are the basic differences between a branch instruction, a call subroutine instruction and program interrupt?

9. Draw the space time diagram for a four segment pipeline showing the time it takes to process five tasks.

10. Draw the flow chart for a six-stage CPU instruction pipeline and Explain.

11. A non-pipeline system takes 50 ns to process a task. The same task can be processed in a six-segment pipeline with a clock cycle of 10 ns. Determine the speed up ratio of the pipeline for 100 tasks. What is the maximum speed up that can be achieved?

12. Explain pipeline conflicts and briefly discuss the remedies for those conflicts.

13. How vector processing is different from array processing and give their application areas?

J.K.NAG
Assoc.Prof. (ECED)

(ii) Operand forwarding — 008

(routing the data through special path between pipeline).

→ Soft ware techniques:

(i) Delay loads

3) Branch difficulty:

→ can be avoided by fetching target instruction

→ Branch prediction

→ Delay Branch

→ Branch target buffer included in fetch segment of pipe.

13A) Vector processing includes multiple pipeline modules within an ALU, where as array processor requires.

→ vector processing is faster than array processing

→ Vector processor is cheaper than array processor.

* Applications of Vector processing:

→ long range weather forecasting

→ Medical diagnosis

→ Image processing

→ Mapping the human genuine.

* Applications of Array Processing:

→ Used in scientific complitations in volving two dimensional matrices.

→ Used in nuclear medicine computer systems.

→ Reconstruction of emission tomographic images

→ digital image enhancement.

→ functional image formation.

8A) Pipeline Conflicts:

There are 3 conflicts in the instruction pipeline.

They are:

1) Resource conflicts.
2) Data depending conflicts.
3) Branch difficulty.

1) Resource Conflicts:

→ It occurs when some resource has not been duplicated enough to allow all combinations of instructions in the pipeline to executed.

Ex:

	Clock 1	2	3	4
Instruction 1	FI	DA	FO	EX
2	FI	DA	FO	EX

2) Data dependency:

→ It occurs when the execution of an instruction depends on the result of a previous instructions.

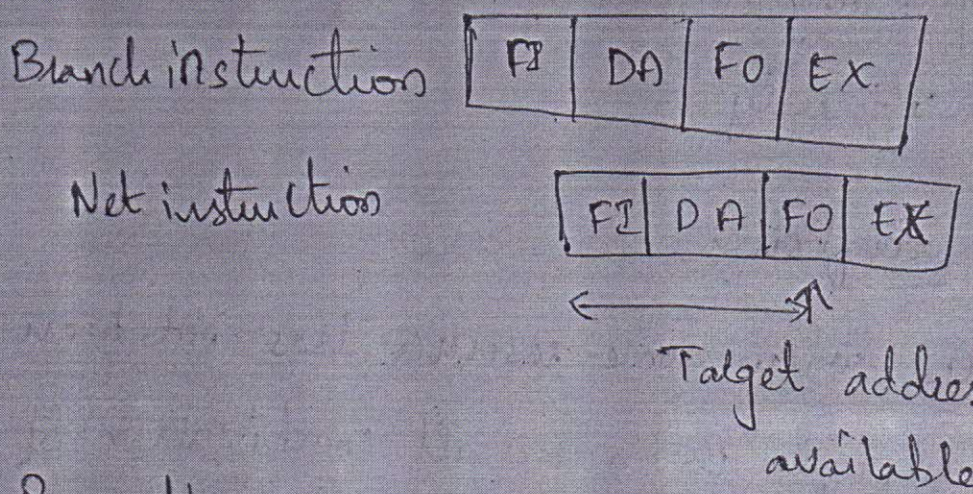
Ex: ADD R_1, R_2, R_3

SUB R_4, R_1, R_5

3.) Branch difficulty:

→ It occurs because branch target address is not known until the branch instruction is completed.

Ex:



* Remedies:

1) Resource conflicts:

→ It can be over come by using seperate memories for data & instructions.

2) Data dependency conflicts:

→ Can be avoided using hardware or software techniques.

→ Hardward techniques

(i) Clock interlocks

(delays the instruction by sufficient clock cycles)

9A.)

Segment \ CLK →	1	2	3	4	5	6	7	8	9
1	T_1	T_2	T_3	T_4	T_5				
2	–	T_1	T_2	T_3	T_4	T_5			
3		–	T_1	T_2	T_3	T_4	T_5		
4			–	T_1	T_2	T_3	T_4	T_5	

Time taken $= (k+n-1)t_p$

k = No. of segments

n = No. of tasks ; t_p = propogation delay.

10A.) The 6 stage CPU instruction pipeline contains following 6 stages.

1.) Fetch the instruction [FI].
2.) Decode the instruction [DI]
3.) Calculate the EA [CA]
4.) Fetch the operand [FO]
5.) Execute the instruction [EX]
6.) Store the result in proper place [WD]

→ From the above space diagram, it takes 9 clock pulses for a 4 instruction to complete 6 tasks.

11A.) non pipeline takes 50ns = t_n

$K = 6$ segments

$t_p = 10ns$

$n = 100$

Speed up ratio = $\frac{n.t_n}{(K+n-1)t_p}$

$= \frac{100 \times 50 \times 10^{-9}}{(6+100-1)10 \times 10^{-9}}$

$= \frac{5000}{105 \times 10}$

Speed up ratio = 4.76

Maximum Speed = $\frac{t_n}{t_p} = \frac{50}{10} = 5$

→ Uses large no. of instructions	→ Uses of less no. of instructions
→ High performance	→ Low performance compared to CISC.

7A.) Data Manipulation Instructions:

1.) ADD R_1, R_2

$R_1 \longrightarrow R_1 + R_2$.

The address field specifies the processing registers R_1, R_2.

2.) AND X

The content of address X is AND (logical AND) with AC and result is stored in AC.

3.) CLRA

The content of the AC is cleared to zero.

4.) LOAD R_1

The address field specifies the address of R_1. The content of R_1 is loaded to AC.

8A) Branch Instruction:

-> when a branch instruction is executed the content of PC is stored in stack, PC is loaded with the address specified by the branch instruction.

-> It may or may not uses status bits for branching.

-> It is written by the programmer.

2) Call Subroutine Instruction:

-> when call subroutine instruction encountered a sequence of instructions are executed and will return to the main program when in the sub routine program.

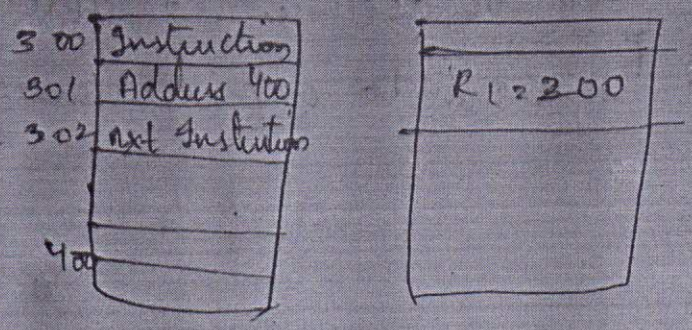
-> It word use the status bits to call a sub routine.

3) Program interrupt:

-> It refers to transfer of program control from a currently running program to another service program.

-> The interrupt address is a vectored address.

3A)



Addressing mode	Effective Address
Direct	400
Indirect	M(400)
Immediate	There no effective address (Data in 400 is specified as # X).
Relative	302 + 400
Register indirect	The data in 400 becomes the EA
Index	400 + 200.

4A) a.) $A^*B+C^*D+E^*F$

b.) $A+B^*[C^*D+E^*(F+G)]$

a.) reverse polish notation

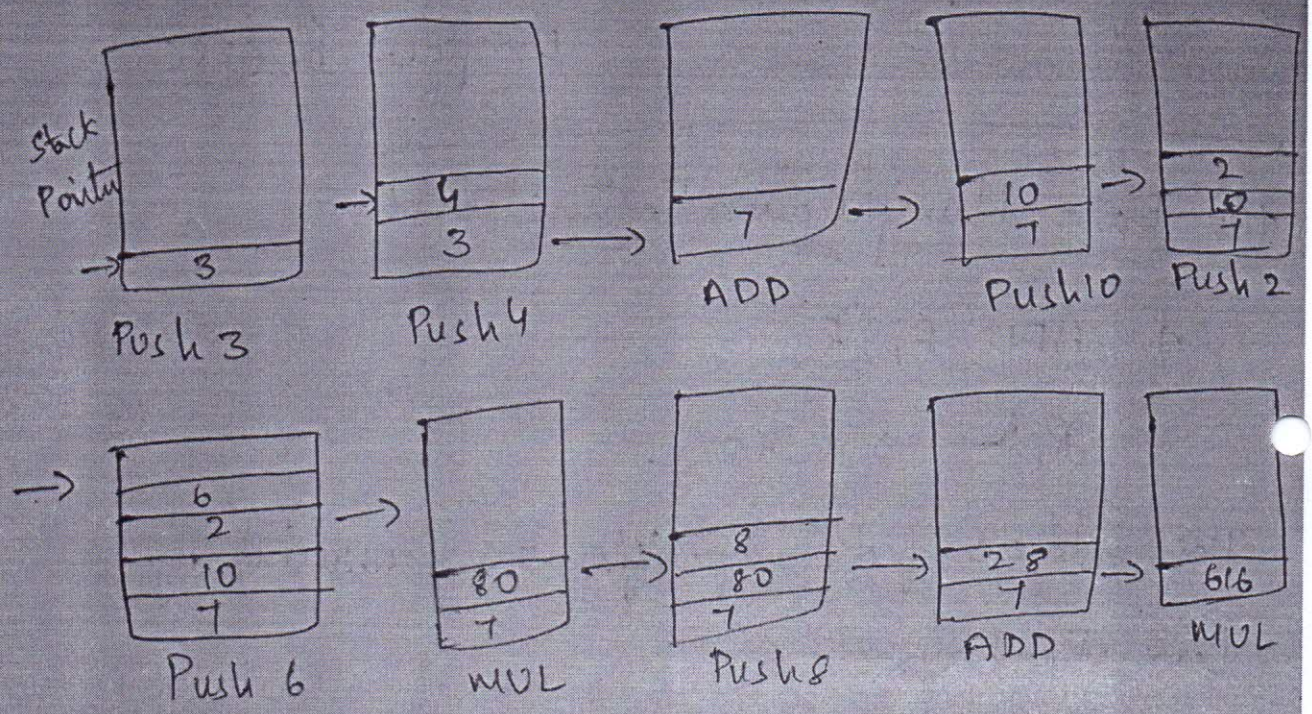
$AB^*CD^*+EF^*+$

b.) $ABCD^*EFG+*+*+$

5A) (3+4) [10(2+6)+8]

Reverse polish notation

34 +1026 + * 8 + *



6A) Features of RISC:

→ Uses less no. of instructions.
→ Uses less addressing modes.
→ Single cycle instruction execution.
→ All operations done with reg in CPU.
→ memory access limited to LOAD and STORE

CISC	RISC
→ Has more no. of addressing modes	→ Has only less no. of addressing modes.
→ Variable instruction length	→ Fixed instruction length.

18.) If FG1 = 0 → No interrupt

19.) If FG0 = 1 → an op interrupt

20.) If then, assigns 1 to 'R' which specifies there is an interrupt.

⇒ If FG0 = 0 → No interrupt.

→ Now the control goes to R.

→ If R = 0, it means that there is no interrupt & the CPU starts fetching other instructions.

→ If R = 1, interrupt cycle occurred.

→ It stores the return address in memory '0'.

→ Now, IEN will be cleared to 0 (IEN = 0) & R will be cleared.

→ That means, we have successfully executed the interrupt related program.

<u>The register transfer statement</u>

T_0, T_1, T_2 (IEN) (FG1, FG0) : R ← 1

RT_0 AR ← 0, TR ← PC

RT_1 M[AR] ← TR, PC ← 0

RT_2 PC ← PC + 1, IEN ← 0, R ← 0, SC ← 0

5 a)

Hard wired control unit	micro programed control unit
1) It generates the control signal needed for the processor using logic circuits	It generates the control signal with the help of micro stored in memory
2) It is faster when compared to micro programed control unit as required control signals.	This is slower than the other as micro instructions are used for generating signals
3) Difficult to modify as the control stands that need to be generated are hard wired	Easy to modify as the modification to be done only at the instruction level.
4) More costlier as everything has to be realized in terms of logic gates.	less costlier than hard wired control as only micro instructions are used.
5) Only limited no. of instructions are used.	Control signals for many instructions generated.
6) It cannot handle complex instructions as the design for it becomes complex.	It can handle complex instructions

operations.

→ The results of those operations is sent to AC while the end around carry is stored in E.

$$AC \leftarrow AC + DR$$

→ AC register stores a value. By adding DR to AC register the sum of AC & DR value again stored in the AC register.

2 Ans.) (a.) This instruction adds the content of the memory word specified by the effective address to the value of AC. The sum is transferred into AC & the output is transferred the E. If the mi Cro operations are needed to execute this instructions are

D_1T_4 : $DR \leftarrow M[AR]$

D_1T_5 : $AC \leftarrow AC + DR$, $E \leftarrow$ Cout, $SC \leftarrow 0$

(b) BSA : Branch & Save return Address.

1.) This instruction is useful for branching to a position of the program called a subroutine or procedure.

2.) when executed, the BSA instruction stores the address of the next instruction in sequence

available input into a memory location specified by the effective address.

3.) This operation was specified with the following register transistor.

$$N\,[AR] \leftarrow PC\,;\ PC \leftarrow AR+1$$

(ii) <u>ISZ</u> : Increment & skip if zero.

1.) This instruction increments the word specified by the effective address & if the incremented value is equal to 0, i.e, is incremented by 1.

2.) The programes usally stores a -ve number in the memory word.

3.) As the -ve number is repeatedly incremented by one, it eventually reaches the value of zero.

5.) This is done with the following sequence of micro operation.

D_6T_4 : $DR \leftarrow M[AR]$

D_6T_5 : $DR \leftarrow DR+1$

D_1T_1 : $M[AR] \leftarrow DR$

if $[DR=0]$ than $[PC \leftarrow PC+1]$, $SC \rightarrow 0$

4A) It is the process by which a computer retrieves a program instruction from its memory determines what actions the instruction requires & carries out those actions.

3.) Firstly, the CPU is fetching an instruction from the memory.

4.) Then, CPU decodes the instructions & the CPU starts executing the instruction.

5.) This can be understood clearly by the flow chart.

6.) From the flow chart, R is an interrupt FF.

7.) If R=1 there is an interrupt signal.
R=0, " " " "

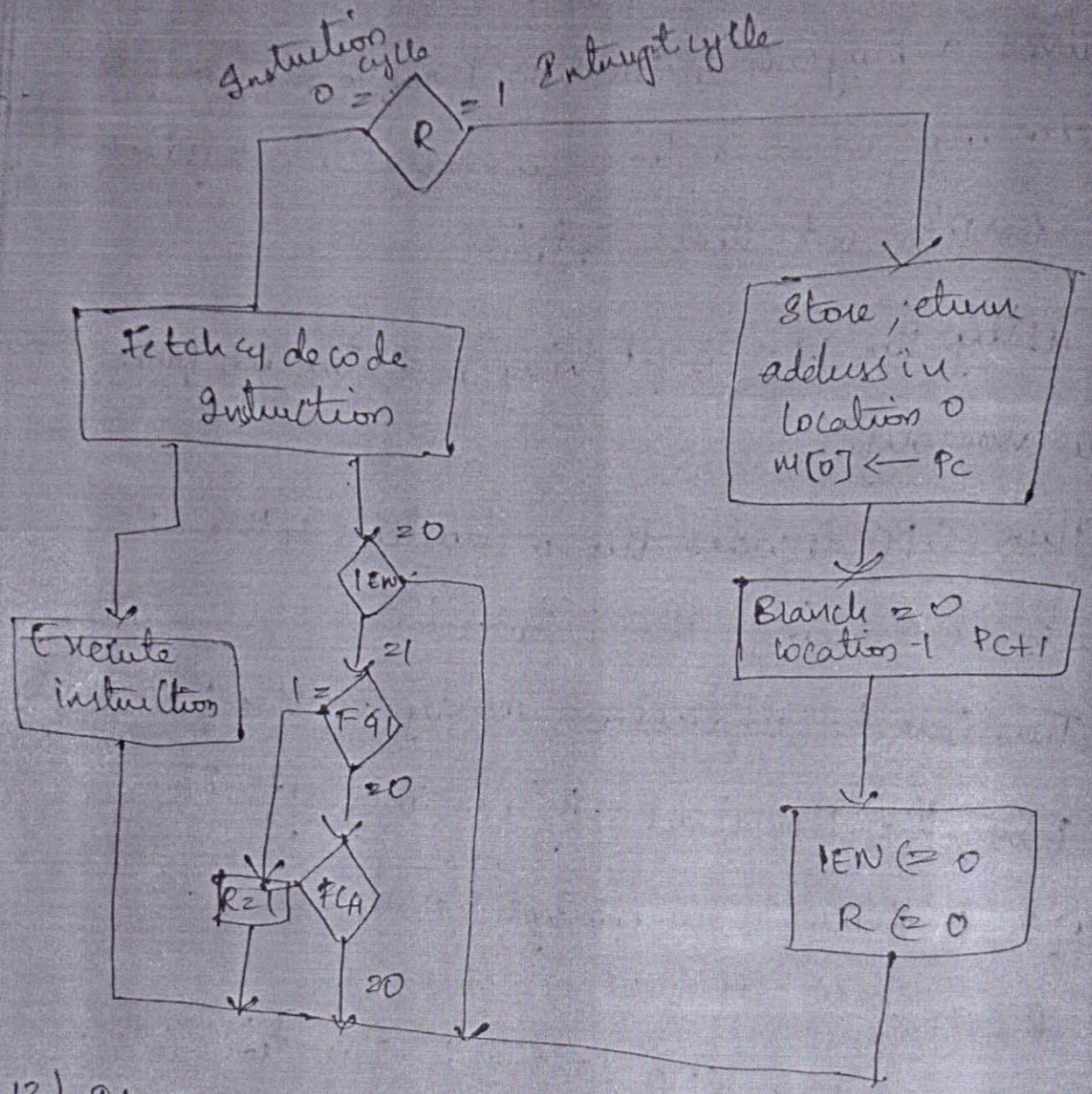
8.) As R=0, the CPU starts instruction code.

9.) The instruction will be fetched from the memory & corresponding instruction is decoded.

10.) Now the instruction is being executed.

11.) For that it checks IEN FF.

23/07/2022
PRINCIPAL
Muffakham Jah College Of Engineering & Technology
Banjara Hills
HYDERABAD



13.) If IEN = 0 → No interrupt

IEN = 1 → An interrupt signal

14.) For that, it checks FGI & FGO FF

16.) If FGI = 1 → an i/p interrupt

17) If assume 1 to 'R' If specifies, there is an interrupt.

b) 1.) A program is a sequence of instruction stored in memory

2.) The program is executed in the computer by going through a cycle for each instruction

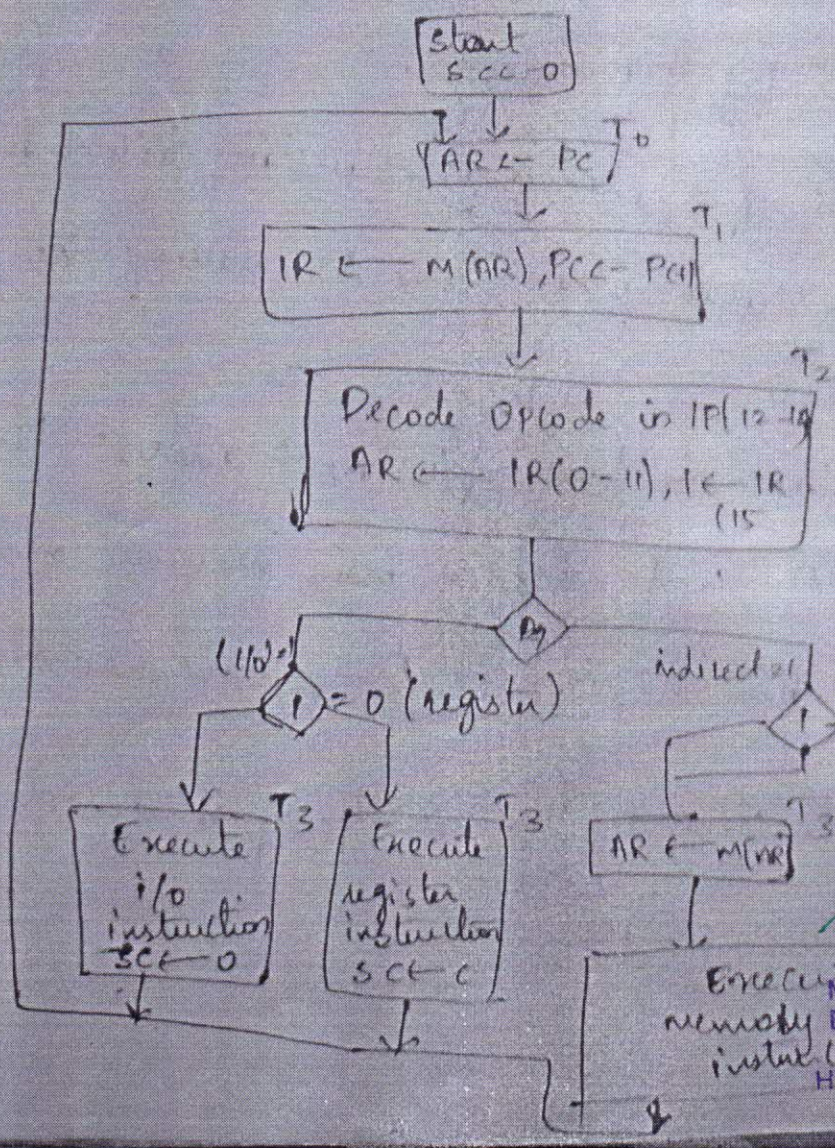
=> Instruction cycle phase:

(i) Fetch an instruction from memory.

(ii) Decode the instruction and calculate effective address (EA).

(iii) Read the effective address from memory if the instruction has an indirect address.

(iv) Execute the instruction.



$D_7 | T_3$: AR ← M[AR]

$D_7 | T_3$: Nothing

$D_7 | T_3$: Execute a register - reference instruction

$D_7 | T_3$: Execute an i/p - o/p instruction.

1.) Fetch stage : The next instruction is fetched from the memory address that is currently stored in the program counter & stored in to the instruction register

2.) Decode Stage : During this stage, the encorted instruction presented in the instruction register is interpreted by the decode.

→ In the case of a memory instruction (direct or indirect the execution phase will be during the next clock pulse.

→ If the instruction has an indirect address, the effective address is read from main memory and any required data is fetched from main memory.

→ If the instruction is direct, nothing is done during the clock pulse

10.) The T value together with the 2 bits from the BR field go to an i/p logic ckt

11.) The i/p logic ckt in a particular sequences will determine the type of operations that are available in the unit

13.) With 3 i/p's the sequencer can provide upto eight address sequencing operations.

14.) The input logic ckt in fig has 3 i/p's I, I, T and three O/P's S, S & L.

15.) Variables S0 & S1 select one of the source address for CAR.

16.) Variable L enables the load input in SBR.

17.) The binary values of the 2 selection variables determines the path in the multiplexer MUX 1.

18.) For example, with S1 = 10,

multiplexer i/p number 2 is selected

19.) establishes a transfer path from SBR to CAR

20.) The bit values for S_1 & S_0 are determined from the stated function and the path in the multiplexer MUX 1 that establishes the required transfer.

21.) The sub routine register is loaded with the incremented value of CAR during a call micro instructions (BR = 01) provided that the status bit condition is satisfied (T=1).

1604-19-735-008

M J C E T
IV-SEMESTER ECE A (2020-2021)

COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 4: Submission Date: 30-08-2021

1. Explain the need for an I/O interface. Show an example of a typical I/O Interface unit.
2. Compare memory mapped I/O and I/O mapped I/O with merits and demerits.
3. What is the disadvantage of transferring data through strobe control method? How handshaking overcomes this disadvantage? Explain.
4. Design parallel priority interrupt hardware for a system with eight interrupt sources.
5. Draw the block diagram of an asynchronous communication interface and explain its operation.
6. Explain the DMA transfer with block diagram.
7. What is an Input-output processor (IOP)? Explain how CPU and IOP communicate with each other through a flow chart.

J.K.NAG
Assoc.Prof. (ECED)

23/03/2021

PRINCIPAL
Muffakham Jah College Of
Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

COA - Assign - 4

Soln Key

Sameera Begum
1604-19-735-008

1A.) The need for an I/O interface:

(i) I/O interface provides a method for transferring information between internal storage & external I/O devices.

(ii) I/O devices can not directly communicate with the CPU due to various differences between them.

* Input devices:

1.) Keyboard
2.) Optical input devices.
3.) Magnetic input devices.
4.) Screen input devices
5.) Analog input devices

* Output devices:

1.) Card puncher, paper tape puncher.
2.) CRT
3.) Printer
4.) Plotter
5.) Analog
6.) Voice.

23/03/2021

*Each peripheral has an interface module associated with it interface.

→ Decodes the device address (device code)

→ Decodes the commands (operation)

→ Provides signals for the peripheral controller

*Example of I/O interface unit.

1.) It consists of two data register called ports, a control register, a status register, bus buffers & timing & control units.

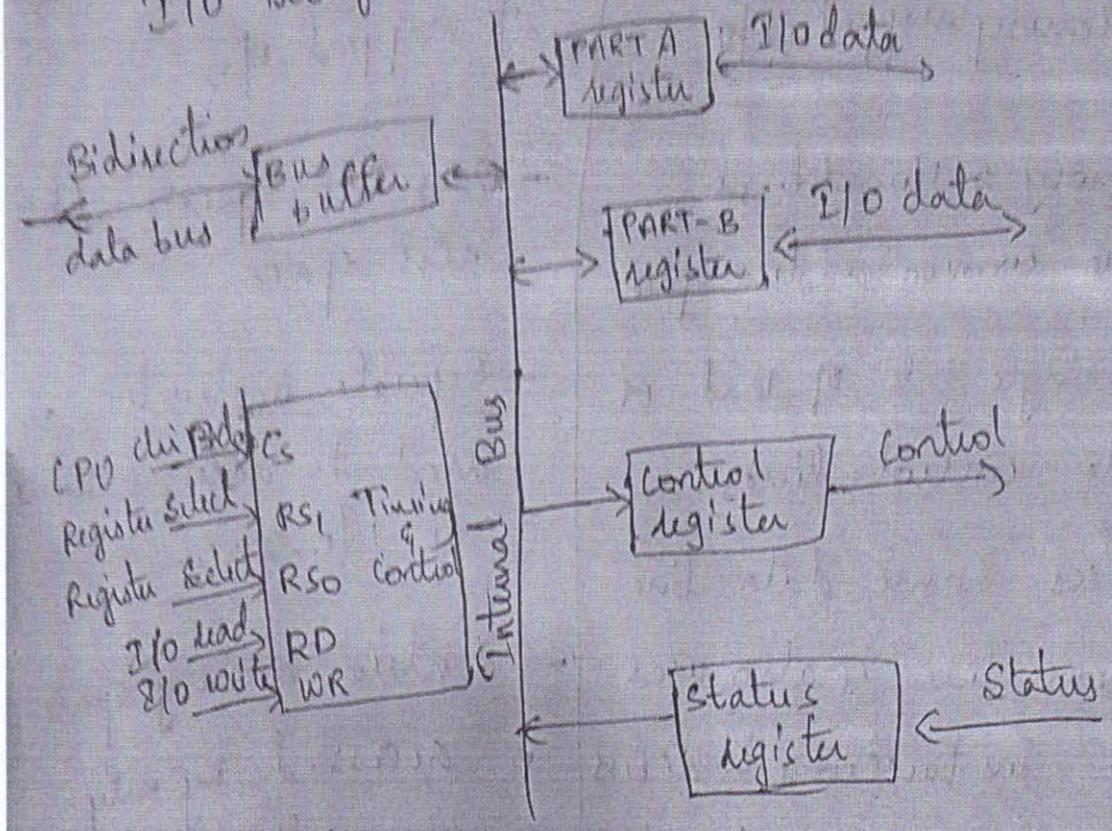
2.) The interface communicates with the CPU through the data bus.

3.) The chip select & register select inputs determine the address to the interface.

5) The four registers communicate directly with the I/O device attached to the interface.

CS	RS_1	RS_0	Register Selected.
0	x	x	None - data bus is high impedence
1	0	0	Port A Register
1	0	1	Port B register
1	1	0	Control register.
1	1	1	Status register.

I/O interface:



* Programmable interface:

→ Information in each port can be assigned a meaning depending on the mode of operation of the I/O device.

→ Port A = Data, Port B = Command, Port C = Status.

→ Programmable port:

By changing the bits in the control register, it is possible to change the interface characteristics.

23/03/2022

2A) Memory mapped I/O

→ Memory & I/O address share common address space

→ A single set of read or write control lines.

→ Uses same data line address line & also uses same instructions to access both memory & I/O address

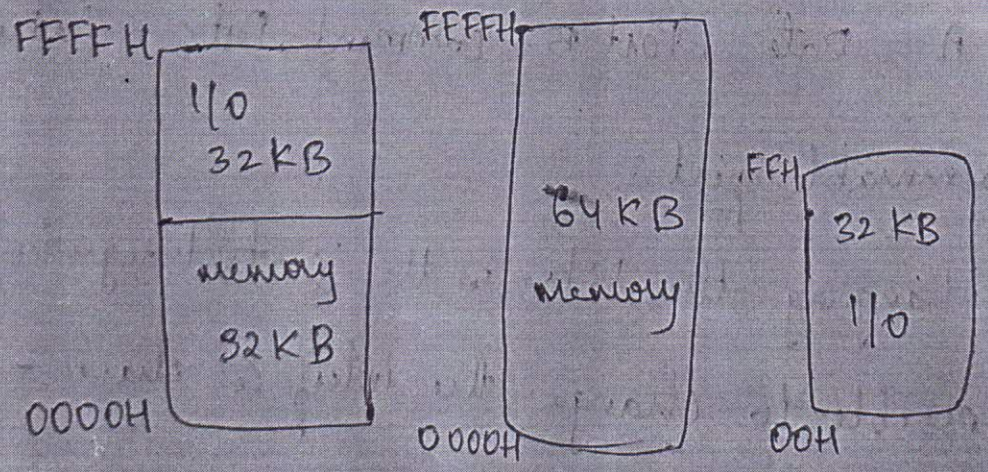
→ Entire memory space can't be accessed by memory unit, it reduces the address range available

I/O mapped I/O

- Uses two seperate address space

- Seperate instructions are used to access memory & I/O space.

- Entire memory space is accessed by only memory unit.

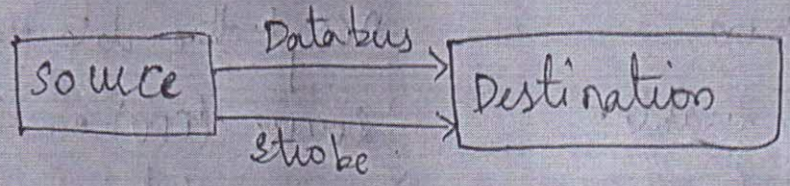


3A) Strobe Control method :

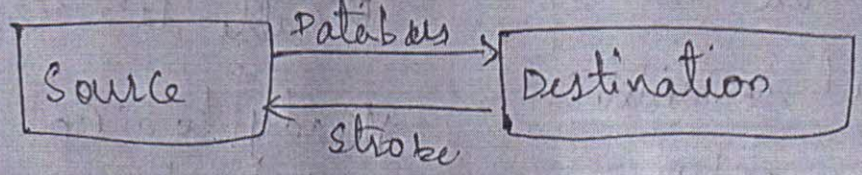
→ In strobe control method a strobe pulse or control signal is sent by source to the destination to indicate when the data transfer has to occur.

→ They can be initiated by either source or destination.

→ Source initiated strobe.



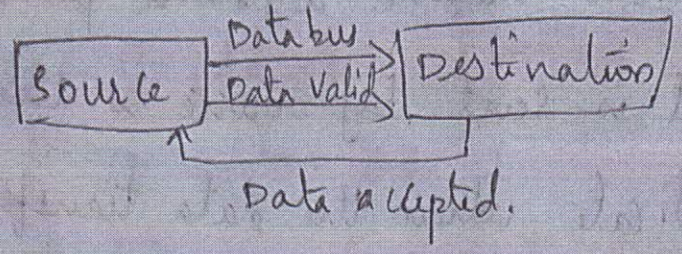
→ Destination initiated strobe:

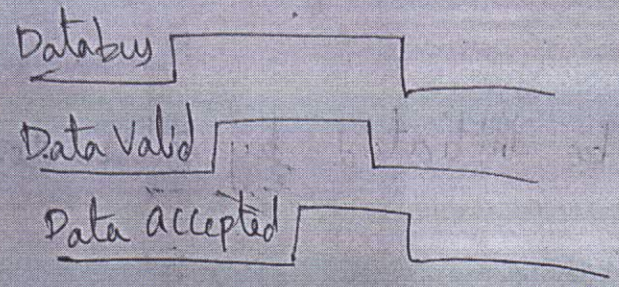


→ To overcome the above demerits we use hand shaking method.

→ In hand shaking method a response signal is sent to source by destination in response of data valid / control signal to initiate the transfer.

→ Source initiated handshaking





Source

Place data on data bus, enable data valid

Disable the data valid, invalidate the data on bus

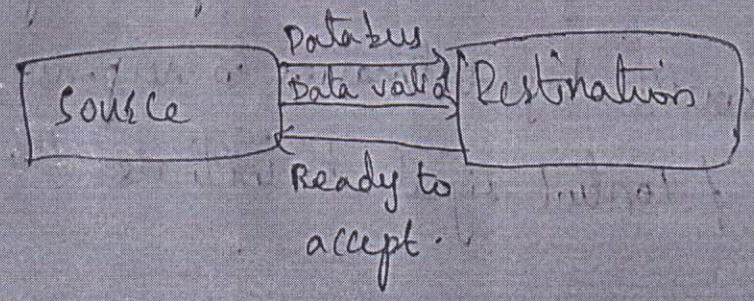
Destination

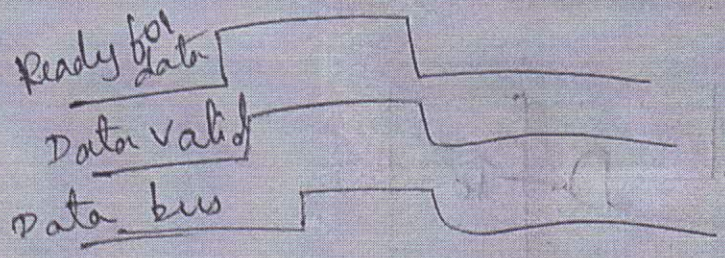
Accept the data from source (CPU) enable data accepted

Disable data accepted

Ready to accept data.

→ Destination initiated handshaking





Source

Place data on data bus &

Enable data valid

Disable valid data

invalidate data on data bus

Destination

Ready to accept data

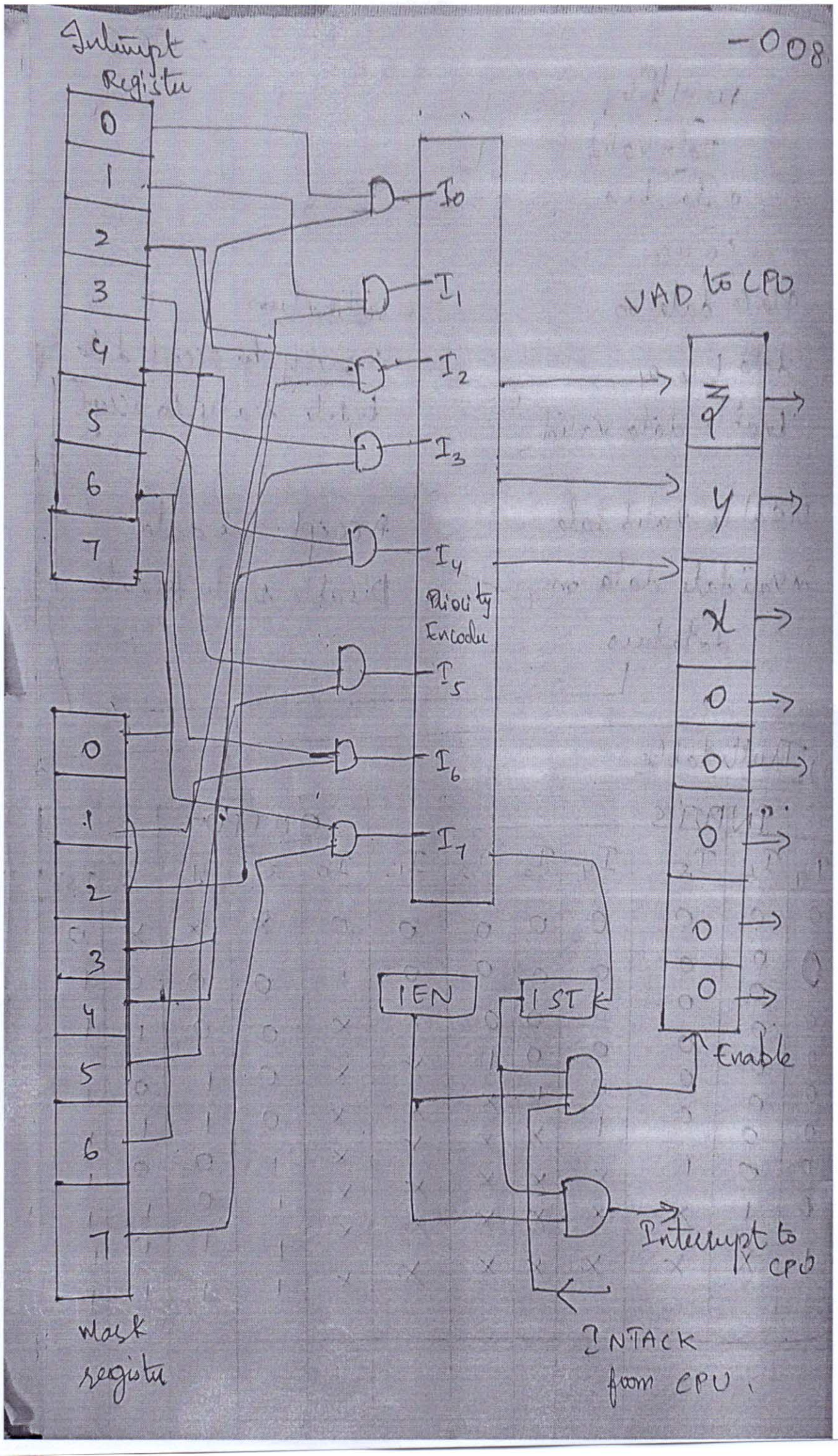
Enable ready to accept

Accept the data

Disable ready for data

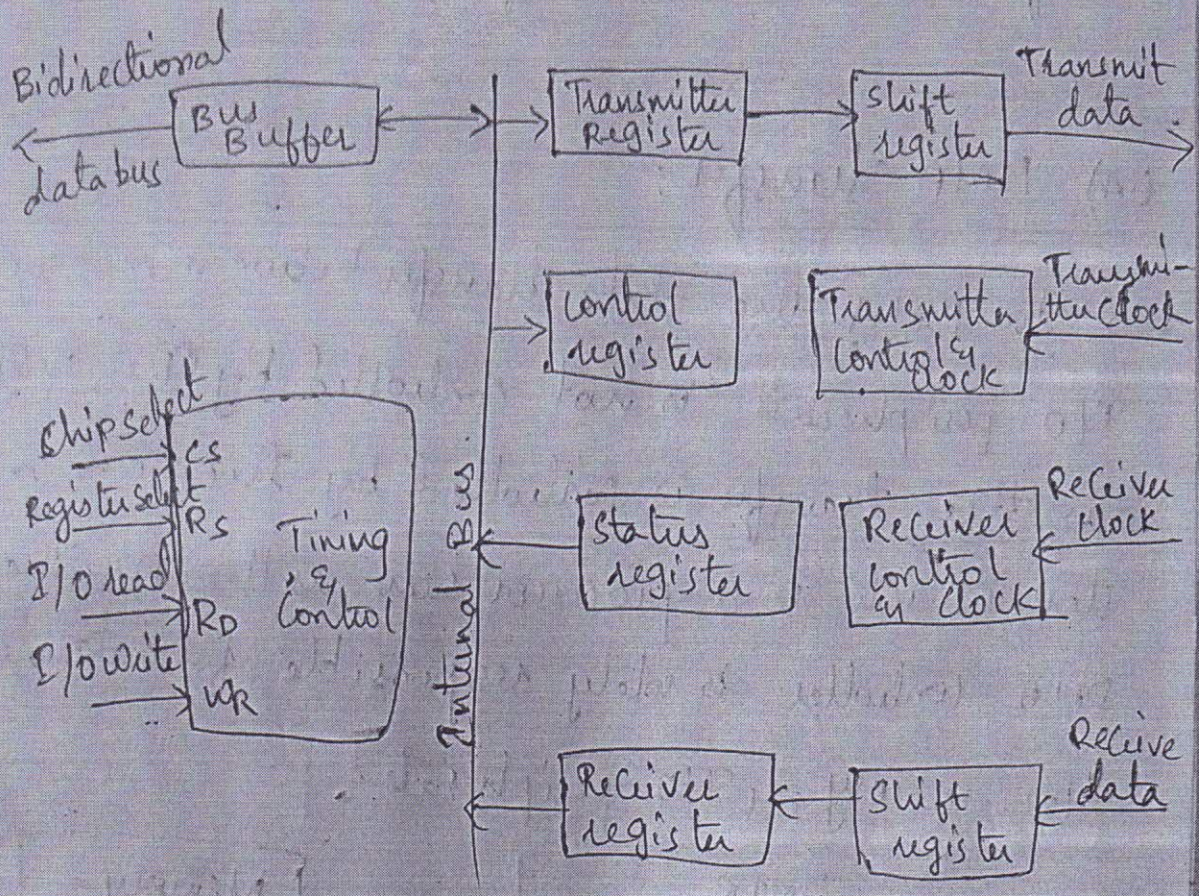
4A) Truth table

INPUTS							OUTPUT				
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	z	y	x	IST
0	0	0	0	0	0	0	0	X	X	X	0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	X	0	0	1	1
0	0	0	0	0	1	X	X	0	1	0	1
0	0	0	0	1	X	X	X	0	1	1	1
0	0	0	1	X	X	X	X	1	0	0	1
0	0	1	X	X	X	X	X	0	0	1	1
0	1	X	X	X	X	X	X	1	1	0	1
1	X	X	X	X	X	X	X	1	1	1	1


Interrupt Register
0
1
2
3
4
5
6
7
I_0
I_1
I_2
I_3
I_4
Priority Encoder
I_5
I_6
I_7
VAD to CPU
z
y
x
0
0
0
0
0
IEN
IST
Enable
Interrupt to CPU
INTACK from CPU.
0
1
2
3
4
5
6
7
Mask register

5A)

Asynchronous Communication interface:



CS	RS	Operation	Register Selected
0	X	X	None! databus is high impedance
1	0	WR	Transmitter register
1	1	WR	Control register
1	0	RD	Receiver register
1	1	RD	Status register

→ Transmitter register:

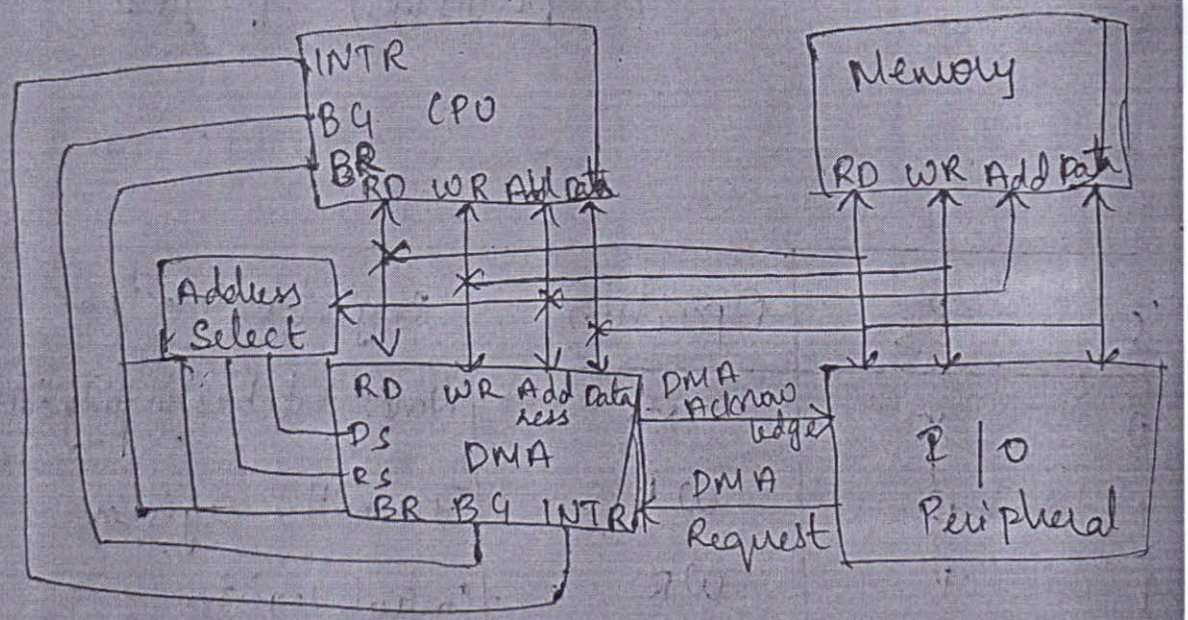
(i) Accepts a data byte through the data bus.

→ Control Register bits:

(i) Define baud rate, no. of bits in each characta.

6A) DMA transfer:

→ In DMA the data transfer between memory & I/O peripherals is not controlled by CPU, where as the transfer is controlled by the DMA controller which is a processor same as the CPU. The DMA controller is solely responsible for data transfer b/w memory & I/O peripherals.



→ Whenever I/O device is ready to transfer data it sends DMA request to controller.

→ The controller sends bus request signal to the CPU when CPU receives BR (Bus request) it

sets bus lines into high impedence state.

→ Now CPU sends bus grant signal in response of bus request signal.

→ When controller recives BG signal, it sends DMA acknowledgement to the I/O device which specifies DMA controller is ready for data transfer.

→ After each word is transferred, address register is incremented by 1 and word count register is decremented by 1.

14.) I/O processor is a processor with direct memory access & I/O devices.

→ It takes care of I/O tasks, releeving the CPU from the burden involved in I/O transfer.

→ Unlike DMA controller, IOP can, fetch & execute its own instructions. It also puforms processing tasks.

23/03/2022

CPU - IOP Communication

CPU	IOP
Send instruction to test IOP path →	Transfer status word to memory location
If status OK send Start I/O instruction to IOP →	Access memory for IOP program
↓	↓
CPU continues with another program	Conducts I/O transfer Prepare status report
↓	↓
Request IOP status ←	I/O transfer completed Interrupt CPU
Check the status for correct transfer ←	Transfer status word to memory location
↓	
Continue	

→ When I/O data transfer is required, first CPU sends a command to the IOP to send the status.

→ The IOP will send the status if it is OK,

→ instruction corresponding to I/O transfer is send to IOP.

→ IOP access memory for the corresponding I/O instruction IOP will transfer the data between memory and I/O.

→ when the transfer is completed IOP interrupts the CPU, CPU requests status of IOP and IOP sends status to specified location & CPU checks the status to ensure the transfer is done perfectly.

23/03/2021

M J C E T
IV-SEMESTER ECE A (2020-2021)
COMPUTER ORGANIZATION AND ARCHITECTURE
ASSIGNMENT SHEET 5: Submission Date: 14-09-2021

1. How many 128 × 8 RAM chips are needed to provide a memory capacity of 4096 bytes? How many lines of the address bus be used to access 4096 bytes? How many lines must be decoded for chip select? Specify the size of the decoders.

2. Derive the Match logic for one word of associative memory and explain how the read operation is performed.

3. Explain the need of memory hierarchy with the help of a Block diagram.

4. Explain Direct and Associative address mapping techniques in Cache memory with an example.

5. Consider a set-associative cache that consists of 256 blocks divided into 4-block sets and a byte addressable main memory of size 512K bytes, with block size of 32 bytes each. How many bits will be there in the TAG, SET and WORD fields respectively?

6. Why page-table is required in a virtual memory system? Draw and explain the organization of typical memory mapping table in a paged system.

7. Using block diagram explain the mapping of logical address into physical address in a segmented-page memory management unit. What is the importance of a TLB in the memory management unit?

J.K.NAG

Assoc.Prof. (ECED)

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034. A.P

COA Assignment - 5

Soln key

Akshaya
16041A0435

1. How many 128x8 RAM chips are needed to provide a memory capacity of 4096 bytes?

Sol:- i) No. of 128 x 8 RAM chips are needed to provide a memory capacity of 4096 bytes = 32

$$\frac{4096}{128} = 32.$$

ii) No of address bus be used to access 4096 bytes = 128×8^7 = 7 address lines

iii) We need 4 lines to select decoded for chip select.

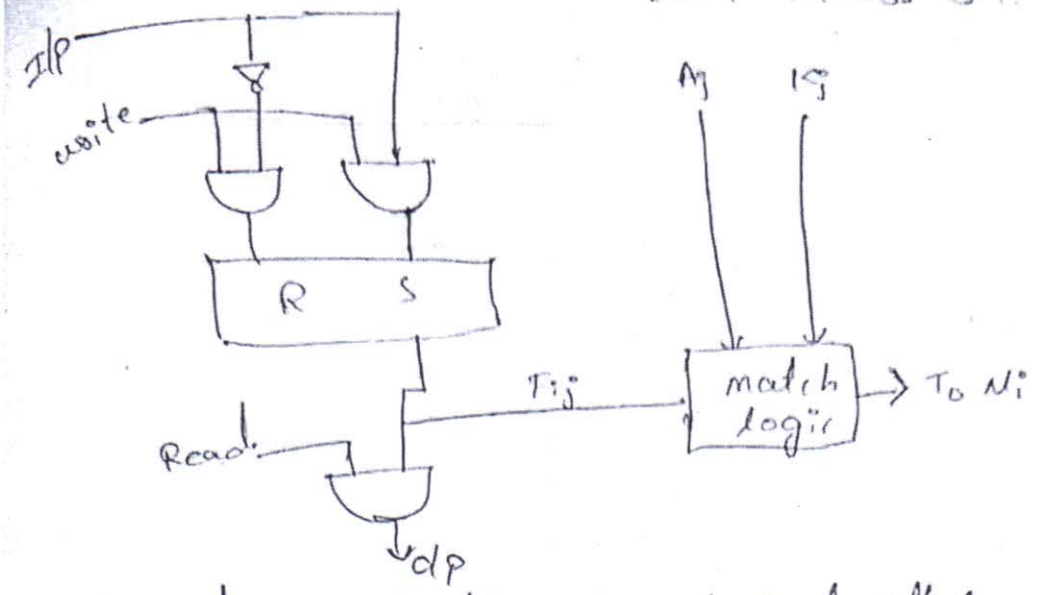
iv) The size of the decoder is,
4 x 32 decoder is needed.

2) Derive the match logic for one word of associative memory & explain how the read operation performed.

The match logic in fig. compares the content of the storage cell with the corresponding unmasked bit of the argument & provides a o/p for the decision logic that sets the bit in M.

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, HYDERABAD - 500 034. A.P
23/03/2022

1604-19-435-306



Internal organization of a typical cell C_{ij}

- The match logic for each word can be derived from the comparison algorithm for 2 binary no.
- 1st, we neglect the key bits & compare the argument in A with the bits stored in the cells of the words.
- Word i is equal to the argument in A if $A_j = F_{ij}$ for $j = 1, 2, \ldots, n$
- Two bits are equal if they are both 1 or both 0.
- The equality of 2 bits can be expressed logically by the Boolean function

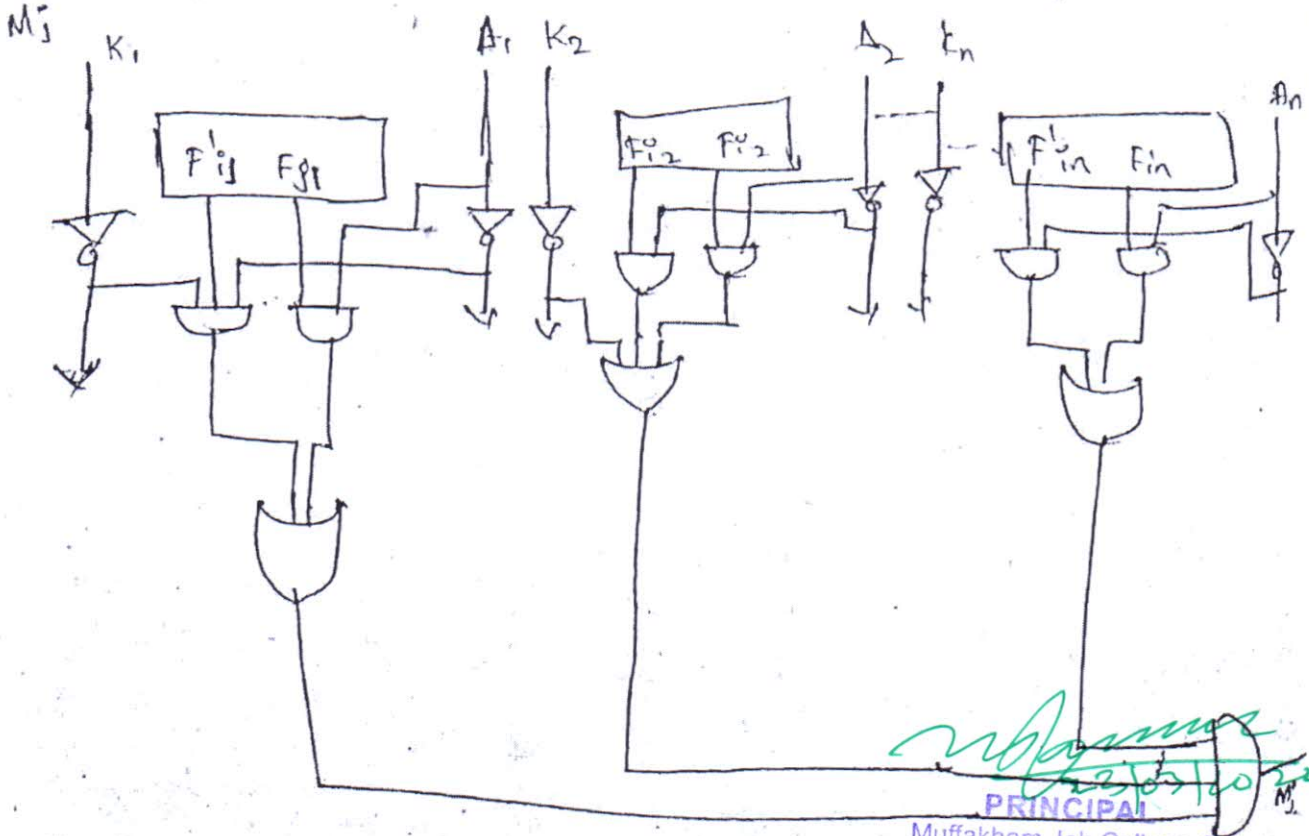
$$x_j = A_j K_{ij} + A'_j F'_{ij}$$

- Where $x_j = 1$ if the pair of bits in position j are equal; otherwise $x_j = 0$
- For a word i to be equal to the argument in A, we must have all x_j variables equal to 1

1604-19-735-30

- This is the condition for setting the corresponding match bit M_i to 1.
- The Boolean function for this condition is

  $M_i = x_1 x_2 x_3 \ldots x_n$ & constitutes the AND operation of all pairs of matched bits in a word.
- We now include the key bit k_j in the comparison logic. The requirement is that if $k_j = 0$, the corresponding bits of A_j & F_j need no comparison.
- Only when $k_j = 1$, must they be compared. This requirements is achieved by ORing each term.



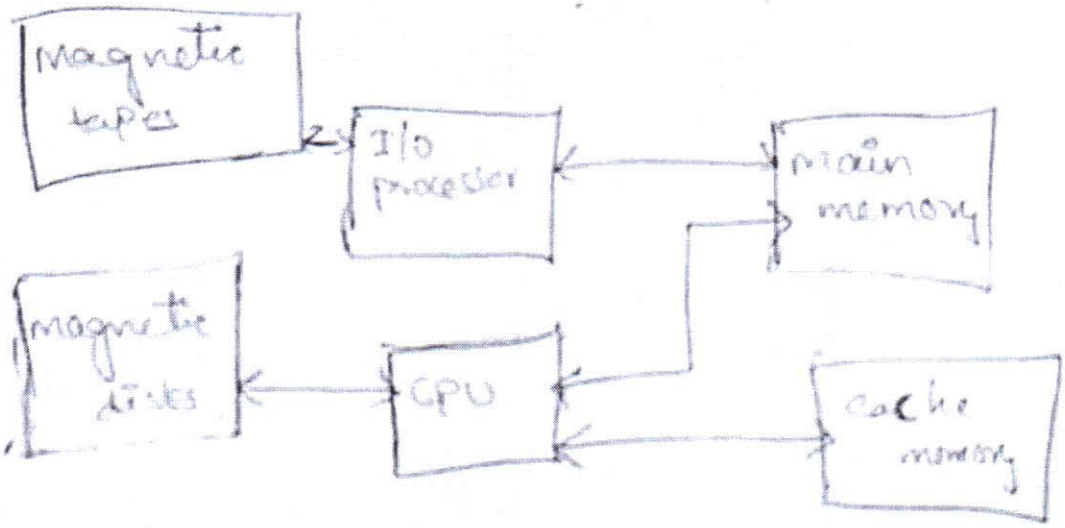
160479-735-504

3) Explain the need of memory hierarchy with the help of a block diagram.

- the memory unit is a essential component in any digital computer since it is needed for storing program & data.
- The memory unit that communicates directly with the CPU is called the main memory.
- Devices that provide backup storage are called auxillary memory.
- the most common auxiliary memory devices used in computer systems are magnetic disks & tapes.
- they are used for storing system programs large data files, & other backup information.
- Only programs & data currently needed by the processor reside in the main memory
- All other information is stored in auxiliary memory & transferred to the main memory when needed.
- The total memory capacity of a computer system can be visualized as being a hierarchy of components.

1604-19-735-304

Auxiliary memory
(secondary)



- At the bottom of the hierarchy are the relatively slow magnetic tapes used to store removable files.
- Next are the magnetic disks used as backup storage
- The main memory occupies a central position by being able to communicate directly with the CPU & with auxiliary memory devices through an I/O processor
- A special very-high speed memory called a cache is sometimes used to increase the speed of processing by making current programs & data available to the CPU at a rapid rate

23/03/2020

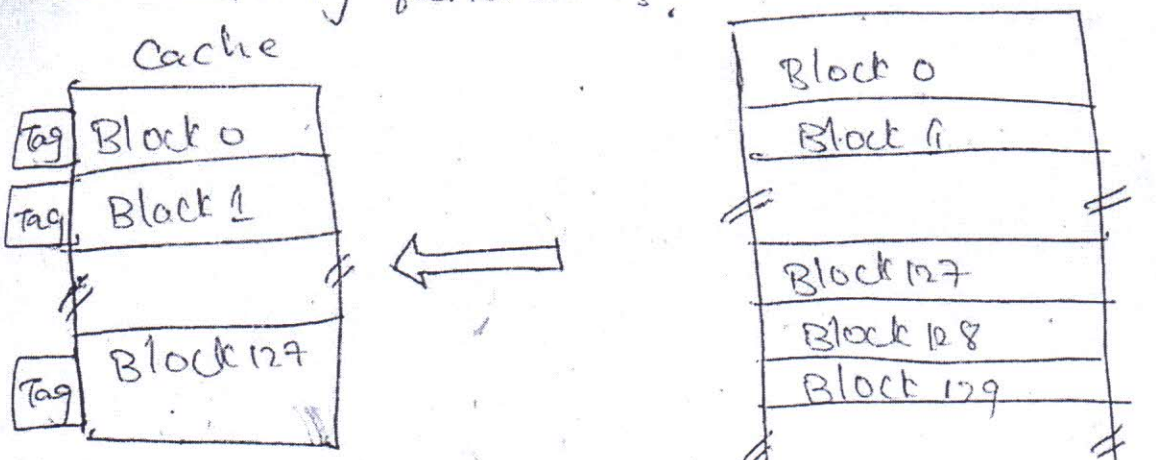
16CL-19-735-304

Explain Direct & Associative address mapping techniques in cache memory with an example

Direct mapping:-

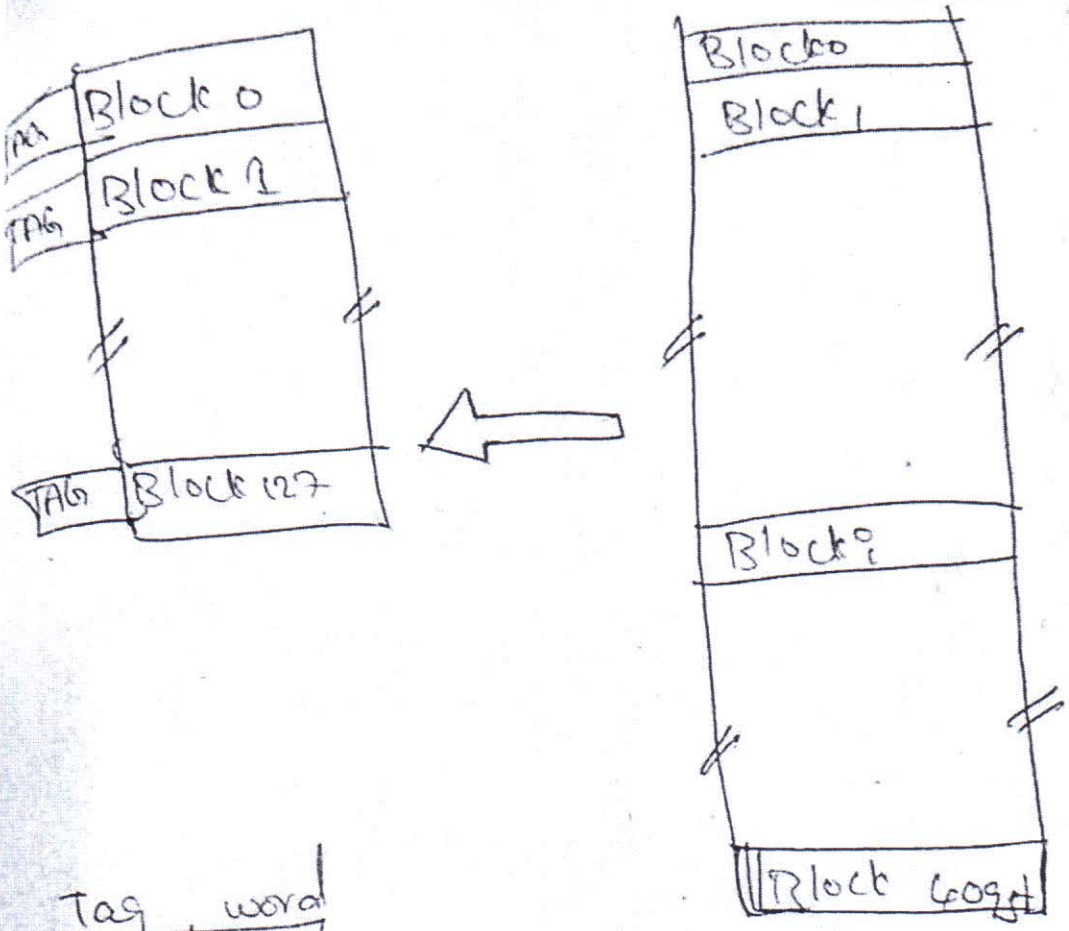
The simplest way to determine cache locations in which to store memory blocks is the direct-mapping technique.

- In direct mapping technique, each main memory block can be placed in only one block in the cache.
- The mapping function is:



Tag	Block	word
5	7	4

main memory address



Tag	word
12	4

main memory address

main memory

- The 12 tag bits identify a memory block residing in the cache.
- The lower order 4 bits select one of 16 words in a block.

23/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology,
Banjara Hills, Road No.3,
HYDERABAD - 500 034 A.P.

1604-19-735-304

- Block j of the main memory maps onto block j modulo 128 of the cache.
- The 7-bit cache block field determines the cache position.
- The high-order 5 to 8 bits identify which of the 32 blocks ~~are~~ currently ~~con~~ resident in the cache.

Ex:-

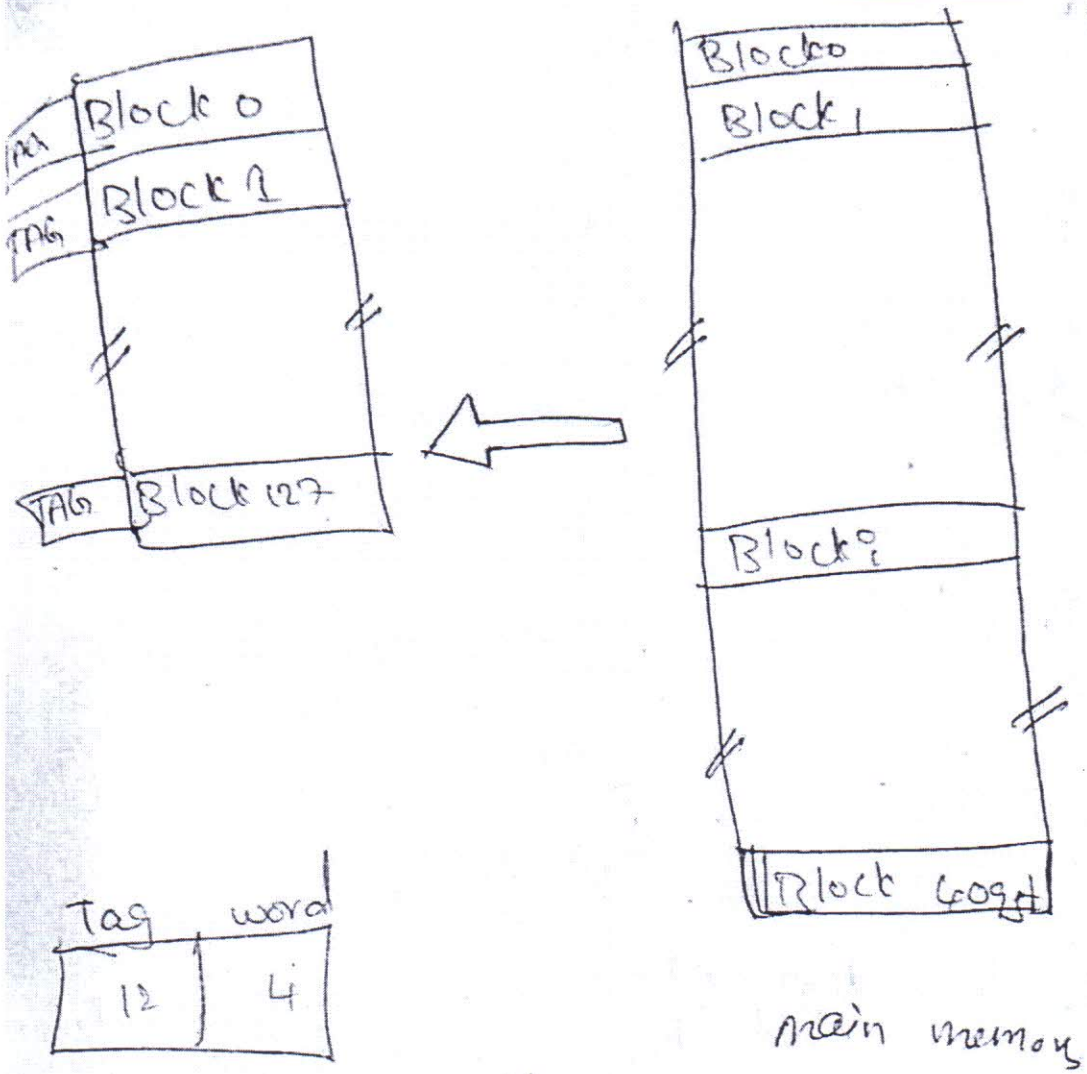
Cache block = (main memory block) % 128

some example mappings:-

0 → 0, 1 → 1, 255 → 255, 256 → 0,
257 → 1, 512 → 0

Associative address mapping:-

- The main memory address is divided into 2 fields: TAG & WORD.
- The cost of an associative cache is relatively high because of the need to search all 128 tags to determine whether a given block is in the cache.



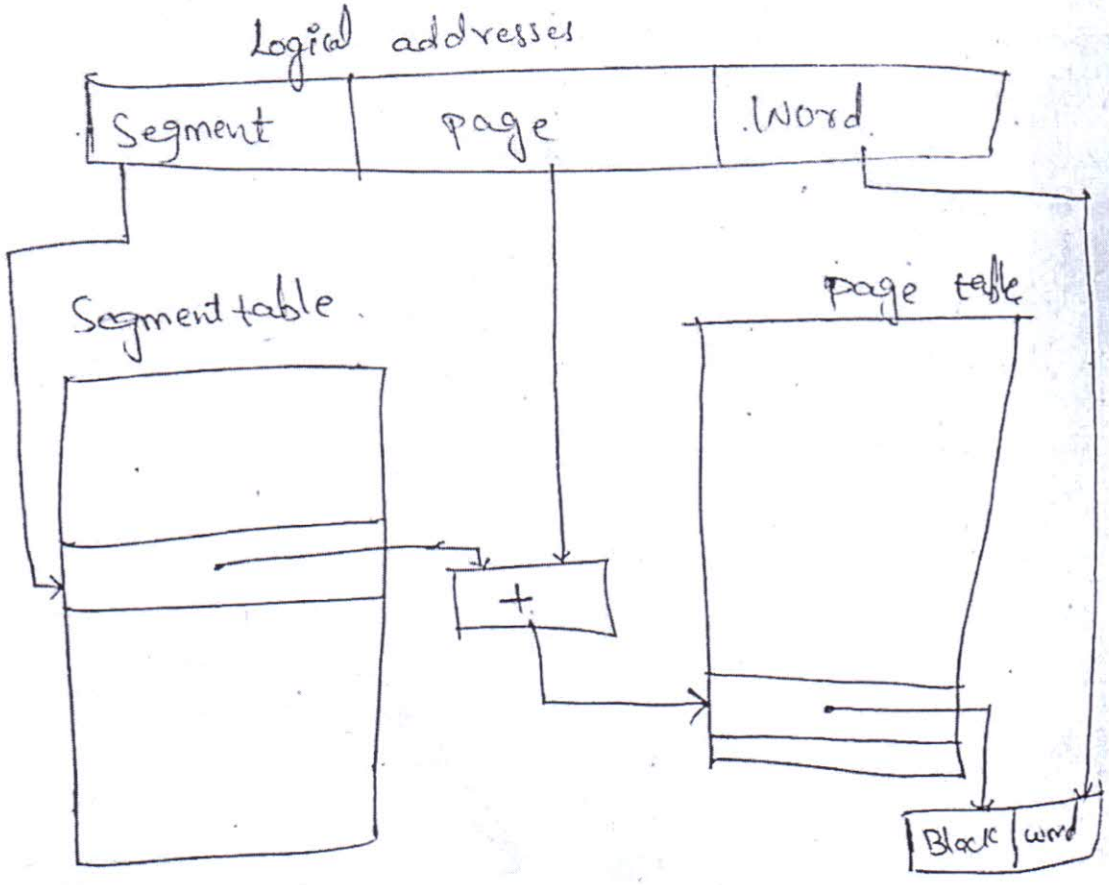
Tag	word
12	4

main memory address

- The 12 tag bits identify a memory block residing in the cache.
- The lower order 4 bits select one of 16 words in a block.

6) Why page table is required in a virtual memory system? Draw & explain the organization of typical memory mapping table in a paged system.

- The length of each segment is allowed to grow & contract according to the needs of the program being executed.
- One way of specifying the length of a segment is by associating with it a no. of equal-sized pages.



logical to physical address mapping

- The logical address is partitioned into 3 fields.
- The segment field specifies a segment no.
- The page field specifies the page within the segment & the word field gives the specific word within the page.
- A page field of k bits can specify up to 2^{k} pages.
- Logical address = segment + page + word.
- A segment no. may be associated with just 1 page or with as many as 2^{k} pages.
- Thus the length of a segment would vary according to the no. of pages that are assigned to it.

1604-19-735-306

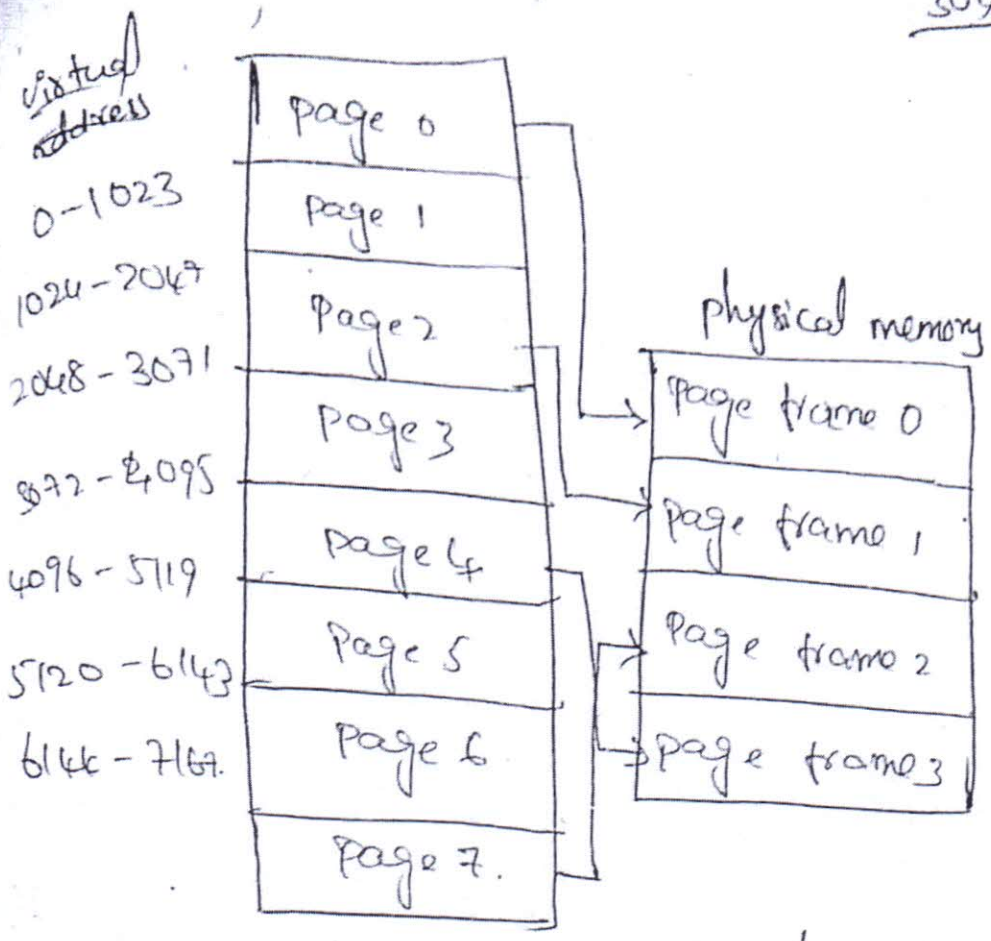
7) Using block diagram, explain the mapping of logical address into physical address in a segmented page memory management unit. What is the importance of a TLB in the memory management unit?

- Paging is a simple technique that translates the virtual addresses into physical addresses by assuming that all programs & data are composed of fixed length units called pages.

- pages are the basic unit of data transfer b/w main memory & the disk & each of them consists of a block of words that occupy contiguous locations in main memory. space

- Length of the page varies from 2K to 16K bytes.

- The page size should not be too small since it increases the access time to fetch more pages from disk.



Physical address

0 - 1023

1024 - 2047

2048 - 3071

3072 - 4095.

- Virtual memory address translation method using the concept of fixed length pages.
- Each virtual address generated by the processor is composed of 2 parts.

23/03/2021

Code No: 15070/BL

FACULTY OF ENGINEERING

BE IV-Semester (ECE) (AICTE) (Main &Backlog) Examination, October 2021

Subject: Computer Organization &Architecture

Time: 2 Hours Max marks: 70

Missing data, if any, may be suitably assumed

PART – A

Note: Answer any five questions. (5x2=10 Marks)

1. Show the hardware for Booth's multiplier.
2. Draw IEEE 754 formats of floating point representation.
3. What is direct and indirect address instructions.
4. What is stored program organization?
5. Distinguish between RISC&CISC processors.
6. What is the role of PC and IR registers?
7. Compare and contrast between synchronous and Asynchronous communication.
8. Explain the need for an I/O interface.
9. Distinguish between SRAM and Associative memory.
10. Explain the terms tag, index and block in relation to cache memory.

PART - B

Note: Answer any four questions. (4x15=60 Marks)

11(a) Draw the flow chart for signed addition/subtraction and also draw the hardware required for that.
(b)Differentiate between Restoring and Non Restoring division algorithm.

12(a) Explain the common bus system of computers with a neat sketch.
(b) Explain various phases of an Instruction cycle in detail.

13(a) Explain possible pipeline hazards with its resolving technique.
(b) What is the basic advantage of using interrupt-initiated data transfer over transfer under program counter without an interrupt?

14(a) Explain the operation of Daisy chaining method of priority interrupt.
(b)What are the merits and demerits of programmed I/O data transfer?

15(a) Explain Set Associative mapping in detail.
(b) Explain the virtual memory in details.

16(a) Describe in detail about IOP organization.
(b) Explain the Interrupt cycle with flow chart.

17(a) Explain the design of control unit. How to decode the micro-operation fields? Explain the process.
(b) Explain Vector processing in detail.

*****

Code No. 15070/BL

BE IV-Sem (ECE) (AICTE) (Main & Backlog) Exam Oct 2021

Subject: Computer Organization & Architecture

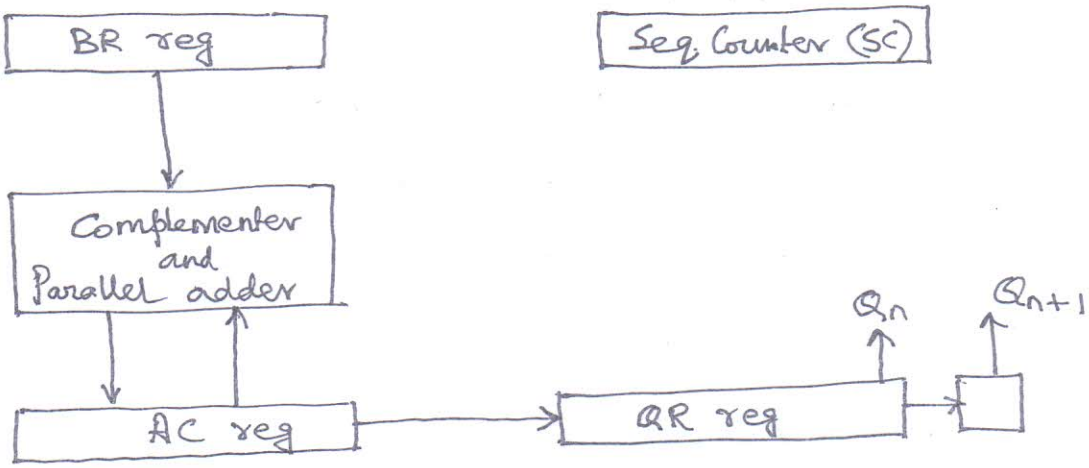
Total: 70

Soln key

PART – A

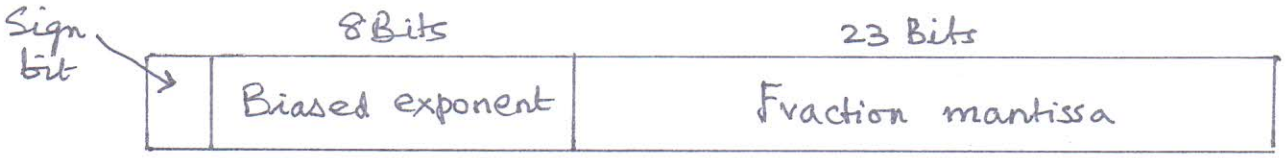
Note: Answer any five questions (5 x 2) = 10 Marks.

① Show the hardware for Booth's multiplier.

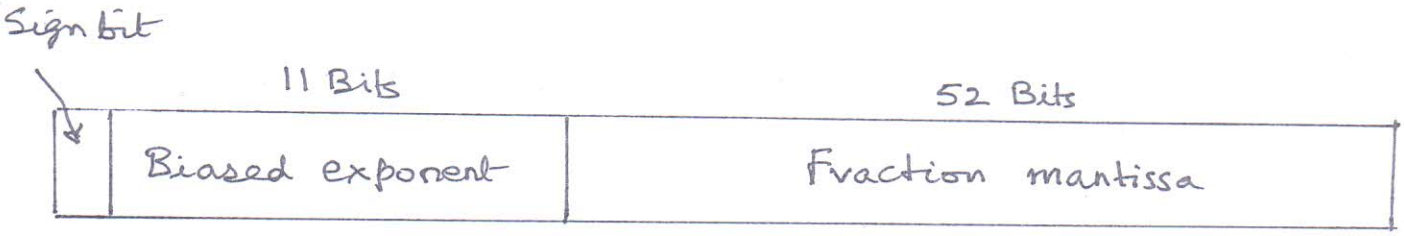


(2)

② Draw IEEE 754 formats of floating point representation.



(a) IEEE 754 Single precision format
32-Bit format



(b) IEEE 754 Double precision Format
64-Bit format

(2)

23/03/2021

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, Hyderabad

PART-A (Contd)

③. What is direct and indirect address instructions

In a direct address instruction, the effective address is equal to the address part of the instruction. The operand resides in memory and its address is given directly by the address field of the instruction.

eg. LD ADR ; AC ← M[ADR]

In an indirect address instrn, the address field of the instrn gives the address, where the effective address is stored in memory. Control fetches the instrn from the memory and uses its address part to access memory again to read the effective address.

Thus the bits in the address part of the instrn designate an address of a memory word in which the address of the operand is found.

Thus, the indirect address instrn needs two references to memory to fetch an operand.

The first reference is needed to read the address of the operand (i.e eff address); the second is for the operand itself.

eg. LD @ADR ; AC ← M[M[ADR]]

for direct memory ref. instrn the I bit = 0

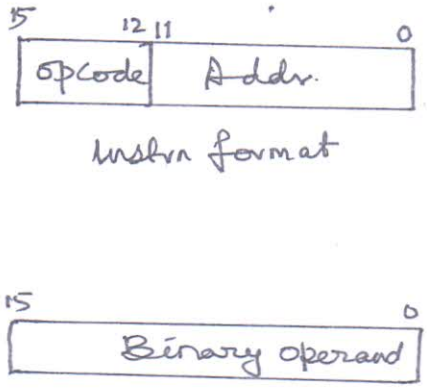
for an indirect addr instrn I = 1

I	Opcode	Addr.

23/03/2021

(4). What is Stored Program Organization?

In a Stored program org attributed Von-Neumann, Program Containing instructions and data are Stored in the Same memory in different Sections.
Here, a Computer Could get its instrns by reading them from memory, and a program Could be set or altered by setting the values of a portion of memory.



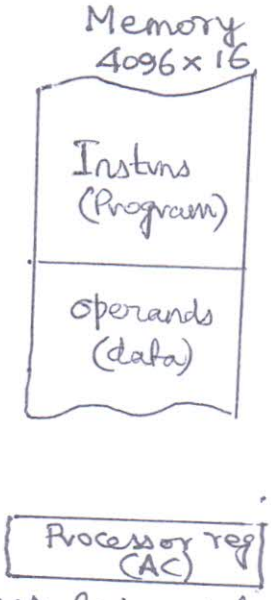


fig- Shows the Stored program organization; where instrns are Stored in one Section of memory and data in another

(2)

(5) Distinguish between RISC and CISC processors.

RISC	CISC
fewer instrns	more instrns
less addressing mode	More addressing modes
requires more RAM	requires minimum amt of RAM
Small Code Size	Large Code Size
fixed instrn format	variable instrn format
multiple reg sets are present	Single reg set
Highly pipelined	less pipelined
focus on software to optimize performance	focus on Hardware to optimize Design

(2)

PART A (Contd.)

Contd

(5)

RISC	CISC
memory access limited to Load and Store instrs	instrns that manipulate operands in memory
Single-Cycle instrn execution	multi-Cycle instrn execu
Fixed-length, easily decoded instrn format	Variable-Length instrn format
Hardwired Control	microprogrammed Control

(2)

(6) What is the role of PC and IR registers?

The program Counter (PC) holds the address of the next instrn to be read from memory after the current instrn is executed. The PC goes thro a a Counting Sequence and Causes the Computer to read Sequential instrns previously Stored in memory.

The instrn read from memory is placed initially in the instruction reg (IR). Thus the IR reg holds the instrn code of the instrn fetched from memory.

(2)

23/03/2022

(7) Compare and Contrast between Synchronous and Asynchronous Communication.

In Synchronous transmission, data is Sent in the form of blocks or frames, the two units Share a Common clock frequency; and bits are transmitted Continuously at the rate dictated by the clock pulses. Synchronization signals are transmitted periodically between the two units to Keep their clocks in Step with each other.

It is more effecient and more reliable than asynchronous transmission to transfer the Large amount of data.

In Asynchronous transmission, data is Sent in the form of byte or character. Here, the binary inf is Sent only when it is available and the line remains idle when there is no inf to be transmitted.

This is in Contrast to Synchronous transmission, where bits must be transmitted Continuously.

In the asynchronous transmission, Start bits and Stop bits are added with data to indicate the start and end of Character transmission.

Synchronous transmission is fast and Costly Compared to asynchronous transmission

In Synchronous tx, there is no gap present between data whereas in asynchronous tx, there is gap between data where the line remains idle during the gap in Character transmission.

(8.) Explain the need for an I/O interface

Input-output interface provides a method for transferring inf. between internal storage and external I/O devices. In order to resolve the differences between peripheral devices and CPU, there is a special need for communication links. Each I/O interface decodes the address and control received from the I/O bus, interprets for peripherals and provides signals for the peripheral controller.

It also synchronizes the data flow and supervises the transfer between peripheral and processor. Each peripheral has its own controller.

(2)

(9) Distinguish between SRAM and Associative memory.

SRAM is a fast and small memory consisting of internal ffs that store binary inf and data is accessed by its address. it reduces the average memory access time. its basic characteristic is its fast access.

A memory unit accessed by its content is called associative memory. it reduces the time reqd to find the item stored in memory.

Here data is accessed by its contents. Hence it is also called as content-addressable memory CAM.

it is used in apps where search time is very short. its basic characteristic is its logic circuit for matching its content.

27/03/2022

(2)

(10) Explain the terms tag, index and block in relation to Cache memory.

The above terms are used for Direct mapping Cache org. The memory address referenced by CPU is divided into ~~three~~ Two fields namely Tag and index.

The number of bits in the in the index field is equal to the number of address bits reqd to access the cache memory

In general case, if there are 2^k words in cache memory and 2^n words in main memory; the n-bit address is divided into two fields: k bits for the index field and (n-k) bits for the tag field

The direct mapping Cache org uses the n-bit address to access the main memory and the k-bit index to access cache.

Each word in Cache consists of the data word and its associated tag.

The tag field of the address indicates how many blocks of main memory can be mapped to a particular block of Cache.

If more no. of blocks of main memory are to be mapped into Cache then the index field is divided further into two subfields: Block and word.

The block field selects the particular block of the Cache where word from main memory is to be placed.

PART B (Any Four) 15 × 4 = 60

(11) (a) Draw the flowchart for signed addition/subtraction and also draw the hardware required for that.

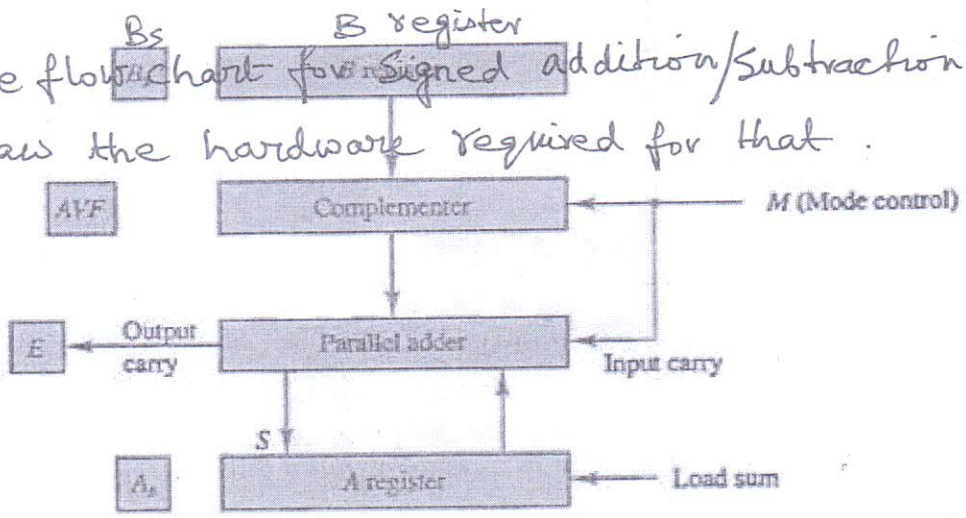


Figure 10-1 Hardware for signed-magnitude addition and subtraction.

the adder is equal to the sum $A + B$. When $M = 1$, the 1's complement of B is applied to the adder, the input carry is 1, and output $S = A + \overline{B} + 1$. This is equal to A plus the 2's complement of B, which is equivalent to the subtraction $A - B$.

Hardware Algorithm

The flowchart for the hardware algorithm is presented in Fig. 10-2. The two signs A_s and B_s are compared by an exclusive-OR gate. If the output of the gate is 0, the signs are identical; if it is 1, the signs are different. For an *add* operation, identical signs dictate that the magnitudes be added. For a *subtract* operation, different signs dictate that the magnitudes be added. The magnitudes are added with a microoperation $EA \leftarrow A + B$, where EA is a register that combines E and A. The carry in E after the addition constitutes an overflow if it is equal to 1. The value of E is transferred into the add-overflow flip-flop AVF.

The two magnitudes are subtracted if the signs are different for an *add* operation or identical for a *subtract* operation. The magnitudes are subtracted by adding A to the 2's complement of B. No overflow can occur if the numbers are subtracted so AVF is cleared to 0. A 1 in E indicates that $A \geq B$ and the number in A is the correct result. If this number is zero, the sign A_s must be made positive to avoid a negative zero. A 0 in E indicates that $A < B$. For this case it is necessary to take the 2's complement of the value in A. This operation can be done with one microoperation $A \leftarrow \overline{A} + 1$. However, we assume that the A register has circuits for microoperations *complement* and *increment*, so the 2's complement is obtained from these two microoperations. In other paths of the flowchart, the sign of the result is the same as the sign of A, so no change in A_s is required. However, when $A < B$, the sign of the result is the complement of the original sign of A. It is then necessary to complement A_s to obtain

complement and increment

03/2022

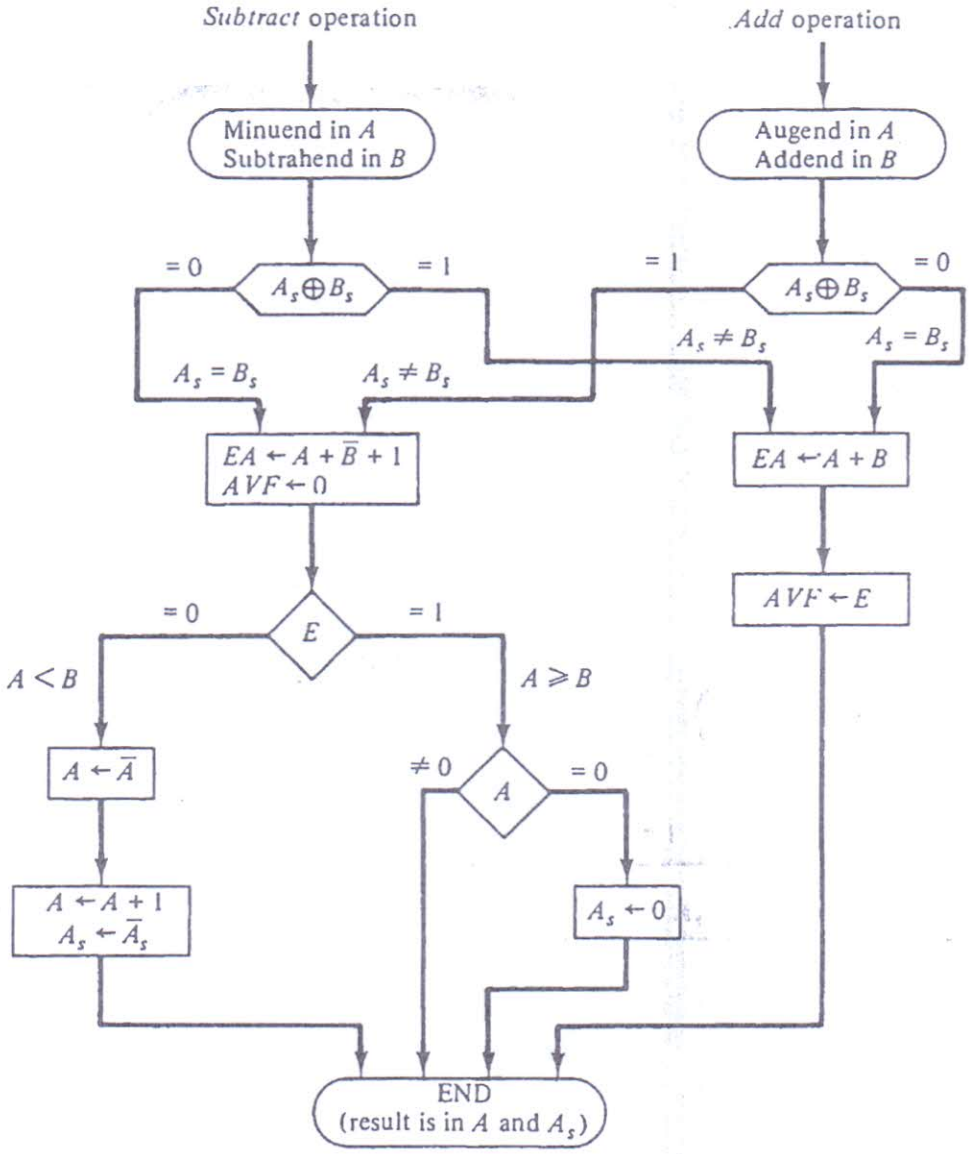


Figure 10-2 Flowchart for add and subtract operations.

the correct sign. The final result is found in register A and its sign in A_s. The value in AVF provides an overflow indication. The final value of E is immaterial.

Addition and Subtraction with Signed-2's Complement Data

The signed-2's complement representation of numbers together with arithmetic algorithms for addition and subtraction are introduced in Sec. 3-3. They are summarized here for easy reference. The leftmost bit of a binary number represents the sign bit: 0 for positive and 1 for negative. If the sign bit is 1, the entire number is represented in 2's complement form. Thus +33 is represented

(11) b) Differentiate between Restoring and Non Restoring division algorithm.

In the restoring method, the partial remainder is restored by adding the divisor to the negative difference.

Initially the divisor is stored in the B reg and the double-length dividend is stored in regs A and Q.

The dividend is shifted Left and the divisor is subtracted by adding its 2's complement value.

The inf about the relative magnitude is available in E.

if $E=1$, it signifies $A \geq B$. A quotient bit 1 inserted into Q_n. and Partial remainder is shifted left to repeat process.

if $E=0$, it signifies $A<B$, So quotient in Q_n remain 0.

The value of B is then added to the negative difference $(A-B)$ to restore the partial remainder in A to its previous value. ie. $(A-B)+B$.

In the non restoring method, B is not added if the difference is negative but instead, the negative difference is shifted Left and then B is added. ie $2(A-B)+B$
$= 2A-B$

in Case of restoring method we get the Same result as follows: $2(A-B+B)-B = 2A-B$

The next time around the loop, this no. is shifted left & B subtracted again.

∴ in the non restoring method, B (divisor) is subtracted if the previous value of Q_n was a 1, and B is added if the previous value of Q_n was a 0 and no restoring of the partial remainder is required.

(12) a) Explain the Common bus system of Computers with a neat Sketch.

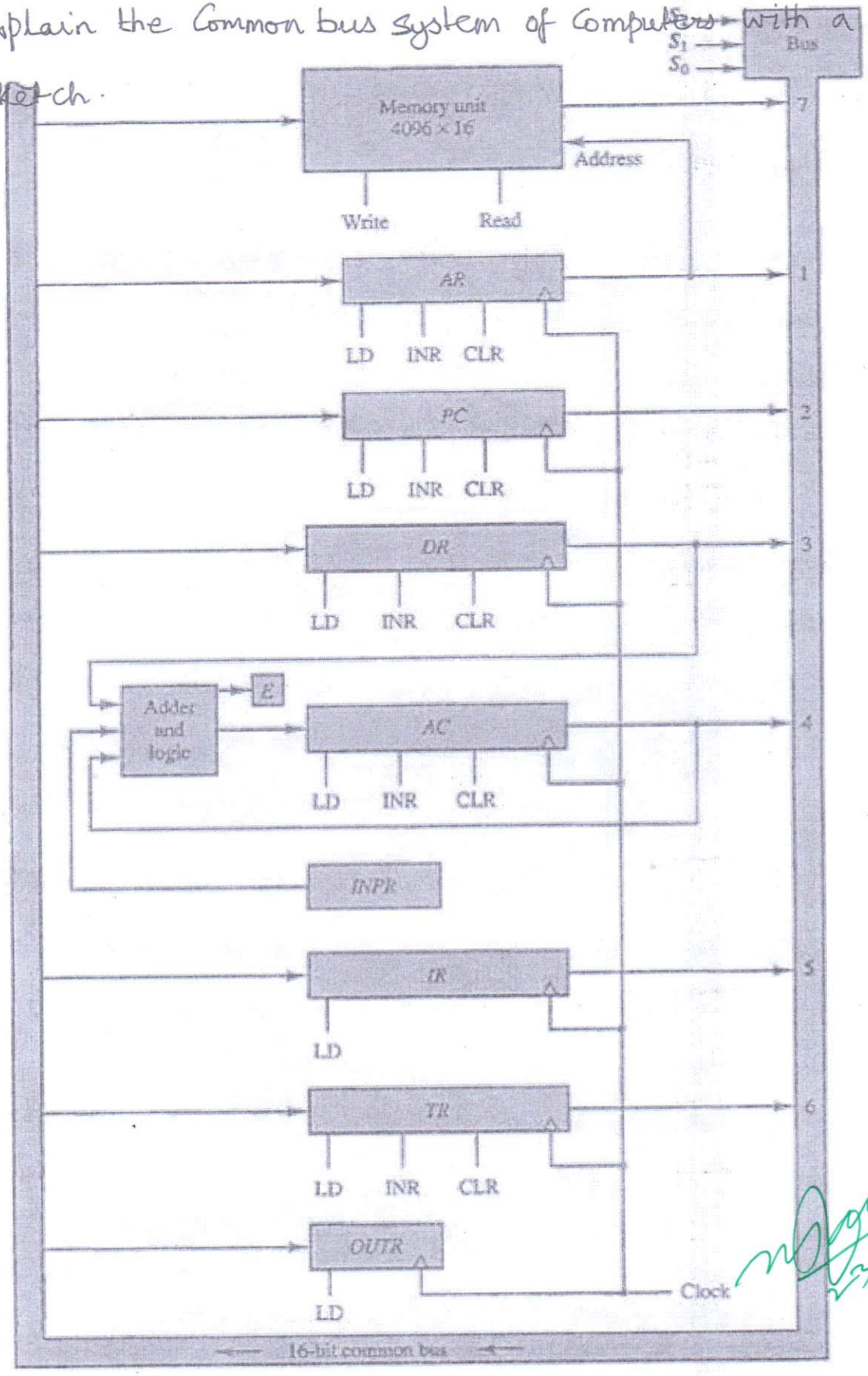


Figure 5-4 Basic computer registers connected to a common bus.

23/03/2022

(7)

Explanation of the above fig.

(12) b) Explain the various phases of an instruction cycle in details

A program residing in the memory unit of a Computer Consists of a Sequence of instrns. The program is executed in the Computer by going thro a Cycle for each instrn. Each instrn Cycle in turn is subdivided into a Sequence of subcycles or phases. In the basic Computer each instrn Cycle consist of the following phases.

1. Fetch an instruction from memory.
2. Decode the instruction.
3. Read the effective address from memory if the instrn has an indirect address.
4. Execute the instruction

Upon Completion of Step 4, the Control goes back to Step 1 to fetch, decode and execute the next instrn. This process Continues indefinitely unless a HALT instrn is encountered.

<u>Fetch & Decode</u> The μops for the fetch and decode phases Can be specified by the following register transfer Statements.

T_0 : $AR \leftarrow PC$

T_1 : $IR \leftarrow M[AR]$, $PC \leftarrow PC+1$

T_2 : $D_0, \ldots, D_7 \leftarrow$ Decode $IR(12-14)$, $AR \leftarrow IR(0-10)$, $I \leftarrow IR(15)$.

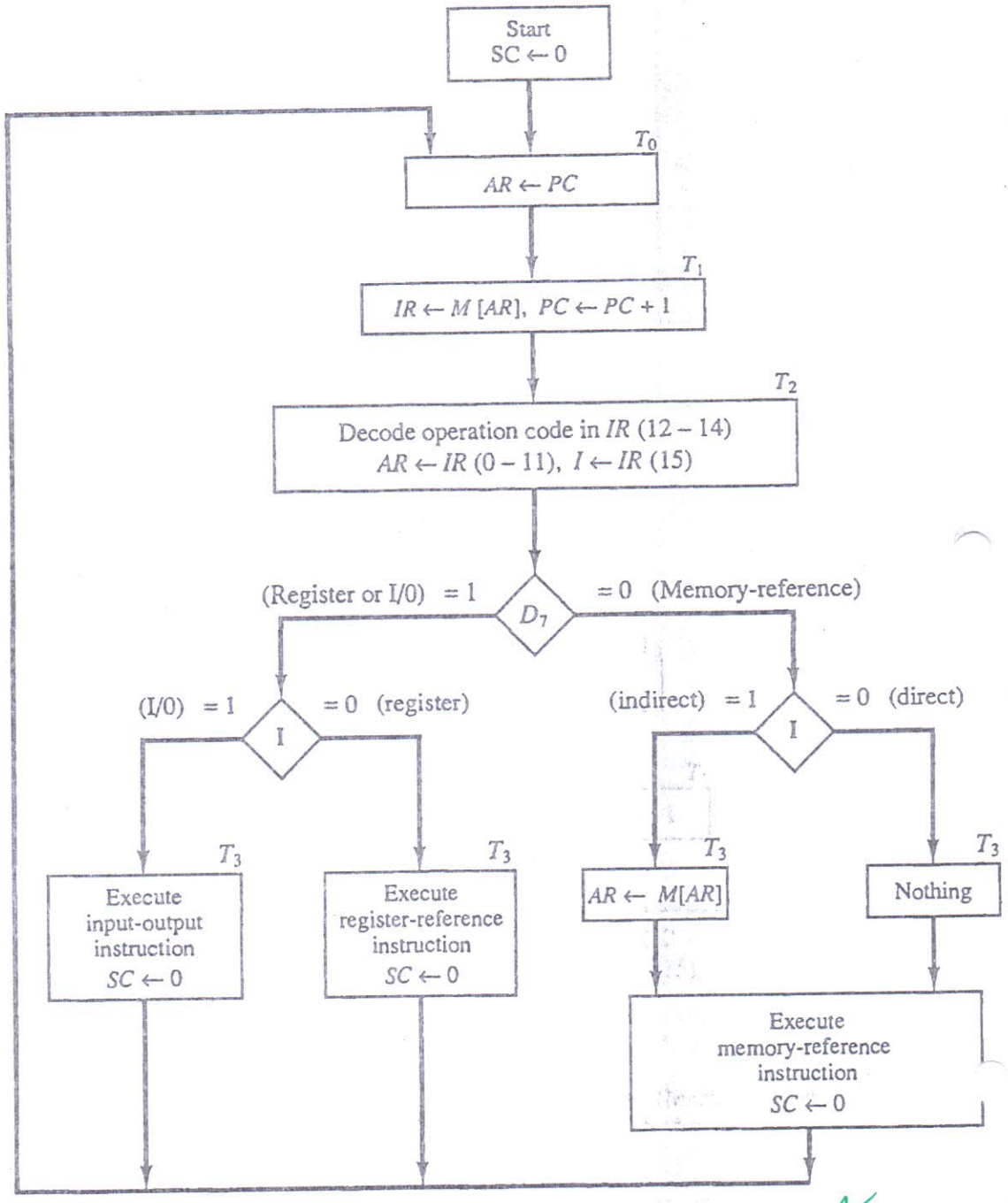


Figure 5-9 Flowchart for instruction cycle (initial configuration).

(8)

The timing signal that is active after decoding is T_3.

During T_3, the Control unit determines the type of instr that was read from memory.

register-reference or input–output type. If $D_7 = 0$, the operation code must be one of the other seven values 000 through 110, specifying a memory-reference instruction. Control then inspects the value of the first bit of the instruction, which is now available in flip-flop I. If $D_7 = 0$ and $I = 1$, we have a memory-reference instruction with an indirect address. It is then necessary to read the effective address from memory.

The three instrn types are subdivided into four separate pa The selected operation is activated with the clock transitio associated with timing signal T_3. This can be symbolized as

$D_7' I T_3$: $AR \leftarrow M[AR]$

$D_7' I' T_3$: Nothing

$D_7 I' T_3$: Execute a reg-ref instr

13 a) Explain possible pipeline hazards with its resolving technique.

Pipeline hazards are situations that prevent the next instruction in the instruction stream from executing during its designated clock cycles.

Any condition that causes a stall in the pipeline operations can be called a hazard.

There are primarily three types of hazards or conflicts that causes the instruction pipeline to deviate from its normal operation.

(i) Data hazards or Data dependency
(ii) Control hazards and
(iii) Structural hazards or Resource conflicts

A data dependency occurs when an instrn needs data that are not yet available.

eg. ADD R1, R2, R3
SUB R3, R1, R4

here, the 2nd instrn SUB for its execution depends on the arrival of the result provided by the 1st instrn ADD in dest. R1 reg. (RAW dependency). If the result in R1 is not available in time, the pipeline stalls, thereby degrading the performance.

The most straightforward way to overcome this conflict is to insert hardware interlocks.

Another technique called operand forwarding uses special hardware to detect a conflict and then avoid it by routing the data thro special paths between pipeline segments.

23/03/2022
PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No.3, HYDERABAD - 500 034. A.P.

Resource Conflicts or Structural hazards are caused by access to memory by two segments at the same time. This situation arises mainly when two instrns require a given hardware resource at the same time and hence for one of the instrns, the pipeline needs to be stalled.

Most of these conflicts can be resolved by using separate instrn and data memories.

Branch Conflicts or Control Hazards arise from branch and other instrns that change the value of PC.

The instrn fetch unit of the CPU is responsible for providing a stream of instrns to the execution unit.

The instrns fetched by the fetch unit are in consecutive memory locations and they are executed.

However, the problem arises when one of the instrns is a branching instrn to some other memory location.

Thus all the instrn fetched in the pipeline from consecu memory locations are invalid and are ∴ discarded.

This induces a stall till new instrns are again fetche from the memory address specified in the branch instr.

Thus the time lost as a result of this pipeline stal is called a branch penalty.

Pipelined computers employ various hardware techniques to minimize the performance degradation caused by instrn branching.

One way of handling a conditional branch is to prefetc the target instrn in addition to the instrn following the branch.

Contd:

(13) a) Both are saved until the branch is executed. If the branch condition is successful, the pipeline continues from the branch target instrn.

Another possibility is the use of a branch target buffer or BTB.

A variation of the BTB is the loop buffer.

Another procedure that some computers use is the branch prediction. A pipeline with branch prediction uses some additional logic to guess the outcome of a conditional branch instrn before it is executed.

The pipeline then begins prefetching the instrn stream from the predicted path. A correct prediction eliminates the wasted time caused by branch penalties

— 10 —

(13) b) What is the basic advantage of using interrupt-initiate data transfer over transfer under program control without an interrupt?

In an interrupt-driven data transfer technique, the CPU initiates the data transfer and proceeds to perform some other task.

When the I/O module is ready for data transfer, it informs the CPU by activating the interrupt request signal. The CPU suspends the task it was doing, services the request and returns back to the task it was doing

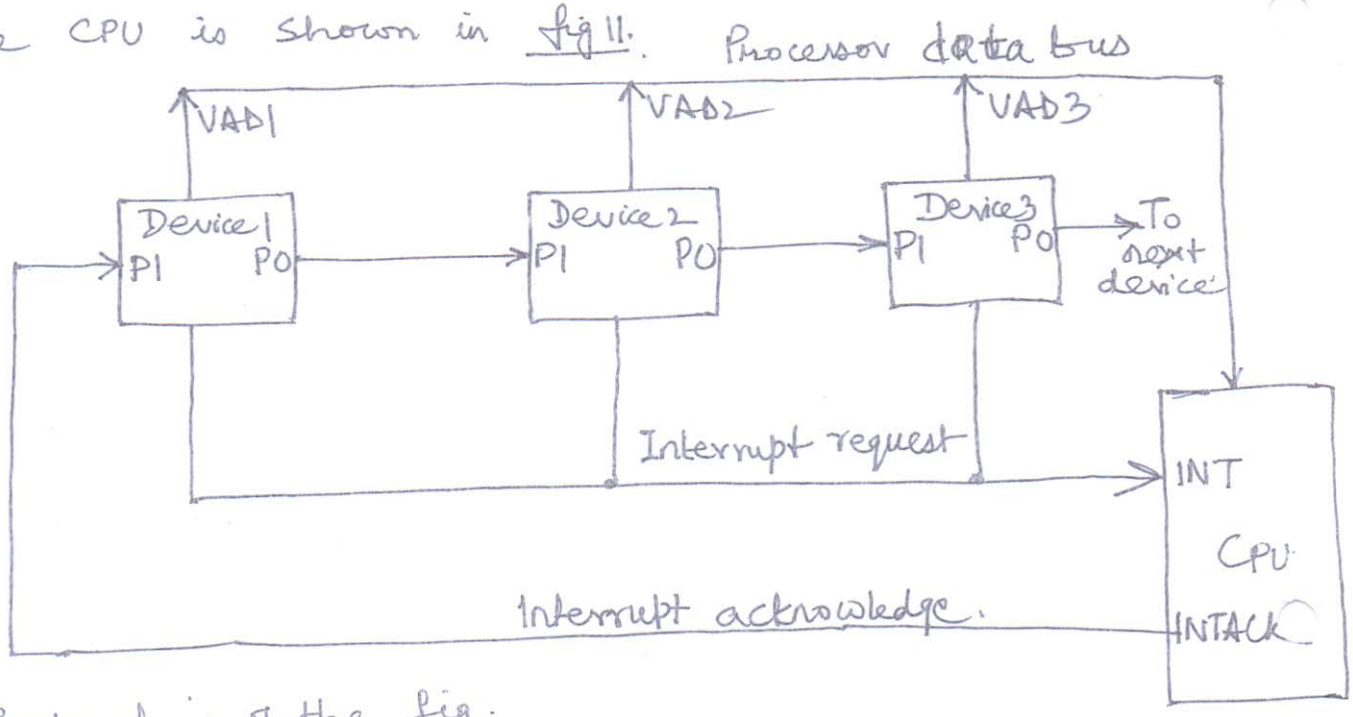
The basic advantage of interrupt initiated data transfer over program control transfer is the efficient utilization of CPU i.e, the CPU time is not wasted while checking the status of the I/O module. CPU time is reqd only during data transfer

— 5 —

(14) a) Explain the operation of Daisy chaining method of priority interrupt.

The daisy-chaining method of establishing priority consists of a serial connection of all devices that request an interrupt. The device with the highest priority is placed in the first position, followed by lower-priority devices up to the device with the lowest priority, which is placed last in the chain.

This method of connection between three devices and the CPU is shown in fig 11.



Explanation of the fig.

A device with a 0 in its PI input generates a 0 in its PO output to inform the next-lower priority device that the ack signal has been blocked.

A device that is requesting an interrupt and has a 1 in its PI input will intercept the ack signal by placing a 0 in its PO output.

If the device does not have pending interrupts, it transmits the ack signal to the next device by placing a 1 in its PO output.

Thus the device with PI = 1 and PO = 0 is the one with the highest priority that is requesting an interrupt, and this device places its VAD on the data bus.

— 10 —

Note: full marks for also showing blocking of one-stage of daisy chain priority arrangement

(14) b. What are the merits and demerits of Programmed I/O data transfer?

<u>Advantage</u> — · Simple to implement, very little hardware support.

A program and processor dedicated to wait and repeatedly tests the status and for I/O data transfer till the I/O operation completes.

<u>Disadvantage</u> — A program has to wait and repeatedly tests the status; waiting period for an asynchronous event can be too large.

· CPU is tied up in the loop with no useful work while checking the status of the I/O peripheral.

· Cannot be used for high speed data transfer.

— 5 —

(15) (a) Explain Set Associativity mapping in detail.

N-Way Set-Associative mapping is another approach — combination of the direct and associative mapping technique

Here the blocks of the cache are grouped into sets, and the mapping allows a block of the main memory to reside in any block of a specific set.

A group of N consecutive blocks in the cache is called a set.

Hence the contention problem of direct mapping is eased by having a few choices for block placement.

N-way Set-Associative Mapping

- Like direct mapping, a main memory block is mapped to a set.
- Set number = (Main memory block no.) % (No. of sets in cache)

The block can be placed anywhere within the set. (There are N choices).

Note: the value of N is a design parameter:

N = 1 : same as direct mapping

N = number of cache blocks : same as associative mapping

Typical values of N used in practice are: 2, 4, or 8.

(15) a) Contd

Blocks of the Cache are grouped into Sets, and the mapping allows a block of the main memory to reside in any block of a Specific Set.

A Cache that has K blocks per set is referred to as K-way Set associative Cache

An eg: of 2-way set associative Cache.

Assuming a main memory of 64KB and Cache of 2K consisting of 128 blocks of 16 words each.

The main memory address of 16 bits is divided into 3 fields: TAG, SET and WORD

∴ There are a total of 2048/16 = 128 blocks in the above 2-way Set associative Cache.

total no. of sets = 128/2 = 64 sets of 2 blocks each.

Each block contains 16 words

∴ 6-bits needed to identify set number.

Here, the tag field of the CPU address must be associatively compared to the two tags of the two blocks of the selected set to check if the desired block is present.

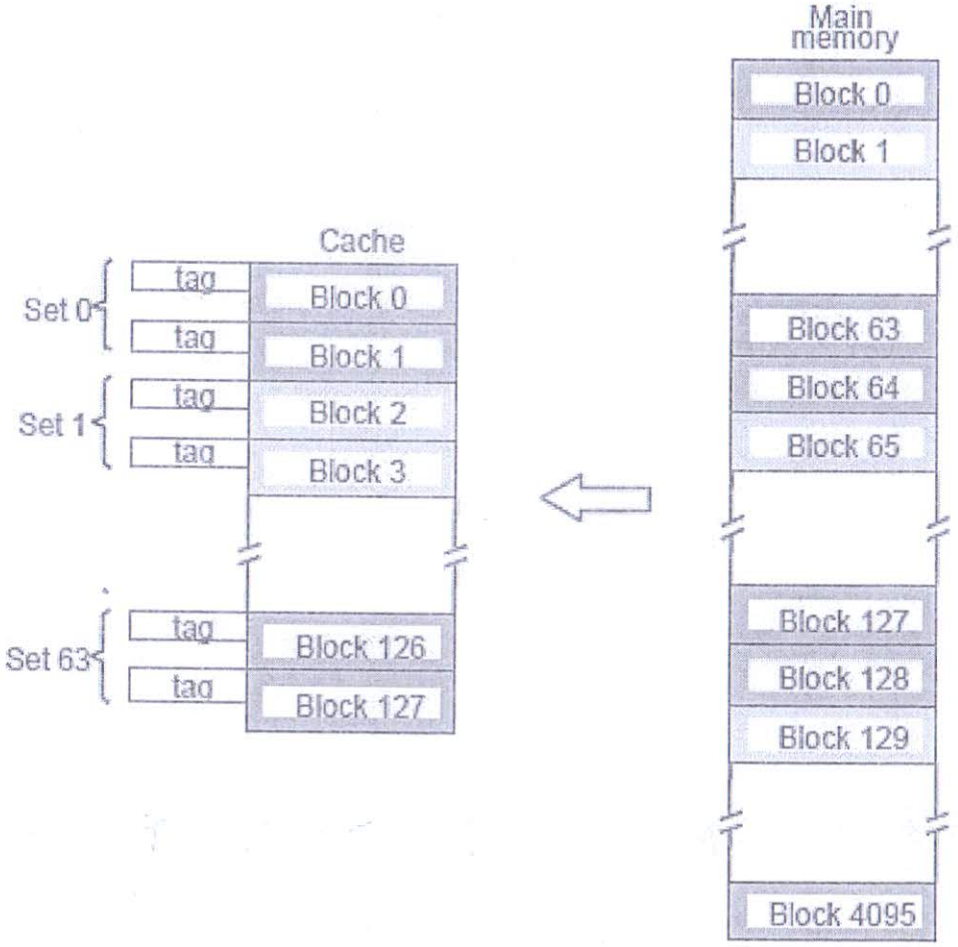
Thus, the hit ratio improves significantly over the direct mapping method.

27/03/2022

PRINCIPAL
Muffakham Jah College Of Engineering & Technology, Banjara Hills, Road No. 1, HYDERABAD - 500 034

Set-Associative Mapping (2)

- The 6-bit set field determines which set of the cache might contain the desired block.
- The tag field is associatively compared to the tags of the two blocks of the set to check if the desired block is present.



Tag	Set	Word
6	6	4

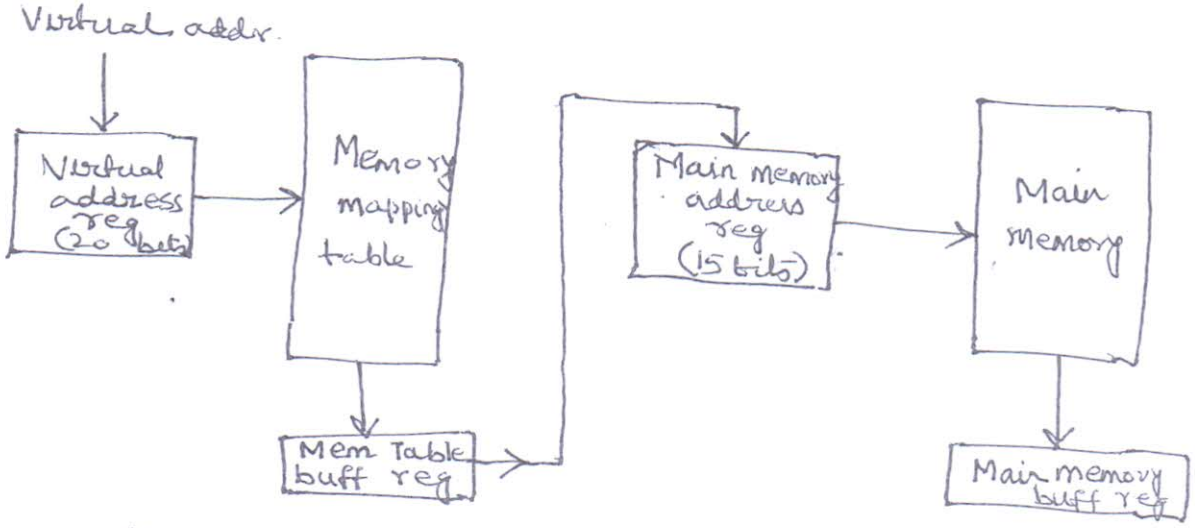
Main memory address

(15) b) Explain the Virtual memory in details.

Virtual memory is a Concept used in some large Computer Systems that permit the user to Construct programs as though a large memory space were available, equal to the totality of auxiliary memory.

Each address that is referenced by the CPU goes thro an address mapping from the Virtual address to a physical address in main memory.

Virtual memory is used to give a mechanism for translating program-generated addresses into Correct main memory Location. This is done dynamically, while programs are being executed in the CPU. The translation or mapping is handled automatically by the hardware by means of a mapping table.

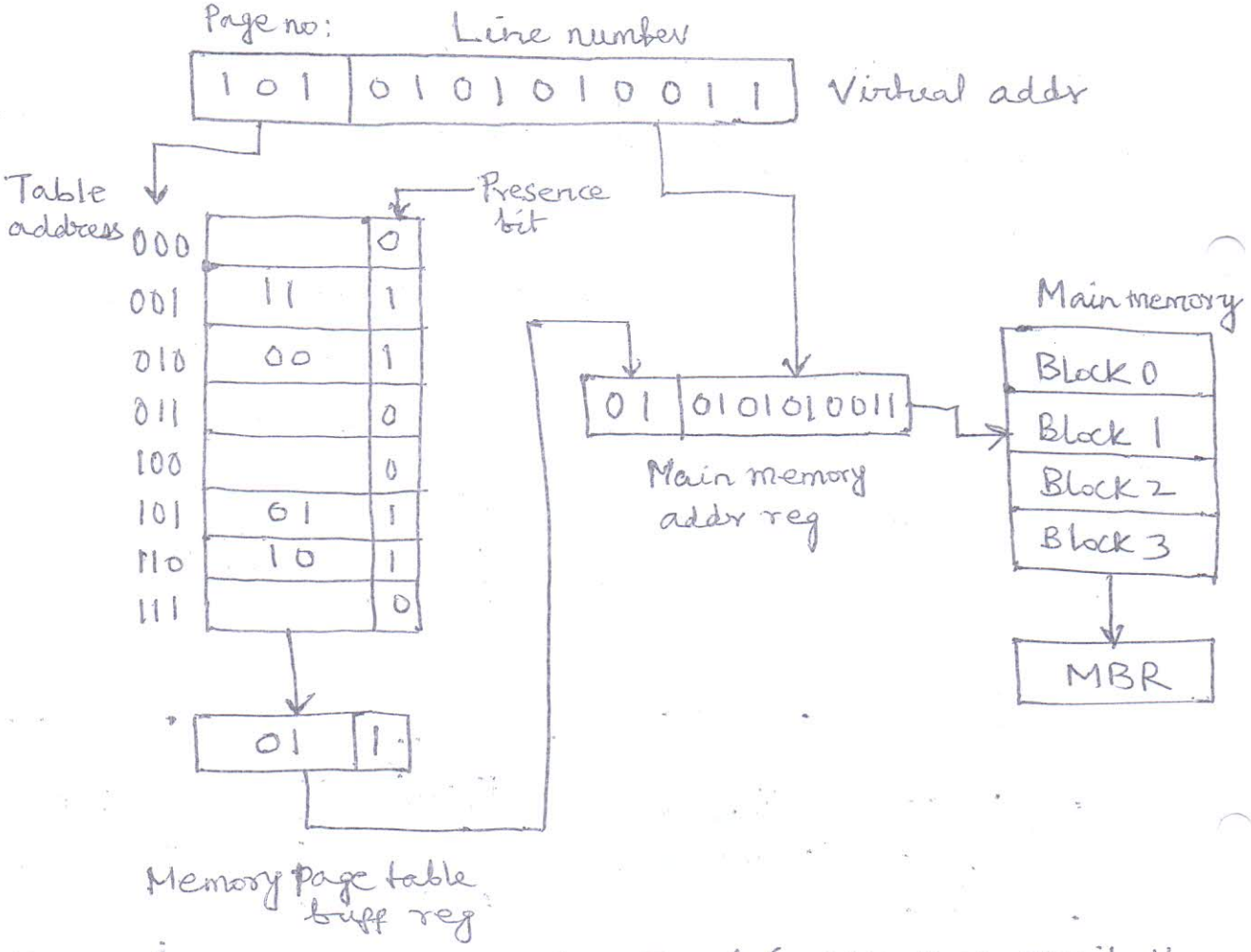


Memory table for mapping a Virtual address

The table implementation of address mapping is simplified if the inf in the address space and memory space are each divided into groups of fixed size. The physical memory is broken down into groups of equal size called blocks, whereas the address space is split up into groups of equal size called pages.

The programs are also considered to be split into pages. Portions of programs are moved from auxiliary memory to main memory in records equal to the size of a page.

✓ The org. of the memory mapping table in a paged system is shown in fig. below.



The table shows that pages 1, 2, 5 and 6 are now available in main memory in blocks 3, 0, 1 and 2 respectively.

The CPU references a word in memory with a virtual addr of 13 bits. The three high-order bits of the virtual addr specify a page number and also an address for the memory-page table. The content of the word in the memory table at the page no. addr is read out into the memory table buff reg. if the presence bit is a 1, the block no. thus read is transfered to the two high order bits of the main mem addr reg. The line no. from the virtual addr is transfered into 10 lower order bits of the memory addr reg.

(16) a) Describe in detail about IOP organization.

Instead of having each interface Communicate with the CPU, a Computer may incorporate one or more external processors and assign them the task of Communicating directly with all I/O devices.

An input-output processor (IOP) may be classified as a processor with DMA Capability that Communicates with I/O devices.

In this Config; the Computer System Can be divided into a memory unit, and a number of processors Comprised of the CPU and one or more IOPs.

Each IOP takes care of input and output tasks, releeving the CPU.

The IOP is similar to a CPU except that it is designed to handle details of I/O Processing. The IOP Can fetch and execute its own instrns. IOP instrns are Specifically designed to facilitate I/O transfers. In addition, the IOP can perform other processing tasks.

The block diag of a Computer with two processors is Shown in fig. below.

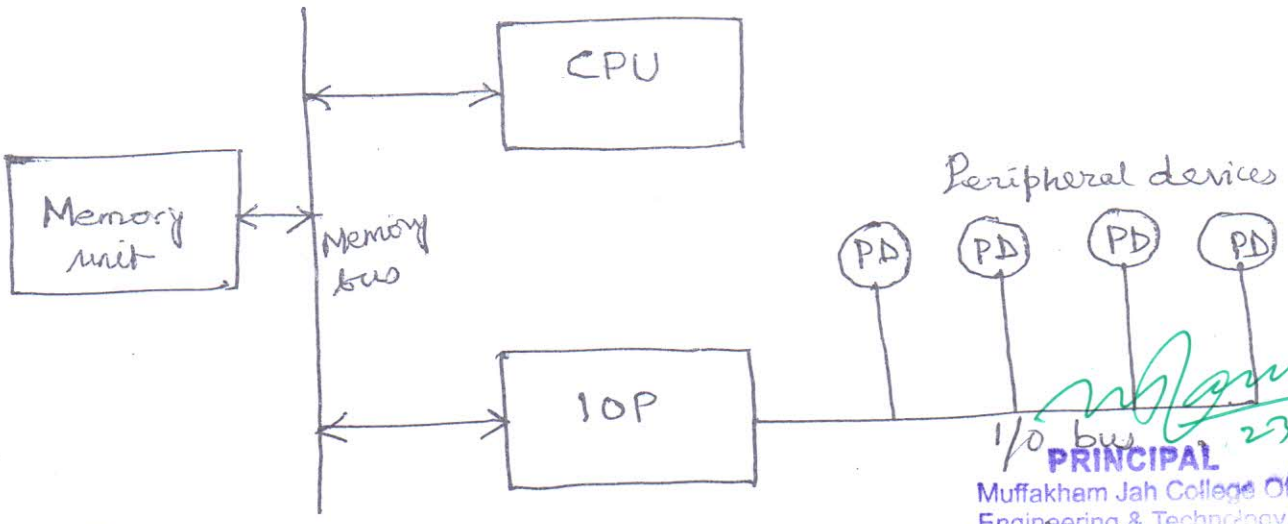


fig: Block diag of a Computer with I/O Processor

In the fig, the memory unit occupies a central position and can communicate with each processor by means of DMA.

The CPU is responsible for processing data needed in the solution of computational tasks.

The IOP provides a path for transfer of data between various peripheral devices and the memory unit.

The CPU is usually assigned the task of initiating the I/O program. From then on the IOP operates independently of the CPU and continues to transfer data from external devices and memory. The data formats of peripheral devices differ from memory and CPU data formats.

The IOP must structure data words from many different sources.

The communication between the IOP and the devices attached to it is similar to the program control method of transfer. Communication with the memory is similar to the DMA method.

The way by which the CPU and IOP communicate depends on the level of sophistication included in the system.

In most computer systems, the CPU is the master while the IOP is a slave processor.

The CPU is assigned the task of initiating all operations but I/O instrns are executed in the IOP. CPU instrns provide operations to start an I/O transfer and also to test I/O status conditions.

The IOP, in turn, typically asks for CPU attention by means of an interrupt. When an I/O operation is desired, the CPU informs the IOP where to find the I/O program and then leaves the transfer details to the IOP.

23/03/2020

(16) b) Explain the interrupt cycle with flow chart.

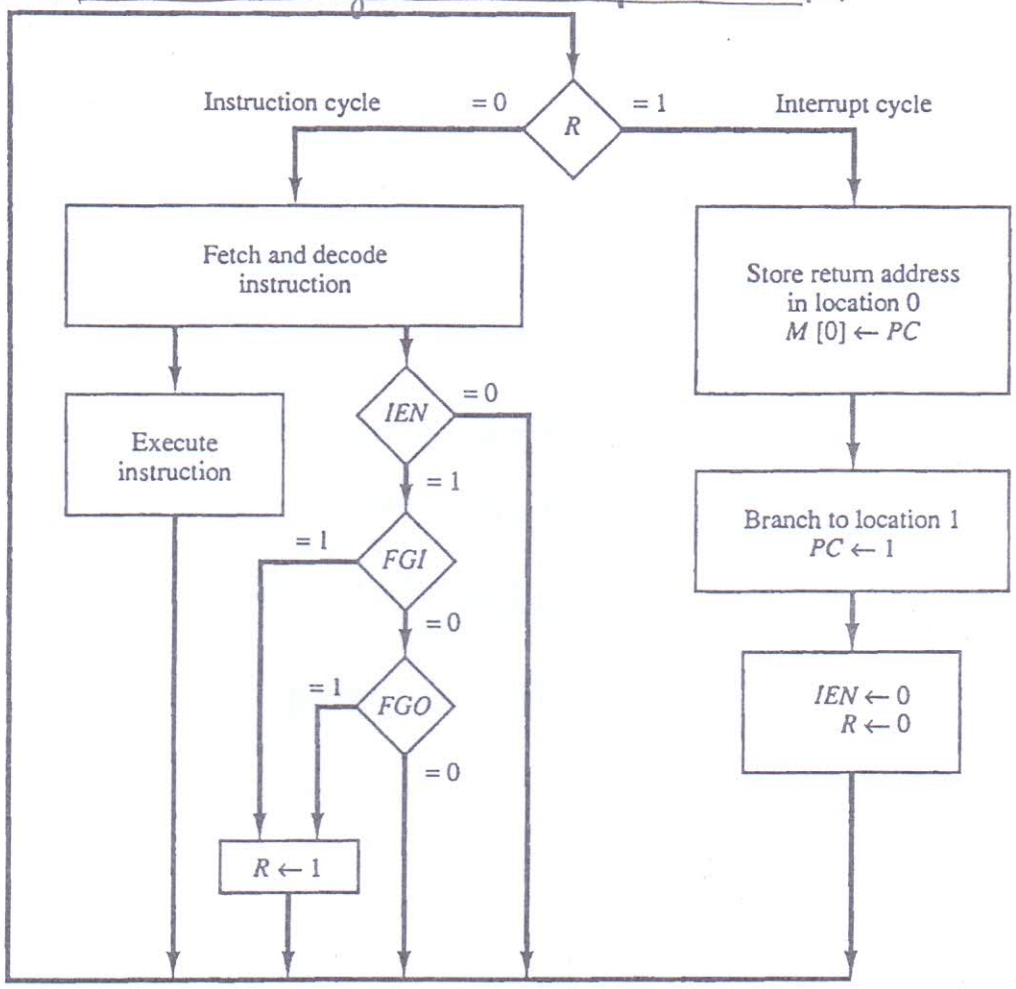


Figure 5-13 Flowchart for interrupt cycle.

The way that the interrupt is handled by the computer can be explained by means of the flowchart of Fig. 5-13. An interrupt flip-flop R is included in the computer. When $R = 0$, the computer goes through an instruction cycle. During the execute phase of the instruction cycle *IEN* is checked by the control. If it is 0, it indicates that the programmer does not want to use the interrupt, so control continues with the next instruction cycle. If *IEN* is 1, control checks the flag bits. If both flags are 0, it indicates that neither the input nor the output registers are ready for transfer of information. In this case, control continues with the next instruction cycle. If either flag is set to 1 while $IEN = 1$, flip-flop R is set to 1. At the end of the execute phase, control checks the value of R, and if it is equal to 1, it goes to an interrupt cycle instead of an instruction cycle.

interrupt cycle

The interrupt cycle is a hardware implementation of a branch and save return address operation. The return address available in *PC* is stored in a specific location where it can be found later when the program returns to the instruction at which it was interrupted. This location may be a processor reg, a memory stack. The memory location at address 0 is choosen for storing the return address. Control then enters inserts address 1 into PC and clears IEN and R so that no more interruptions can occur until the interrupt request from the flag has been serviced.

(17) a) Explain the design of Control unit. How to decode the micro operation fields? Explain the process.

The bits of the microinstruction are usually divided into fields, with each field defining a distinct, Separate function. The various fields encountered in instrn formats provide Control bits to initiate microoperations in the System, Special bits to Specify the way that the next address is to be evaluated, and an address field for branching.

The number of control bits that initiate microoperations Can be reduced by grouping mutually exclusive Variables into micro ops fields and encoding the k bits in each field to provide 2^k microoperations.

Each field requires a decoder to produce the Corresponding Control Signals. This method reduces the Size of the micro-instrn bits but requires additional hardware external to the Control memory.

Fig 7.7 Shows the three decoders and Some of the Connection that must be made from their outputs.

Each of the three fields of the microinstrn presently availab in the output of the Control memory are decoded with a 3x8 decoder to provide eight outputs.

Each of these outputs must be Connected to the proper Ckt to initiate the Corresponding microoperation.

The other outputs of the decoders that are associated with an AC operation must also be Connected to the arithmeti logic shift unit in a Similar fashion.

22/07/2020

(17) a) Contd

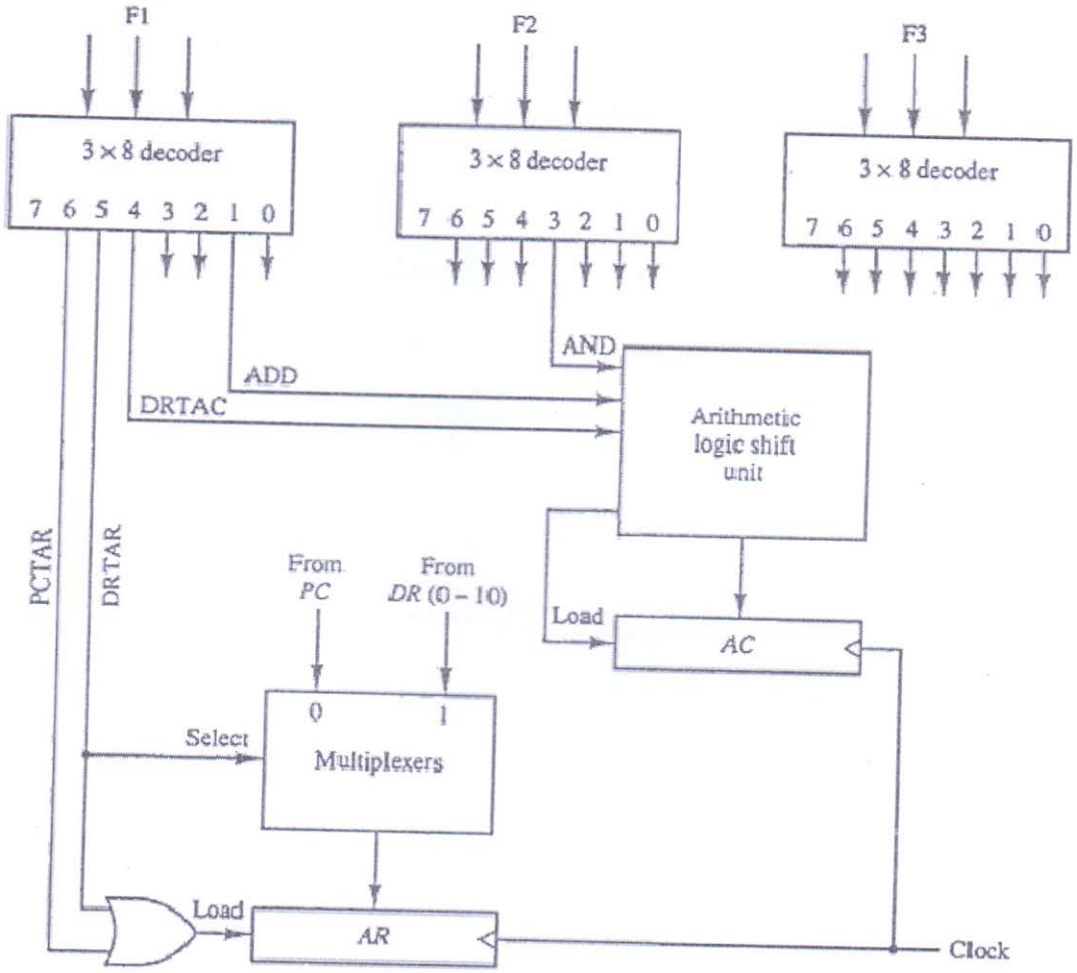


Figure 7-7 Decoding of microoperation fields.

(17) b) Explain Vector Processing in detail. — 8 —

There is a class of computational problems that are beyond the capabilities of a conventional computer. In many science and engineering applications, the problems can be formulated in terms of vectors and matrices that lend themselves to vector processing.

(17) b) Contd

Explain Vector processing in detail.

Computers with vector processing capabilities are in demand in specialized applications. Some of these applications where vector processing is of utmost importance are:

Long-range weather forecasting
Petroleum explorations
Seismic data analysis
Medical diagnosis
Aerodynamics & Space flight simulations
AI and expert systems
Mapping of human genome and
Image processing

A vector processor computer has the ability to process vectors, and related data structures such as matrices and multi-dimensional arrays, much faster than conventional computers. Vector processors may also be pipelined.

A vector V of length n is represented as row vector:

$$V = [V_1 \ V_2 \ V_3 \ \cdots \ V_n]$$

The element V_i of vector V is written as V(I) and the index I refers to a memory address or reg where the number is stored.

A computer capable of vector processing eliminates the overhead associated with the time it takes to fetch and execute the instrns in the program loop.

23/03/2020

(17) b) Contd

It allows operations to be specified with a single vector instruction of the form:

$$C(1:100) = A(1:100) + B(1:100)$$

A typical instrn format for vector processor is shown below

Op code	Base addr Source 1	Base addr Source 2	Base addr destination	Vector Length

This vector instrn includes the initial addresses of the two source operands, one destination operand, the length of the vectors and the operation to be performed.

In vector processing, identical processes are repeatedly invoked many times, each of which can be subdivided into subprocess

In vector processing, successive operands are fed thro the pipe-line segments and require as few buffers and local controls as possible. This parallel vector processing allows the generation of more than two results per clock period.

The parallel vector operations are automatically initiated either when successive vector instrns use different functional units and different vector regs, or when successive vector instrns use the result stream from one vector reg as the operand of another operation using different functional units. This process is known as chaining.

Vector processing is usually faster and more efficient than scalar processing because it reduces the overhead associated with the maintenance of the loop control variable.