

KONGU ENGINEERING COLLEGE, PERUNDURAI 638 060

CONTINUOUS ASSESSMENT TEST - 1

Regulations 2018

Month and Year : August 2019		Roll Number :
Programme	: B.E/B.Tech	Date : 19.07.19
Branch	: CSE/ IT	Time : 9.15 AM to 10.45 AM
Semester	: III	
Course Code	: 18CST32	Duration : 1 ½ Hours
Course Name	: Computer Organization	Max. Marks : 50

**PART - A (10 × 2 = 20 Marks)**

ANSWER ALL THE QUESTIONS

- Find the range of integers that can be represented by an n bit 2's complement number system. [CO1,K3]
- In 8 bit 2's complement representation, how the decimal number 43 is represented? [CO1,K3]
- Consider  $Z=X-Y$ , where X, Y and Z are all in sign-magnitude form. X and Y are each represented in n bits. To avoid overflow how many bits that the representation of Z would require? [CO1,K2]
- Identify the suitable addressing modes for the following [CO1,K2]
  - i)Array
  - ii)LOOPS
  - iii) Pointers
  - iv)Constants
- Show that the logic expression  $C_n \oplus C_{n-1}$  is a correct indicator of overflow in the addition of 2's complement integers by using an appropriate truth table. [CO2,K3]
- Let  $A=1\ 1\ 1\ 1\ 1\ 0\ 1\ 0$  and  $B=0\ 0\ 0\ 0\ 1\ 0\ 1\ 0$  be two 8 bit 2's complement numbers. Find the product in 2's complement using manual method. [CO2,K3]
- Show that the worst case delay through an  $n \times n$  array multiplier is  $6(n-1) - 1$  gate delays. [CO2,K3]
- Using Booth's algorithm, how many addition operations are to be reduced for the bit pattern 1 1 1 1 1 1 1 1 ? Similarly suppose a bit pattern consists of hundred 1's how many addition operations are reduced? [CO2,K3]
- How many 3-2 reduction levels are needed to reduce 16 summands to 2? [CO2,K3]
- Construct the sequential multiplication chart for the multiplicand 1 1 0 1 and the multiplier 1 0 1 1. [CO2,K3]

**PART – B (3 × 10 = 30 Marks)**  
ANSWER ANY THREE QUESTIONS

- i) Explain the basic functional units of a computer. (5) [CO1,K2]
- ii) Convert the following pairs of decimal numbers to 5 bit 2's complement numbers, then perform addition and subtraction. Indicate whether overflow occur (or) not. (5) [CO1,K3]
  - a) 7 and 9
  - b) -13 and 4
- Write RISC and CISC versions of the program for finding sum of N numbers (10) [CO1,K3]
- i) Illustrate 4 bit carry lookahead adder operation with appropriate logic expressions and circuits. (5) [CO2,K2]
- ii) Perform CSA for the multiplication of two numbers 45 and 63. (5) [CO2,K3]
- Apply both Booth multiplication and Bit-pair recoding for the multiplication of signed 2's complement numbers  $A=0\ 1\ 0\ 1\ 1\ 1$  and  $B=1\ 1\ 0\ 1\ 1\ 0$ . In both algorithms assume that A is the multiplicand and B is the multiplier. (10) [CO2,K3]

Bloom's Taxonomy Level	Remembering (K1)	Understanding (K2)	Applying (K3)	Analysing (K4)	Evaluating (K5)	Creating (K6)
Percentage	-	22%	78%	-	-	-

1).  $-2^{n-1}$  to  $+2^{n-1} - 1$

2).  $\begin{array}{r} 0010 \quad 1011 \\ \hline 8\text{-bit} \end{array} \quad (43)$        $\begin{array}{r} \checkmark & \checkmark & \checkmark \\ 32 \quad 16 \quad 8 \quad 4 \quad 2 \quad 1 \\ = 43 \end{array}$

$\begin{array}{r} 1 \quad 0 \quad 1 \\ \hline 6\text{-bit} \end{array}$

3).  $n+1$

4). i) Index Addressing mode

ii) Auto Increment / Auto Decrement A.M

iii) Indirect A.M

iv) Immediate A.M

5).  $O_{\text{Overflow}} = \bar{x}_{n-1} \bar{y}_{n-1} \bar{s}_{n-1} + \bar{x}_{n-1} \bar{y}_n \bar{s}_{n-1}$

XOR GATE is used.

0	0	0
0	1	1
1	0	1
1	1	0

$$6) \quad A = 1111 \ 1010 \ (-6) \quad 0000 \ 0101 \quad ;$$

$$B = 0000 \ 1010 \ (+10) \quad \underline{0000 \ 0110} \quad (+6)$$

(-7)  $1111 \ 1010$

~~$0000 \ 1010 \ X$~~

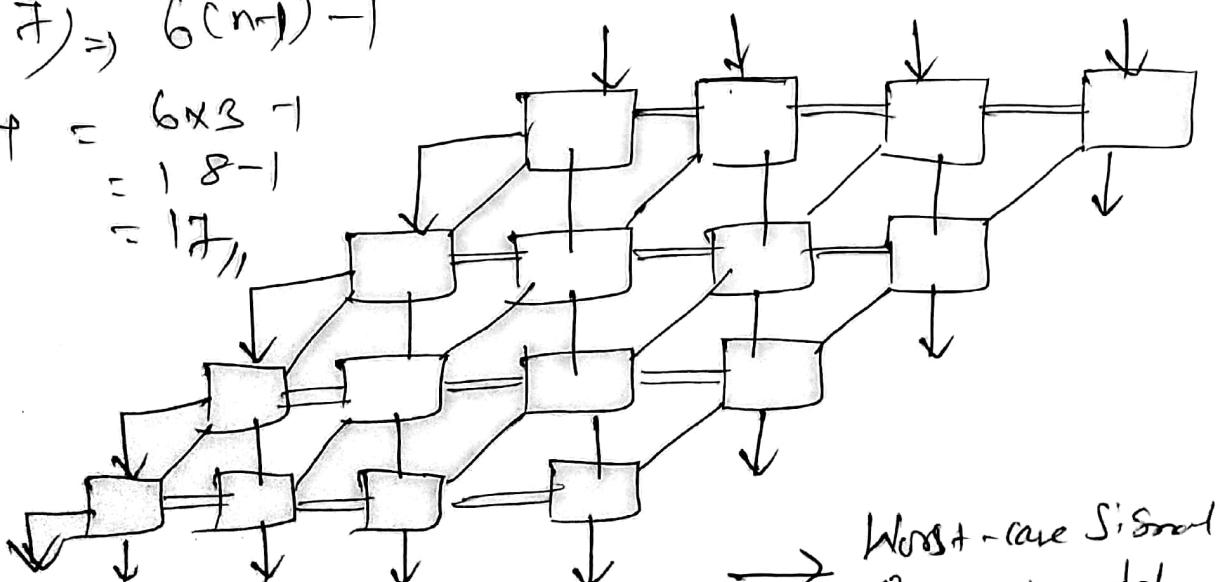
$$\begin{array}{r}
 11111111 \\
 00000000 00000000 \\
 11111111111010 \\
 0000000000000000 \\
 11111111111010 \\
 \downarrow \quad \leftarrow 0 \\
 \hline
 1111111111000100 \quad (-6)
 \end{array}$$

Cross check

$$\begin{array}{r}
 & & & 1 & 1 \\
 & & & | & | \\
 & & 00000000000111011 \\
 & & | \\
 & & 0000000000111100 \\
 & & 321684 \Rightarrow (+6)
 \end{array}$$

$$\therefore \Rightarrow 6(n-1)$$

$$\begin{aligned}
 n=4 &= 6 \times 3 - 1 \\
 &= 18 - 1 \\
 &= 17,
 \end{aligned}$$



Worst-case Signal propagation delay PATH.

8).



0 000000-

8 addition operations into 1 addition operation.

So Total no. of operations Reduced =  $n-1$

$$= 8-1$$

$$= 7$$

11) for 100 1's  $\Rightarrow n-1$

$$= 100-1$$

$$= 99$$

9) for 3-2 Reducers,  $k = 16$

$$L = 1.7 \log_2 k - 1.7$$

$$= 1.7 \log_2 16 - 1.7$$

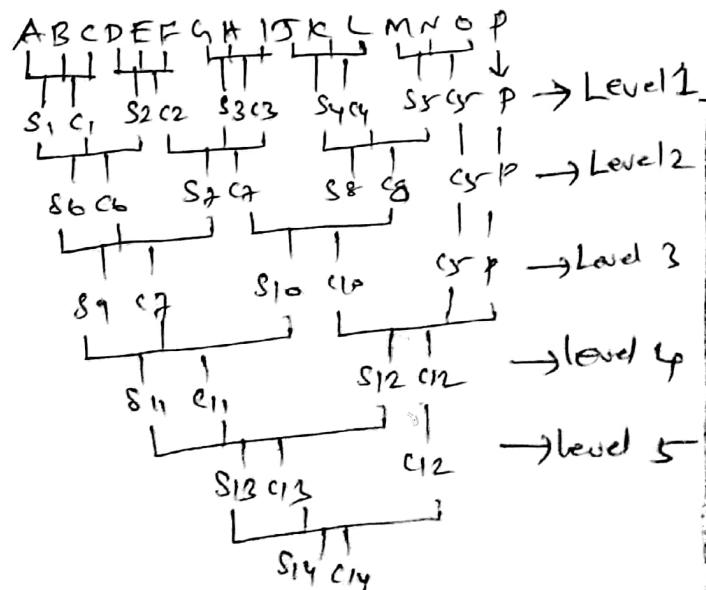
$$= 1.7 \times 4 \times \log_2 2 - 1.7$$

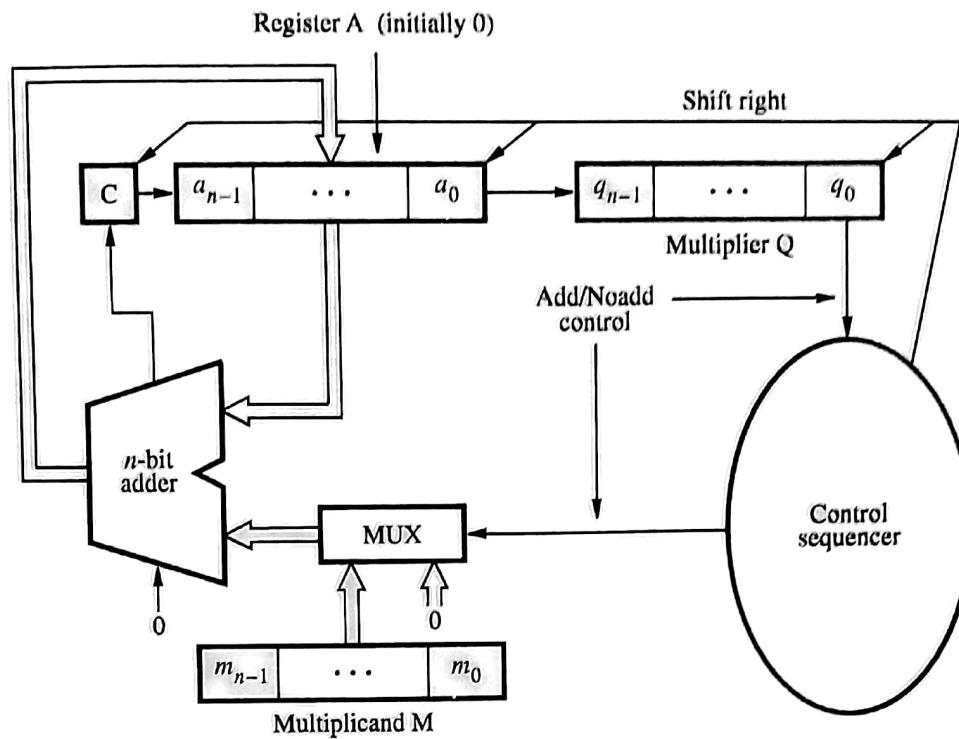
$$= 6.8 - 1.7$$

$$= 5.1$$

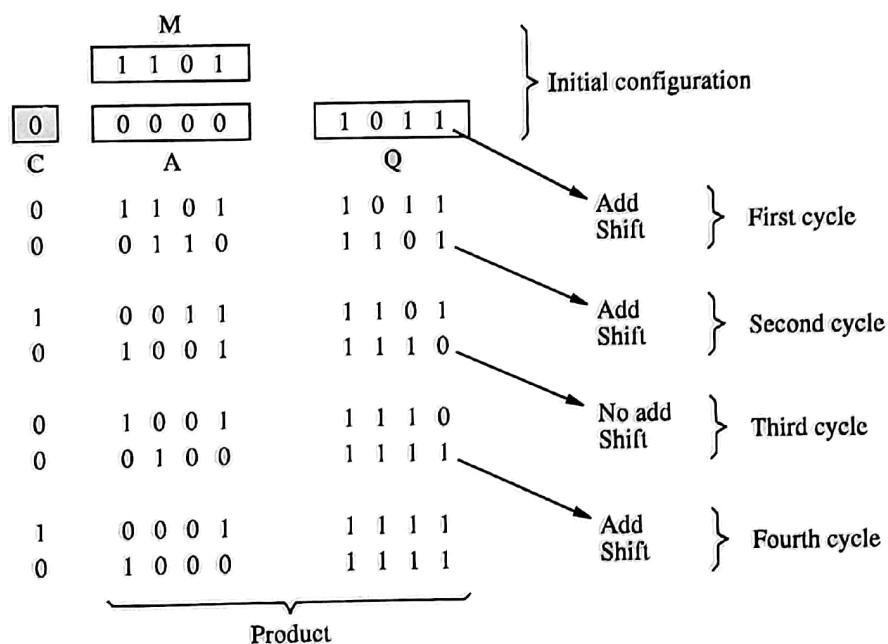
(QD).

$$\boxed{L = 5}$$





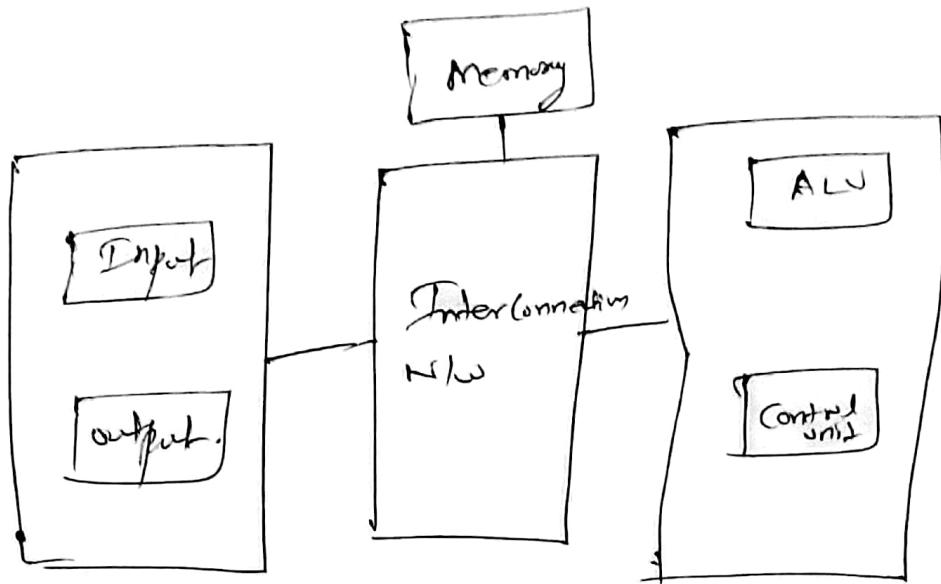
(a) Register configuration



(b) Multiplication example

**Figure 9.7** Sequential circuit binary multiplier.

17. q).



- 1) Input unit
- 2) memory unit      Primary memory  
Cache memory  
Secondary memory
- 3) ALU
- 4) output unit
- 5) Control unit

3) i). 7 and 9

$$+7 \Rightarrow$$

$$(+) +9 \Rightarrow$$

$$\underline{+16}$$

$$\begin{array}{r}
 & 0111 \\
 & 1001 \\
 \hline
 & 10000
 \end{array}
 \quad (+)$$

Overflow occurred

ii) -13 and +4

$$-13 \Rightarrow +13 \Rightarrow$$

$$\begin{array}{r}
 0110 \\
 1001 \\
 \hline
 \end{array}$$

$$-13 \Rightarrow$$

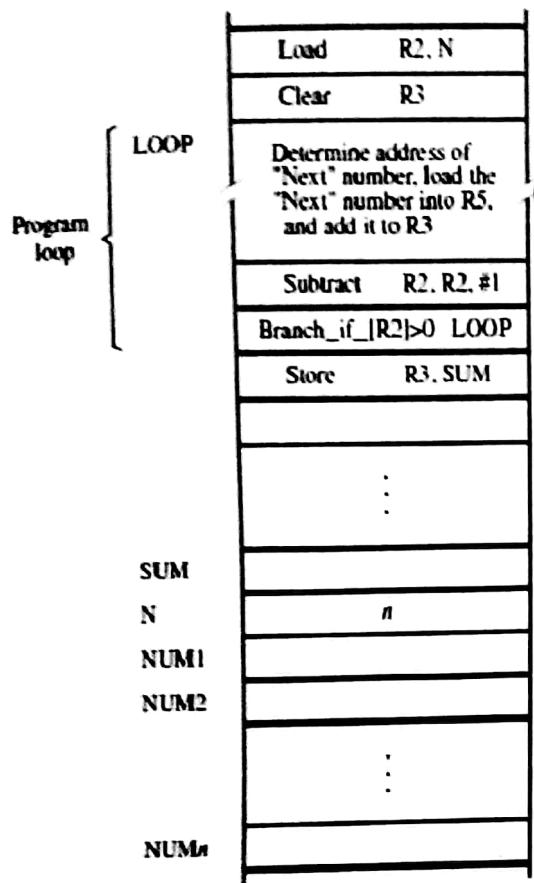
$$\begin{array}{r}
 1001 \\
 \hline
 00100
 \end{array}$$

$$+4 \Rightarrow$$

$$\begin{array}{r}
 00100 \\
 \hline
 10111
 \end{array}$$

No overflow,

$$\begin{array}{r}
 10111 \\
 \hline
 01000 \\
 \hline
 01001 = (+9)
 \end{array}$$



**Figure 2.6** Using a loop to add  $n$  numbers.

CISC (5 marks)

---

	Load	R2, N	Load the size of the list.
	Clear	R3	Initialize sum to 0.
	Move	R4, #NUM1	Get address of the first number.
LOOP:	Load	R5, (R4)	Get the next number.
	Add	R3, R3, R5	Add this number to sum.
	Add	R4, R4, #4	Increment the pointer to the list.
	Subtract	R2, R2, #1	Decrement the counter.
	Branch_if_[R2]>0	LOOP	Branch back if not finished.
	Store	R3, SUM	Store the final sum.

---

**Figure 2.8** Use of indirect addressing in the program of Figure 2.6.

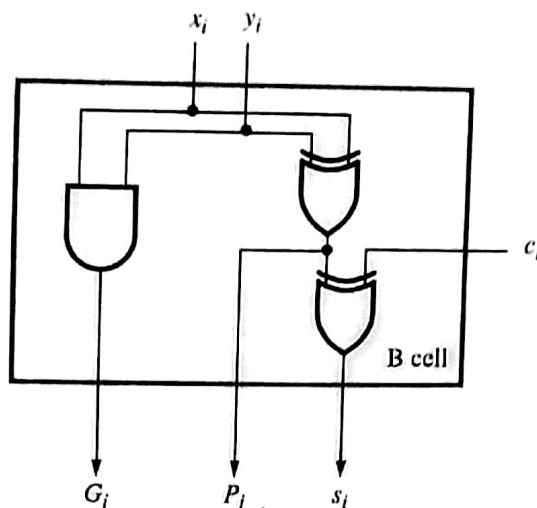
CISC (5 marks)

---

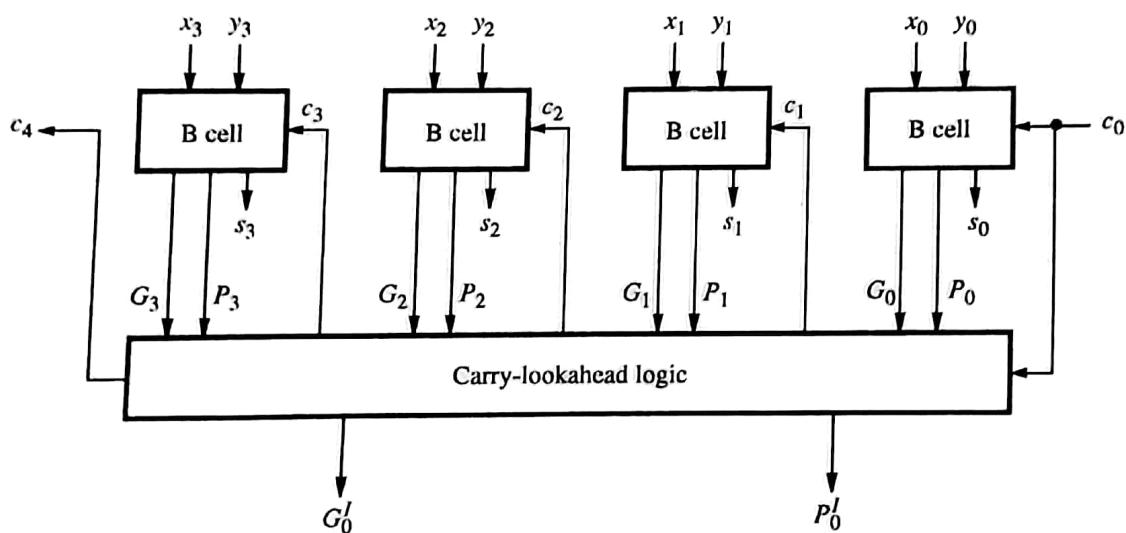
	Move	R2, N	Load the size of the list.
	Clear	R3	Initialize sum to 0.
	Move	R4, #NUM1	Load address of the first number.
LOOP:	Add	R3, (R4)+	Add the next number to sum.
	Subtract	R2, #1	Decrement the counter.
	Branch>0	LOOP	Loop back if not finished.
	Move	SUM, R3	Store the final sum.

---

**Figure 2.26** A CISC version of the program of Figure 2.8.



(a) Bit-stage cell



(b) 4-bit adder

**Figure 9.4** A 4-bit carry-lookahead adder.

Let us consider the design of a 4-bit adder. The carries can be implemented as

$$c_1 = G_0 + P_0 c_0$$

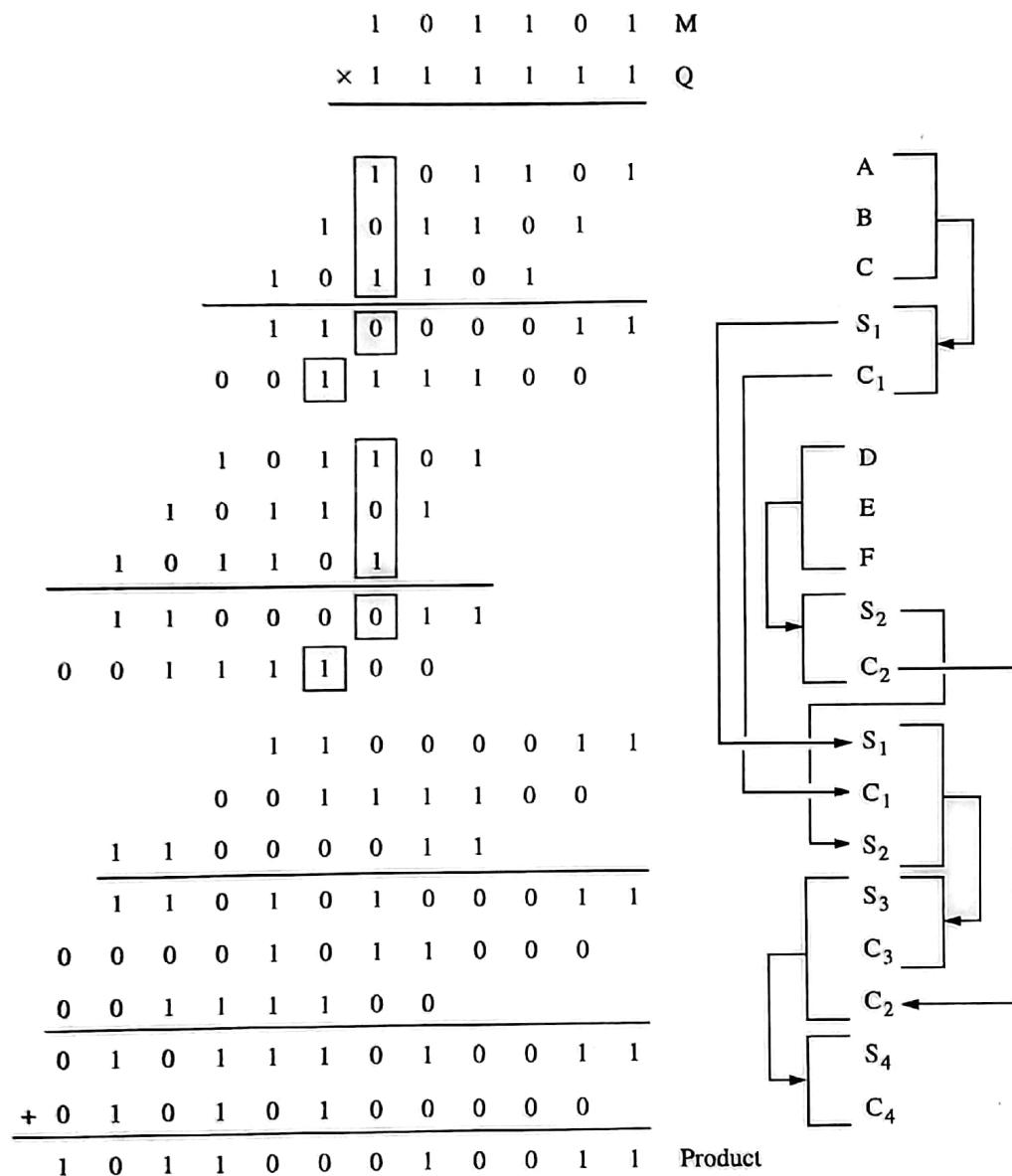
$$c_2 = G_1 + P_1 G_0 + P_1 P_0 c_0$$

$$c_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 c_0$$

$$c_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 c_0$$

(3) b)

bits of the multiplier are equal to 1. The six summands,  $A, B, \dots, F$  are added by carry-save addition in Figure 9.18. The blue boxes in these two figures indicate the same operand bits, and show how they are reduced to sum and carry bits in Figure 9.18 by carry-save addition. Three levels of carry-save addition are performed, as shown schematically in Figure 9.19. This figure shows that the final two vectors  $S_4$  and  $C_4$  are available in three adder delays



**Figure 9.18** The multiplication example from Figure 9.17 performed using carry-save addition.

$$A = 010111 \quad (+23)$$

$$B = 110110 \quad (-10)$$

$$\begin{array}{r} \checkmark & \checkmark & \checkmark \\ 16 & 8 & 4 & 2 & 1 \\ | & 0 & 1 & 1 & 1 \\ 110 & 110 \\ 001 & 001 \\ | & & & & \\ \underline{\underline{001010}} & & & & (+10) \end{array}$$

$\boxed{110} \boxed{110} \boxed{0}$

$0 - 1 + 1 0 - 1 0$

$$\begin{array}{r} 001001 \\ | \\ \underline{\underline{001010}} \end{array}$$

$(+10)$

$$010111$$

$$0 - 1 + 1 0 - 1 0$$

$$-1 \times \cancel{+23} = -23$$

$$\cancel{010111}$$

$$\cancel{215} \rightarrow \boxed{101001}$$

$$1 \quad \begin{array}{r} 11101 \\ \hline 00000000000000 \end{array}$$

$$11111101001$$

$$0000000000000$$

$$000010111$$

$$10101001$$

$$00000000$$

$$D) \quad \begin{array}{r} 111100011010 \\ \hline 000011100101 \end{array} \quad (-230)$$

$$\begin{array}{r} 000011100101 \\ \hline 1 \end{array}$$

$$000011100101$$

$$128 \quad 64 \quad 32 \quad 4 \quad 2 \Rightarrow (+230)$$

$$\boxed{110} \boxed{1100}$$

$$\begin{array}{r} \checkmark \quad \checkmark \quad \checkmark \\ 64 32 16 8 4 2 1 \\ 0101110 \quad (+46) \end{array} \quad \begin{array}{r} 010111 \\ -1 + 2 - 2 \\ \hline 01010010 \end{array} \quad \begin{array}{r} -2 \times \cancel{+23} = -46 \\ 0101110 \\ 1010001 \\ \hline 1010010 \end{array}$$

$$11111010010$$

$$0000101110$$

$$11101001$$

$$D) \quad \begin{array}{r} 111100011010 \\ \hline 111100011010 \end{array} \quad (-230)$$

DEPT. OF COMPUTER SCIENCE & ENGG.  
KONGU ENGINEERING COLLEGE,  
THOOPUPALAYAM (PO),  
PERUNDURAI (TK), ERODE - 638 060



**KONGU ENGINEERING COLLEGE**  
PERUNDURAI ERODE - 638 060.  
(Autonomous)



SV  
Name and signature of Hall Supdt. with Date

Name of the Student	V-RUBASHREE	Register No.	I 8 C S R I 6 3
Programme	B.E	Branch & Semester	CSE - III
Course Code and Name	18CST32 COMPUTER ORGANIZATION	Date	19-08-19 No. of Pages Used 15

**MARKS TO BE FILLED IN BY THE EXAMINER**

PART - A		PART - B		Grand Total Max. Marks : 50
Question No.	Max Marks : 2	Question No.	Max Marks : 10	
1	✓ 2	11	i) 4 ✓	
2	✓ 2	ii) 2 ✓		
3	✓ 2	12	i) 2 ✓	
4	✓ 2	ii) 2 ✓		
5	✓ 2	13	i) 3 ✓	
6	✓ 2	ii) 2 ✓		
7	✓ 1	14	i) 1 ✓	
8	✓ 2	ii) 2 ✓		
9	✓ 0			
10	✓ 0			
TOTAL	17	TOTAL	26 ✓	

Total marks in words: *Twenty Six Marks (26)*

**INSTRUCTION TO THE CANDIDATE**

- Check the Question Paper, Programme, Course Code, Branch Name etc., before answering the questions.
- Use both sides of the paper for answering questions.
- POSSESSION OF ANY INCRIMINATING MATERIAL AND MALPRACTICE OF ANY NATURE IS PUNISHABLE AS PER RULES.

*S. H. Allyn*  
Name of the Examiner

*21/8/2019*  
Signature of the Examiner with Date

Part-A.

1. The range of integers that can be represented by  $n$  bit 2's complement number system is  $-2^{n-1}$  to  $2^{n-1} - 1$ .

2)  $43 = (0101011)_2$

$$\begin{array}{r} 00101011 \\ 11010100 \\ \hline 11010101 \end{array}$$

: 2's complement of 43 is  $(11010101)_2$

- 3) \* To avoid overflow no. of bits that the representation of  $x$  would require is  $(n+1)$  bits.

4) i) Array = Index Addressing mode

ii) LOOPS = Offset with Addressing mode (or) Register Addressing mode

iii) Pointers = Indirect Addressing mode

iv) Constants = Immediate Addressing mode.

$$5. Y = C_n \oplus C_{n-1}$$

$C_n$	$C_{n-1}$	$Y$
0	0	0
0	1	1
1	0	1
1	1	0

When addition of two integer, where the sign of two summands same, but sign of result is different, it is said to overflow occurred.

$$6. A = 1111 \ 1010 ; B = 0000 \ 1010$$

$$\begin{array}{r}
 01010 \times 1010 \\
 \hline
 00000 \\
 01010 * \\
 00000 ** \\
 01010 *** \\
 \hline
 01100100
 \end{array}$$

$$\therefore A \times B = 1110 \ 0100.$$

$$\begin{array}{r}
 1010 \\
 6000 \\
 1010 \\
 0000 \\
 \hline
 01100100
 \end{array}$$

### 7. Gate Delays:

In Full adder for  $C_{n-1} = 2(n-1)-1$   
delays occurs  
In Add / Subtractor control unit  
 $C_{n-1} = 2(n-1)$  gate delays occurs.

Here  $n=1$

$$\text{Full adder, } C_{n-1} = 2(2-1)-1 \\ = 2(1)-1 \\ = 2-1$$

$$\boxed{C_{n-1} = 1 \text{ GID}}$$

$$\text{Add / Subtractor, } C_{n-1} = 2(2-1) \\ = 2(1)$$

$$\boxed{C_{n-1} = 2 \text{ GID}}$$

### 8. Booth algorithm:

In bit pattern,

11111111

No. of addition operation reduced  
is one.

Likewise in 100 '1' bit pattern, no. of addition operation reduced is one.

9. 3-2 reduction levels:

\* The reduction levels needed for

to reduce 16 summands

to 2 is 14 levels.

In general, reduction level is  $(n-2)$  levels.

where  $n$  is no. of summands.

$$M = 1101$$

$$Q = 1011$$

$$\begin{array}{r}
 C \quad 1101 \quad M \quad 1011 \\
 \hline
 0 \quad 0000 \quad A \\
 \hline
 0 \quad 10101 \quad 1011 \\
 \hline
 0 \quad 0110 \quad 1101 \\
 \hline
 1 \quad 0011 \quad 1101 \\
 \hline
 0 \quad 1001 \quad 1110 \\
 \hline
 0 \quad 0100 \quad 1111 \\
 \hline
 1 \quad 0001 \quad 1111 \\
 \hline
 0 \quad 1000 \quad 1111
 \end{array}$$

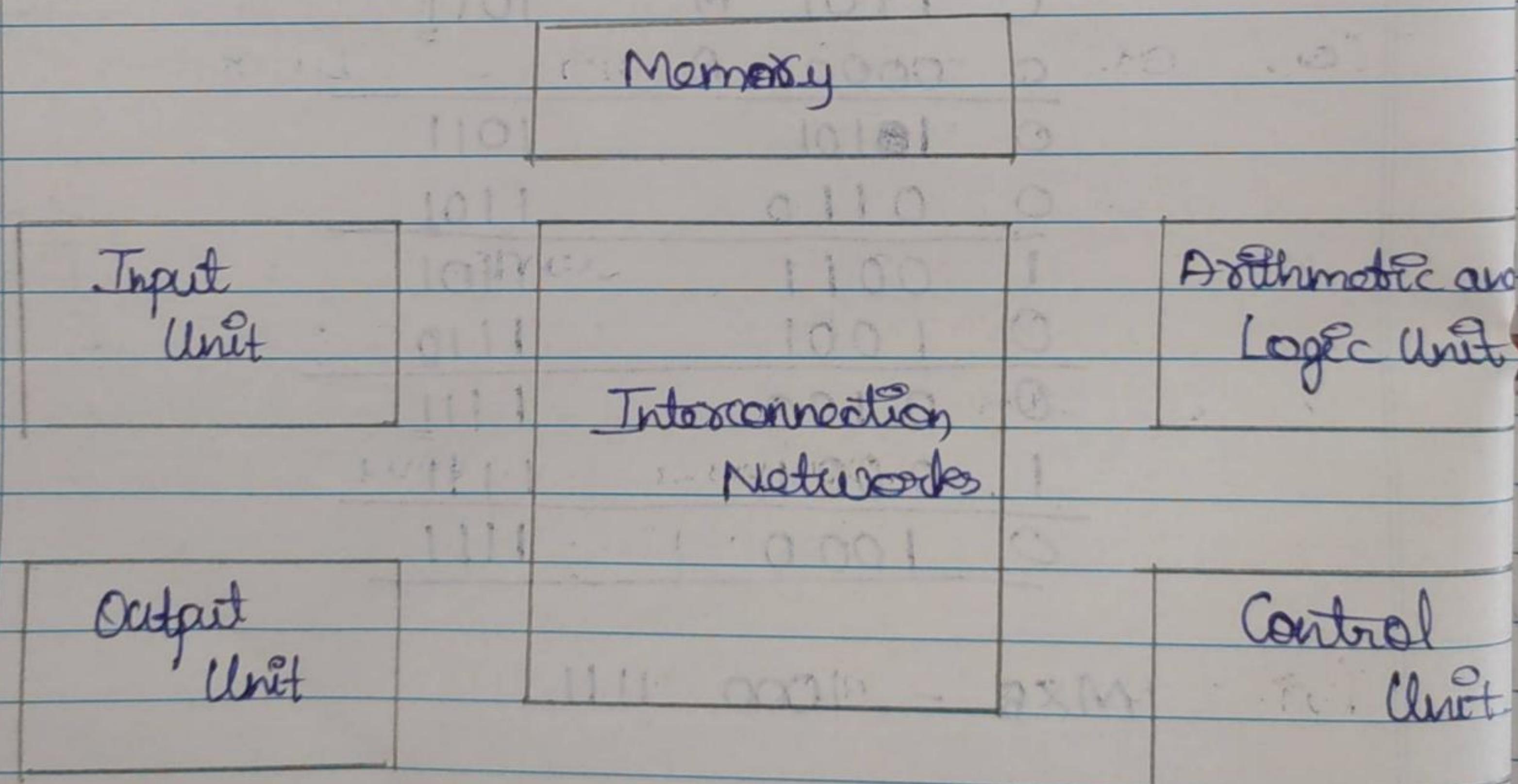
$$M \times Q = 1000 \quad 1111$$

ii)

## Part-B) ~~Computer System~~ Basic Functional Units of a Computer:

The Basic Functional Units of Computer are,

- \* Input Unit
- \* Memory Unit
- \* Arithmetic and Logic Unit
- \* Output Unit
- \* Control Unit.



Functional Unit.

## Input Unit:

\* Input Unit has functions which has to receive input in the form of data or instructions and give it to the memory.

\* Usually, we give inputs to the system through keyboards, mouse etc. Other Input units are joystick, Barcode reader, USB - Pendrive etc.

## Memory Unit:

\* Memory Unit plays vital role in computer which store information and data into it.

\* The input we give is stored in this memory unit.

\* There are two types of memory they are,

\* primary memory

\* secondary memory.

## Primary memory:

\* Primary memory is also called as temporary memory, since it is a volatile memory.

RAM and ROM are primary memory. This primary memory is also called as main memory. When we turn off our computer this gets volatile and when we switch on, it is empty. RAM memory very small in size.

### Cache memory:

\* Cache memory is the fastest memory than RAM. It is built in within processor. It makes the execution of program quicker.

\* Cache memory is also volatile memory.

### Secondary memory:

\* Secondary memory is very large and non permanent memory.

\* In this memory the programs and instructions which needs are stored here.

## Arithmetic and Logic Unit:

- \* This unit is present within the processor, where arithmetic operations are carried out.
- \* In this Unit, Registers are used as operands and Registers are also speediest execution of memory.

## Output Unit:

- \* Output Unit conveys the information to the outside world through monitor, printers etc.
- \* Speed of printer that can print 20 or more pages per minutes.
- \* These instructions are fetched from memory and displayed.

## Control Unit:

- \* Control Unit is unit which execute and coordinate all the unit for execution of instructions or data.

\* This control unit sends the signal through interconnection network which decides and sends signal to corresponding unit.

\* Functional units are coordinated by Control Unit.

ii) ii)

a) 7 and 9

$$\begin{array}{r} 7+9: \\ \text{Sum} = 10110 \\ \text{Minuend} = 01111 \\ \text{Subtrahend} = 01001 \\ \hline \text{Result} = 10000 \end{array}$$

Here, overflow occurred because the sign of summands are same, but sign of result is different.

$$\begin{array}{r} 7-9: \\ 7 = 01111 \\ -9 = 10111 \\ \hline \text{Result} = 10111 \end{array}$$

\* Overflow does not occur.

b)  $-13$  and  $4$

$$-13 + 4 = -9; \quad -13 = 10011$$

$$4 = \begin{array}{r} 00100 \\ 10111 \end{array}$$

\* Overflow does not occur.

$$-13 - 4 = -17$$

$$-13 = 10011$$

$$-4 = \begin{array}{r} 11100 \\ 10111 \end{array}$$

\* Overflow occurs.

12) Finding Sum of N Numbers.

RISC:

Load R<sub>0</sub>, N

Clear R<sub>1</sub>

Load R<sub>1</sub>, N

13) i) 4 bit carry lookahead adder operation.

Logic expression:

$$P = D = 4$$

$$C_{i+1}^0 = G_i + P_i C_i$$

$$C_{4+1}^0 = G_4 + P_4 C_4$$

$$C_5 = G_4 + P_4 (G_3 + P_3 C_3)$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 C_3$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 (G_2 + P_2 C_2)$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 (C_2)$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 (G_1 + P_1 C_1)$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2$$
  
$$(G_0 + P_0 C_0)$$

$$C_5 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2 P_1 G_0$$
  
$$+ P_4 P_3 P_2 P_1 P_0$$

29) Carry Save Addition for multiplication:

45 and 63

$$\begin{array}{r} 32 \\ 16 \\ 8 \\ 4 \\ \hline 56 \end{array}$$

$$\begin{array}{r} 111 \\ 111 \\ \hline 111 \end{array}$$

ate

$$\begin{array}{r} 64 32 16 8 4 2 \\ 0 1 1 1 1 1 \\ \hline 0 1 0 1 1 0 1 \end{array}$$

$63 \times 45$ .

$$\begin{array}{r} 01111111 \\ \times 000101101 \\ \hline \end{array}$$

$$\begin{array}{r} 01111111 \quad A \\ 100000000* \quad B \\ \hline 01111111** \quad C \\ \hline 000000011 \quad S_1 \\ \hline 001111100 \quad C_1 \\ 01111111*** \quad D \\ 00000000**** \quad E \\ \hline 01111111**** \quad F \\ 011000011000 \quad S_2 \\ 000111100000 \quad C_2 \\ \hline 00000000**** \quad G \\ 00000000***** \quad H \\ 00000000***** \quad I \\ \hline 0000000000000000 \quad S_3 \\ 0000000000000000 \quad C_3 \\ \hline 0100000011 \quad S_1 \\ 0011111100 \quad C_1 \\ \hline 011000011000 \quad S_2 \\ \hline 011001100011 \quad S_4 \\ 000100110000 \quad C_4 \\ \hline 0000000000000000 \quad S_3 \\ 0000000000000000 \quad C_3 \\ \hline 000111100000 \quad C_4 \\ 000111100000 \quad S_5 \\ 0000000000000000 \quad C_5 \end{array}$$

000000101101010011 Ans.

$$63 \times 45 = 0010110101001100$$

Booth multiplication:

$$A = 010111$$

$$B = 110110$$

Booth Code:

$$\begin{array}{r} 110110 \\ \times 011010 \\ \hline 110110 \end{array}$$

$$\begin{array}{r} 010111 \\ \times 101000 \\ \hline 101001 \end{array}$$

$$\begin{array}{r} 010111 \times 0-110-10 \\ \hline \end{array}$$

$$\begin{array}{r} 1110111 \\ \times 000000000 \\ \hline 1111101001 \\ 000000000 \\ 00010111 \\ 1101001 \\ 0000000 \\ \hline \cancel{X} \cancel{11100011010} \end{array}$$

$$A \times B = 11100011010$$

Bit pairs recording:

$$A = 010111$$

$$B = 110110$$

Bit pair code:

11101100

0101110

0000000

-1 +2 -2

010111

0101110

011011

1010000

1010010

010111 x -1 +2 -2

1010010

0101110

110111

01101100

101000

101001

00101110

101001\*

101001\*\*

X1100011010

000000000

$A \times B = 1100011010$

1001011

0000000

010110001111

010110001111

perhaps any 6s

111010 = 8

011011 = 8