COA

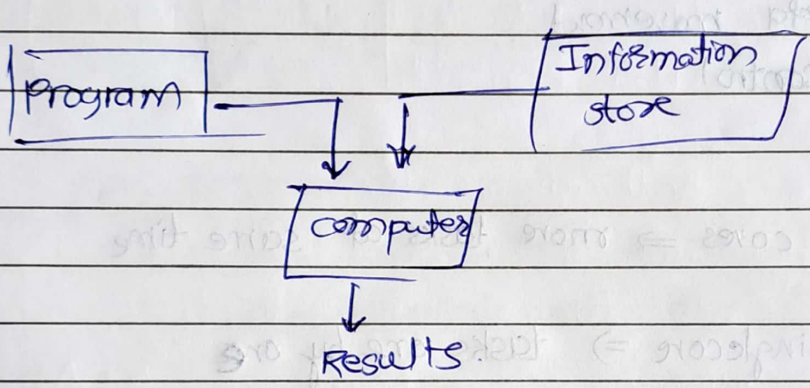
classmate
Date
Page

Textbook :- (3)

Dr. Anjali Gautam

Computer Architecture
* software
* ~~set of~~ instruction set Architecture

computer organisation
* Hardware
*



AGP slot - Graphics card
PCI slot - sound card, etc.
copper coil ← to remove any power spike.

<u>Northside</u>
RAM, CPU

<u>Southside</u>
PCI, AGP

* [Bus helps in traversing data]

Von Neumann developed basic Architecture of comp.

most famous 1st Gen comp :- IAS computer
possessed memory so also called
stored-program concept.

Function:-
Data processing
Data storage
Data movement
control

more cores ⇒ more tasks at same time

singlecore ⇒ tasks one by one

volatile – RAM (data will be there till power)

non-volatile – ROM, PROM, EPROM, EEPROM

1 byte = 8 bit
1 word = 8 byte.

32 bit computer :- bus transfer 32 bits at a time.

1st Gen comp:-

Drawback:- cathode-anode data transfer time ↑.
:- insect went inside vacuum tube.
:- Glass covering

2nd Gen comp:-

:- punch cards for input & printers for output.
:- metal covering

Drawback:- Lot of heat.

3rd Gen comp:-

- Used ICs, contains thousand of transistors
- instead of punch card, used keyboard.

4th Gen comp:-

- micro-processor containing many ICs, built on silicon chip made by Intel.
- Addition of GUI.

5th Gen comp:-

- Today. (1980 - present)
- AI • nanotech • NLP • etc.
- Multicore processor

Moore's law :- Increased density of components on chip.
:- no. of transistor double every year.
:- cost of chip is unchanged
:- High package density & shorter electrical paths, giving high performance.

Machine instructions

Assembly Language
- MUL R5, R0, R1
- DIV R6, R2, R3

(R6 = address)

Inst. set format

family	operation	input 1	input 2
10	0000001	0001	0011

10 → arithmetic operation.

MIPS:-

High $\xrightarrow{\text{compile}}$ Low $\xrightarrow{\text{assemble}}$ Machine

Level of abstraction:-

- machine code
- Assembler Language.
- Procedural Languages.
- Object-oriented Language.
- Modelling languages.

Tutorial-1

Name:- Saurav Vishnu Gitte
Class:- Sec A
Enroll. No:- IIT2022066

Q.1)

Computer organisation	Computer Architecture
• The actual implementation of a computer in Hardware	• The view of computer presented to software developers.
• The organisation describes how it does it.	• Architecture describes what the computer does.

Computer structure	Computer function
• The way in which ~~can~~ components are interrelated	• The operation of each individual component as part of structure.

Q.2) Four main functions of computer are:-
- Data processing.
- Data storage
- Data movement.
- Control

Scanned with OKEN Scanner

27 March

- size ↓
- speed ↑
- cost ↑.

↑
Registers.
Cache
main memory
RAM.
Magnetic tape

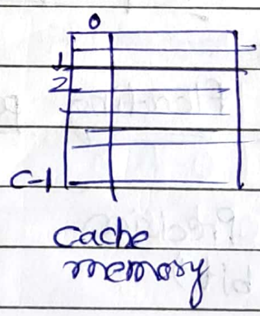
Registers (32).

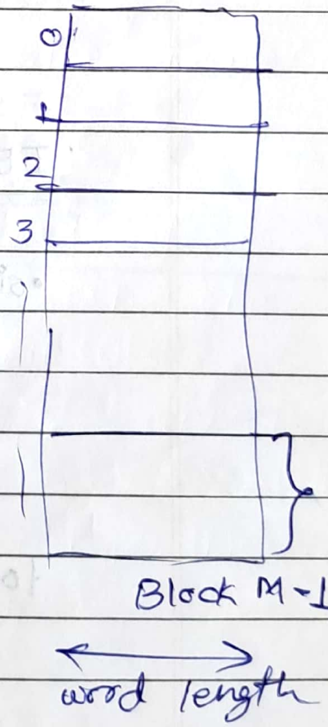
(MIPS Architecture)   ADD R_1 , R_2 , R_3.
$\$t_0$ $\$t_1$ $\$t_2$

cache

- small in size
- very fast.
- sits betn CPU and main memory.

C < M.





Cache Types

- L1
- L2
- L3

CPU ⇄ L1 ⇄ L2 ⇄ L3 ⇄ Main memory

← speed fast

0111
7 1
1000

classmate
Date
Page

2's complement

8 4 2 1

ways of representing zero → 0000

only 1 way ~~1000~~

$\boxed{1000 \text{ is given to } -8}$

* Floating Point Representation :-

$\pm b.bbbb * 2^{\pm exp}$

3 fields :- sign, exponent, fraction

S	Exp	Fraction

IEEE floating point Formats :-

• single Precision (32 bit)

• Double precision (64 bit)

1024 + 256.

1024
128
1152
64
1216
32
1248
16
1264
12
1276
7

1024 + 128 + 64 + 32 + 8 + 2 + 1

2^{10} 2^7 2^6 2^5 2^3 2^1 2^0

$(10011101011.001)_2$.

Normalize

1 must be present here.

$$\boxed{1.00111\ 01011001 \times 2^{10}}$$

S = 0 for +ve
1 for -ve.

0	10001001	0011101011001 00000

31 30 23 22 0

S Exponent Mantissa

Instead of 2's comple, they introduced bias factor of 127.

$$\begin{array}{r} 127 \\ +\ 10 \\ \hline (137)_{10} \end{array} = (10001001)_2$$

If exponent was -10, it becomes $127 - 10$

$(binary)_2 = (117)_{10}$

ASCII code

A — 65
B — 66
|
Z 90

0 — 48
1 — 49
|
9 — 57

a — 97
b — 98
|
z — 122

Scanned with OKEN Scanner

COA Tutorial 2

Q.1)

1.
```
   1
  0110
+ 0101
------
  1011
```

2.
```
   1
  0101
+ 0101
------
  1010
```

3.
```
   11
   1011
+  1110
-------
  11001
```

4.
```
  1001
+ 0110
------
  1111
```

5.
```
   1
   1010
+  1011
-------
  10101
```

6.
```
   111
   1111
+  1001
-------
  11000
```

7.
```
   1
  0110
+ 0101
------
  1011
```

~~Q.2)~~

~~1)~~

	8421		8421		8421
	0110	+	0101	=	1011
	(4+2)		(4+1)		(8+2+1)
	6	+	5		11

~~2)~~

	8421		8421		8421
	0101	+	0101	=	1010
	(4+1)		(4+1)		(8+2)
	5		5		10

Scanned with OKEN Scanner

$(-0.75)_{10} \longrightarrow$ convert to floating point Represent

Arithmetic

1) Addition
2) Multiplication

binary $(-0.11)_2$

$\boxed{1}$ - $\boxed{}\boxed{}\boxed{}$ 0.11

1

normalize.

$\therefore -1.1 \times 2^{-1}$

$e = 127 - 1$

$= (126)_{10}$

126

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

sign, e, mantissa — 16 bit.

	1						

$9.999_{10} \times 10^1 + 1.610 \times 10^1$

$$\begin{array}{r} 9.999 \\ 1.610 \\ \hline 11.609 \end{array}$$

$\Rightarrow 11.609 \times 10^1$

$\Rightarrow 1.1609 \times 10^2$

convert to binary & get answer

~~0 0 0 1 0~~

exp man

Q) $9.999_{10} \times 10^{1} + 1.610_{10} \times 10^{-1}$

$\Rightarrow \cancel{99.99} + 1.610 \times 10^{-1}$

999.9×10^{-1}

$\Rightarrow \cancel{101.600 \times 100}$

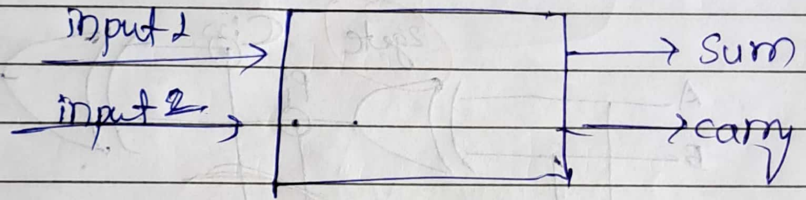
$$\cancel{\begin{array}{r} 99.99 \\ +\ 1.610 \\ \hline 101.600 \end{array}}$$

$\Rightarrow 1001.510 \times 10^{-1}$

$\Rightarrow 1.001510 \times 10^{2}$

$$\begin{array}{r} 999.9 \\ 1.610 \\ \hline 1001.510 \end{array}$$

Half Adder :-



K-Map for Half adder.

S

	0	1
0		1
1	1	

C

	0	1
0		
1		1

a) $S = X\bar{Y} + \bar{X}Y = X \oplus Y$

$S = (X+Y) \cdot (\bar{X} + \bar{Y})$

$C = X \cdot Y$

$= \overline{(\overline{X \cdot Y})}$

Best

b) $S = (X+Y) \cdot (\bar{X} + \bar{Y})$

$C = X \cdot Y$

not good

c) $S = \left(\overline{C + \bar{X} \cdot \bar{Y}}\right)$

$C = X \cdot Y$

not good

Full-Adder

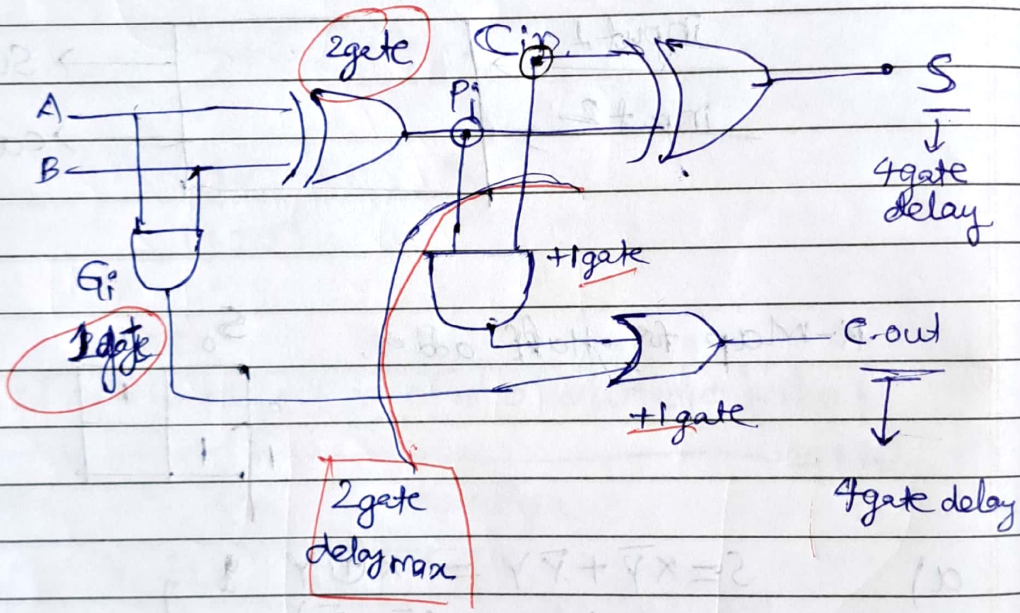
for 0 carry in, it is same as half adder
for 1 carry in it, it behaves like Full adder.

$$S = X \oplus Y \oplus Z$$

$$C = \underbrace{XY}_{\text{carry generate}} + \underbrace{(X \oplus Y)Z}_{\text{carry propagate}}$$

carry generate — generated from input directly

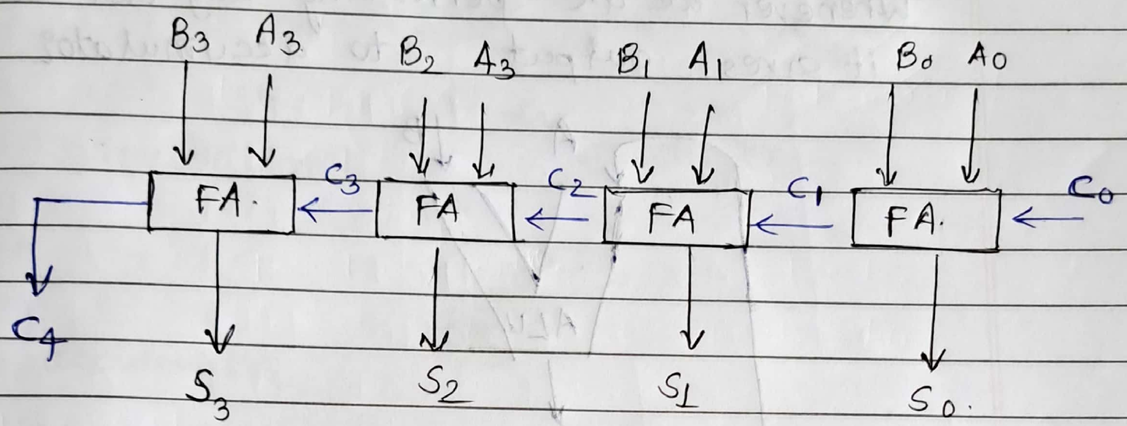
carry propagate — propagated from another output



Alternative :- Carry-look-ahead
(no delay)

classmate
Date
Page

4 Bit Ripple - Carry Binary Adder (Delay)



ex:-

	A_3	A_2	A_1	A_0
	0	0	1	1
+	0	1	0	1
	B_3	B_2	B_1	B_0

C_1 = 1, C_0 = 0

	0	0	1	1
+	0	1	0	1
	0	1	0	10
	S_4	S_2	S_1	S_0

Drawback • Untill C_2 is not available,
S_2 cannot be produced.

• C_4 has to wait from C_3 to born.
• Hence ripple effect & delay adds up.

Q. why other sum S_0, S_1, S_2 ignored?
ans:- carry का 20 gate delay होगा तो sum हो जाएगा sabka except final S_3

AND/OR → 1 gate delay
XOR → 2 gate delay
∴ Gate delay for sum → 4
for carry-out → ~~3~~ 4 gate.

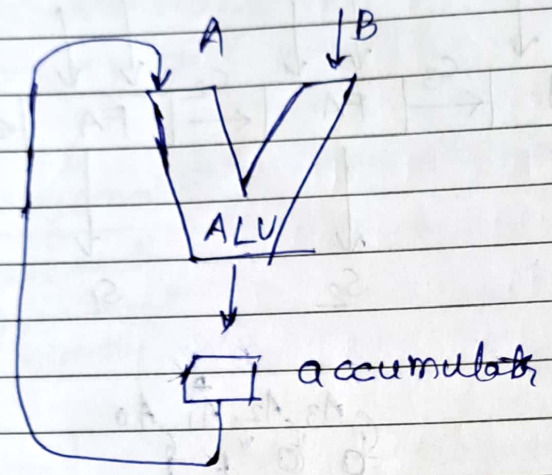
∴ Total gate delay ⇒ 4 carry delay + last sum delay
⇒ $(4\times4) + (1\times4)$
⇒ $\boxed{\text{20 gate delay for ripple}}$

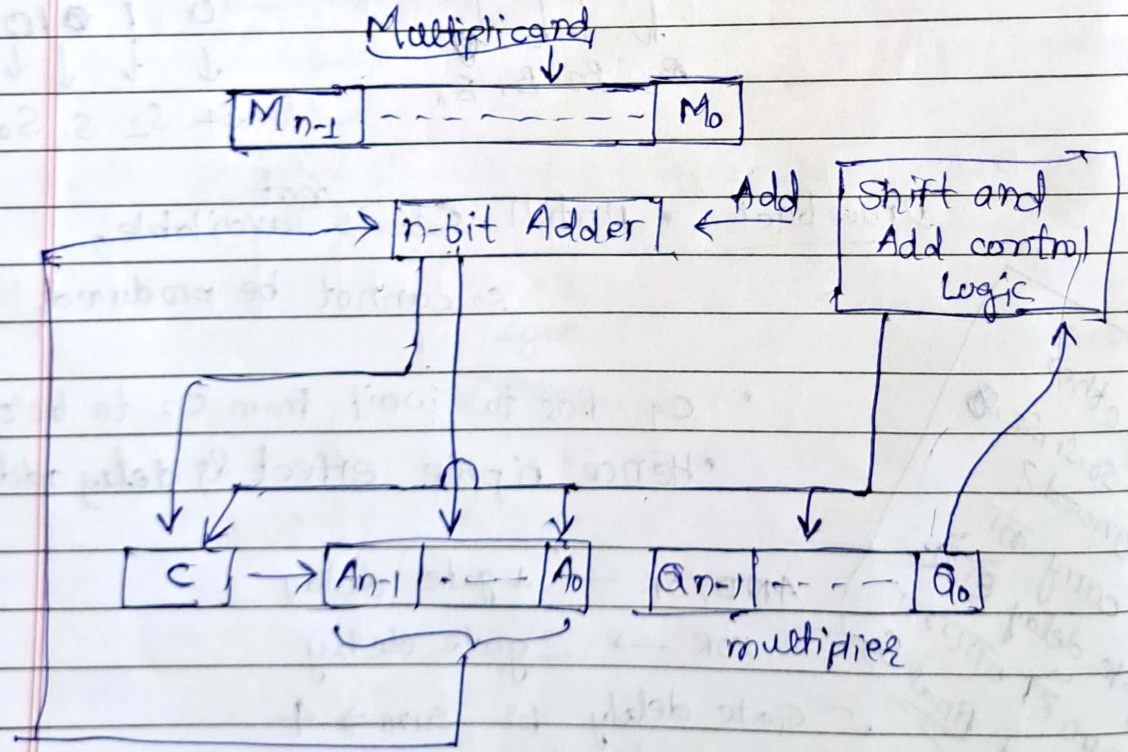
Scanned with OKEN Scanner

Multiplication

Shift and Add Algorithm

whenever we are performing any task on ALU it saves output to accumulator



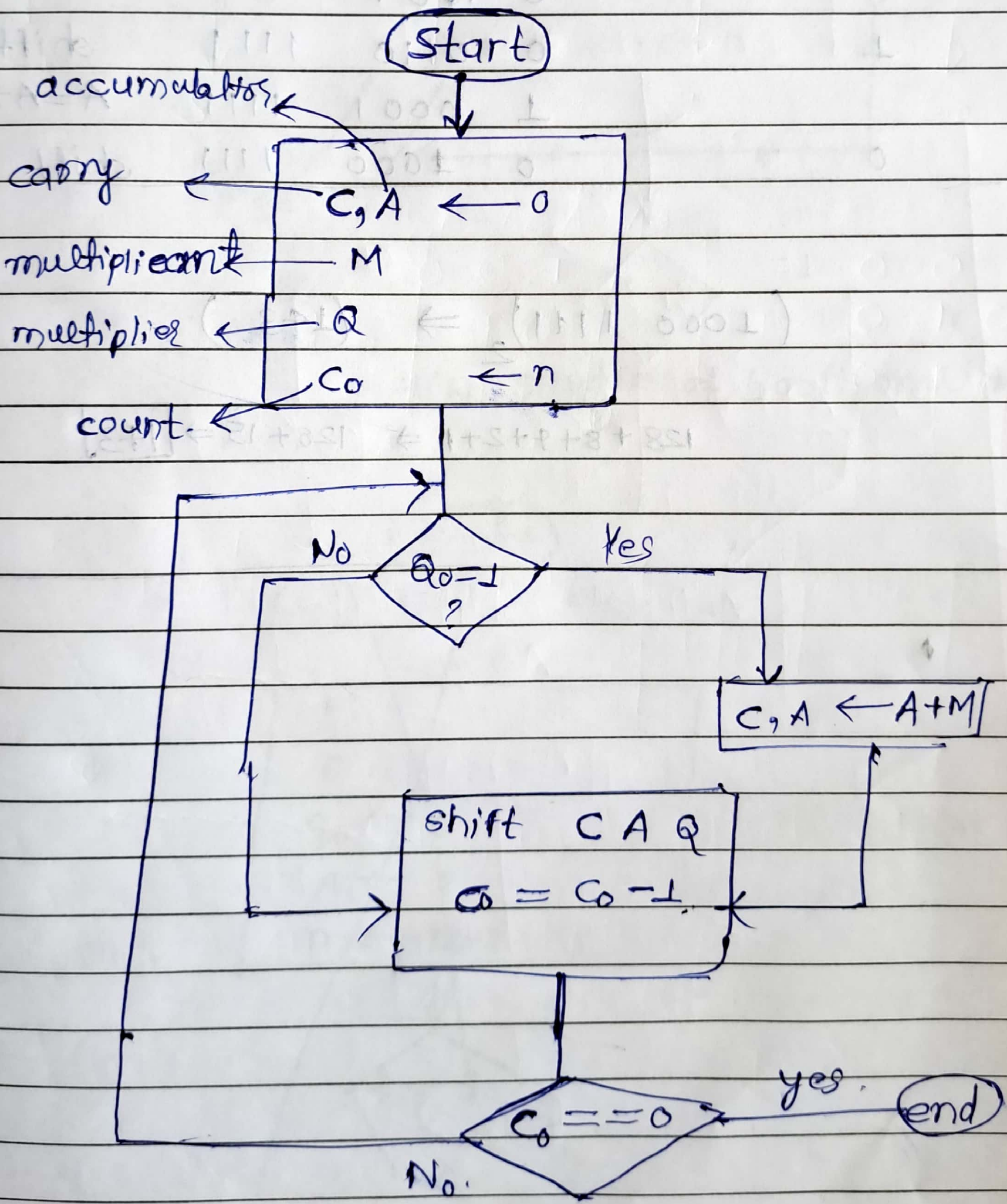


$C_0 \longrightarrow$ count.

11×13

11 → multiplicant

13 → multiplier.



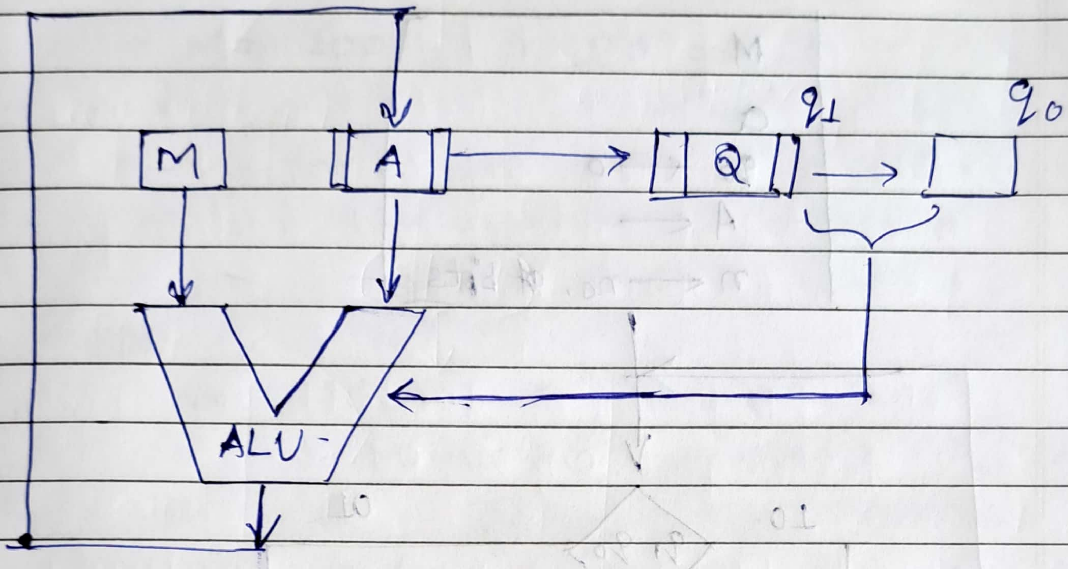
23.9L9190
1101

count	M	C	A	Q	operation
4	1011	0	0000	1101	initialisation
		0	1011	1101	A=A+M
3		0	0101	1110	shift
2		0	0010	1111	shift
		0	1101	1111	A=A+M
1		0	0110	1111	shift
		1	0001	1111	A=A+M
0		0	1000	1111	shift

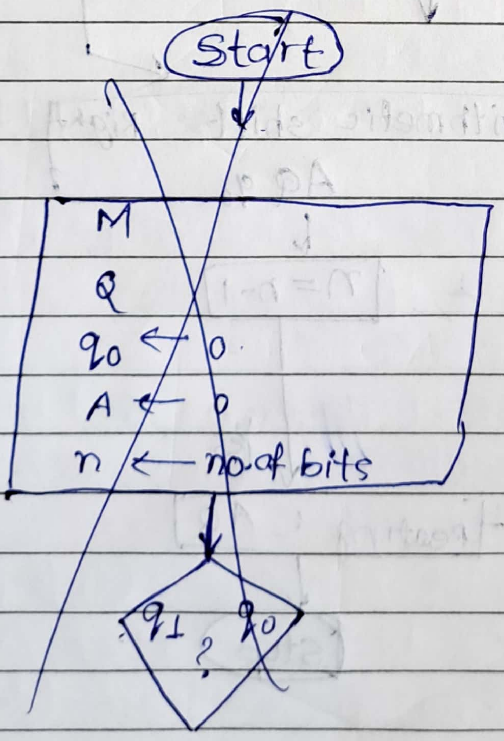
$(1000\ 1111)_2 \Rightarrow (143)$

$128+8+4+2+1 \Rightarrow 128+15 \Rightarrow \boxed{143}$

BOOTH MULTIPLIER:-

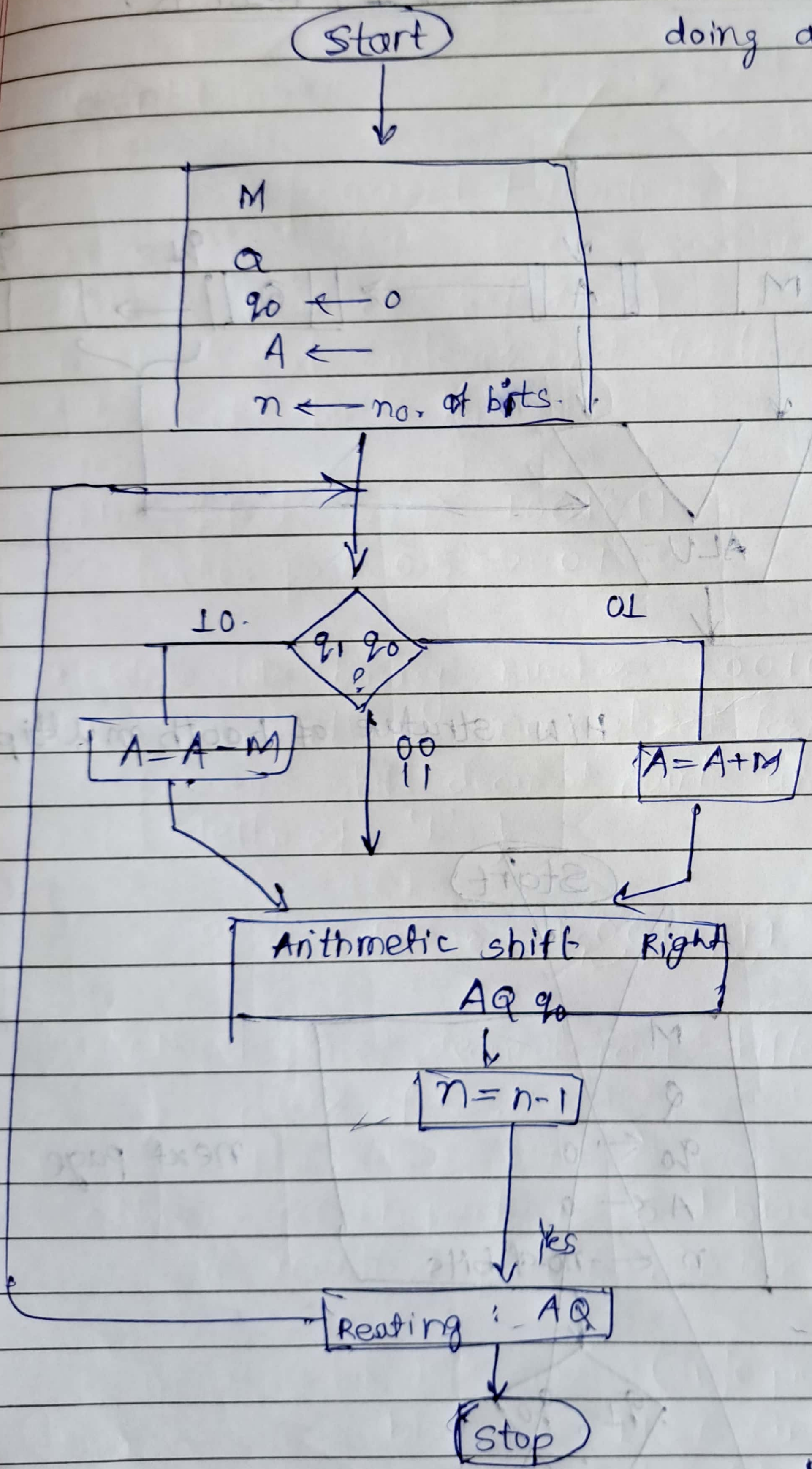


H/W struture of booth multiplier.

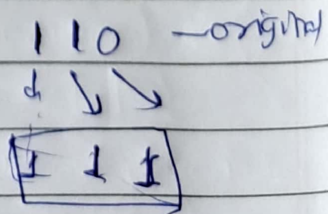


next page

Here we are doing arithmetic ship.



Shifting meaning



Ex:- -7×3

$-7 \rightarrow M$, $3 \rightarrow Q$

$M = 1001_2$ in 2's comp

$-M = 0111_2$

n	A	Q	q_0	operation
4	0000	0011	0	initialized
	0111	0011	0	A = A − M
	0011	1001	1	shift AQq_0
	0001	1100	1	shift AQq_0
	1010	1100	1	A = A + M
	1101	0110	0	shift

COA Tutorial 3

IEEE format :-

23 bit (single)

52 bit (double)

| sign | exponent | mantissa |

- Take a number. say 75.5
- separate its integer and fraction part
- convert to binary

integer part 75 ⇒ 1001011

fraction part 0.5 ⇒ 0.1

- Write total binary number 1001011.1
- Now convert to normalized form.

{ - means shift decimal point untill it reaches leftmost '1' }

ans: 1.0010111×2^6

- No. of places, decimal point shifted, represents power of 2.

- Now, exponent part is 6 + bias
  & mantissa part is number after decimal.

- In 32 bit representation (single precision)
  bias is +127 (in 8 bits)

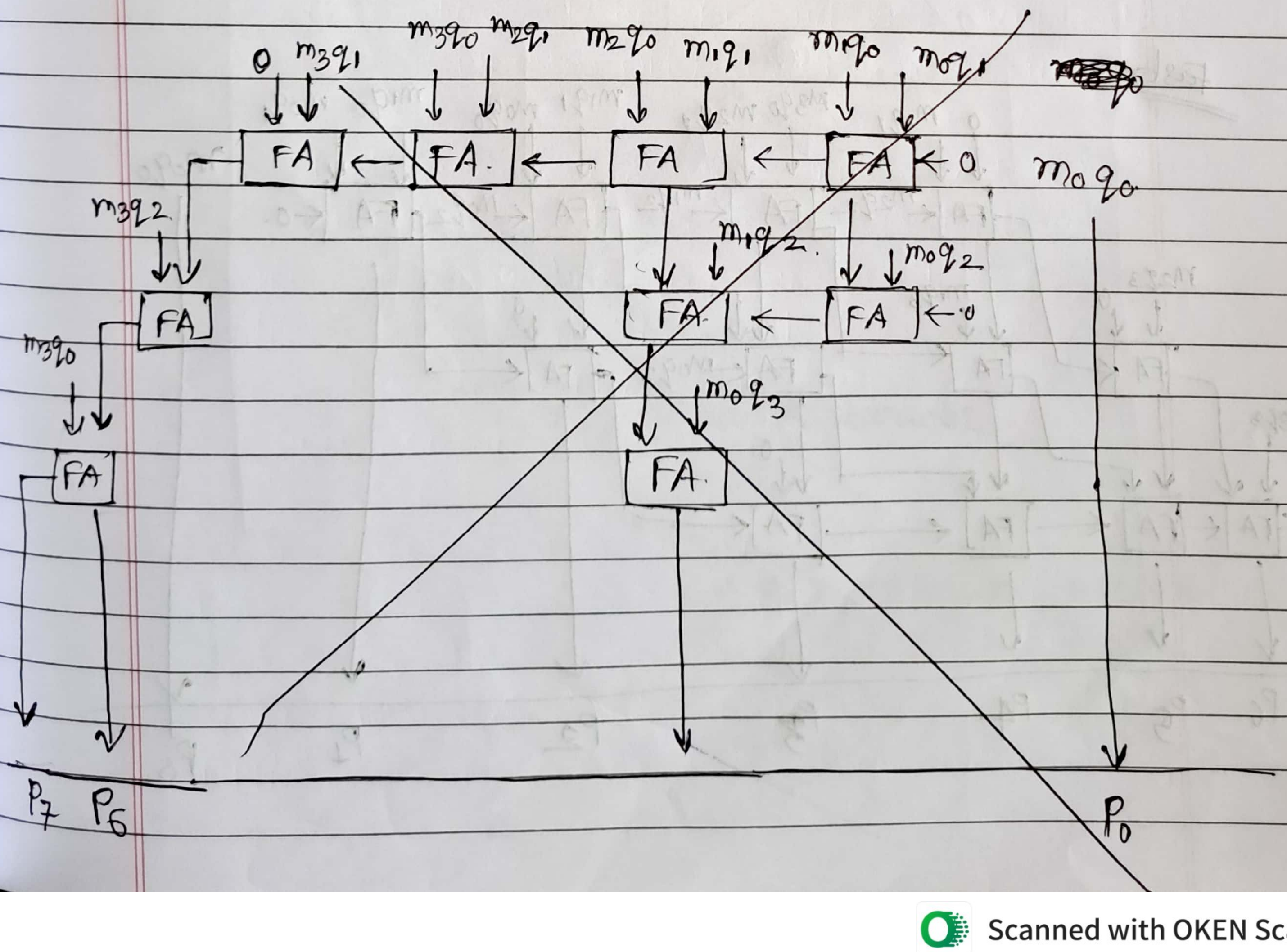
  In 64 bit representation (double precision)
  bias is +1023 (in 11 bits)

Scanned with OKEN Scanner

classmate
Date
Page

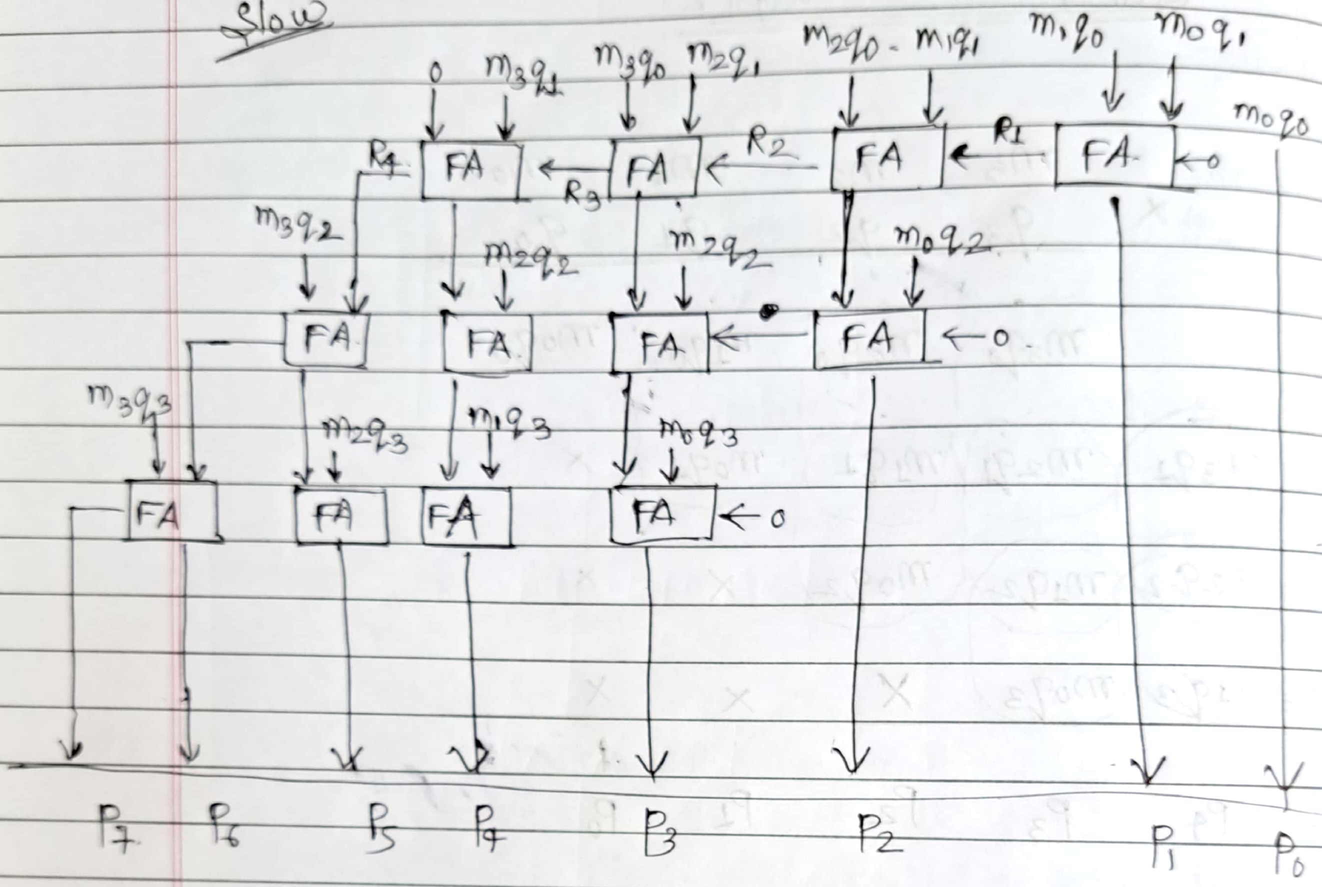
Carry Save Multiplier

				m_3	m_2	m_1	m_0
			$\times$	q_3	q_2	q_1	q_0
				m_3q_0	m_2q_0	m_1q_0	m_0q_0
			m_3q_1	m_2q_1	m_1q_1	m_0q_1	$\times$
	m_3q_2	m_2q_2	m_1q_2	m_0q_2	$\times$	$\times$	
m_3q_3	m_2q_3	m_1q_3	m_0q_3	$\times$	$\times$	$\times$	
P_6	P_5	P_4	P_3	P_2	P_1	P_0	

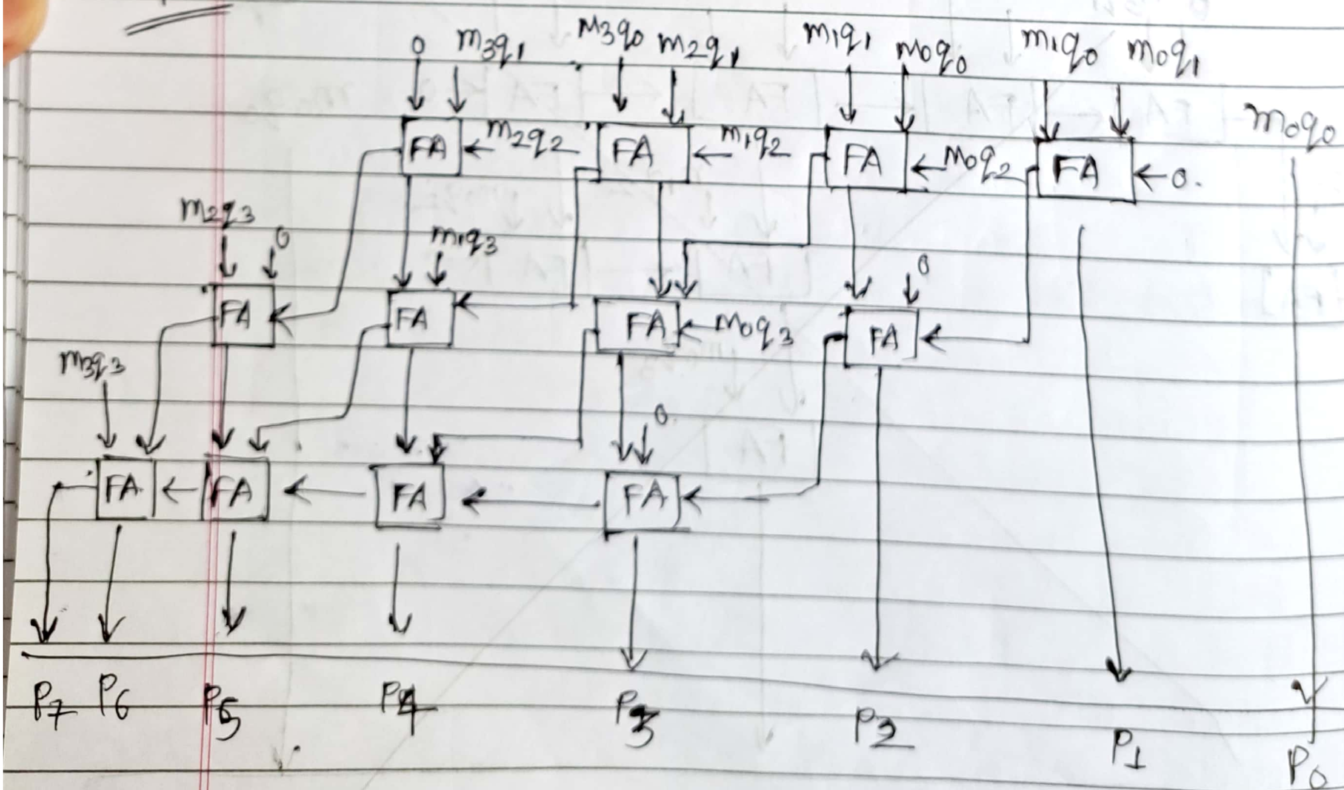


classmate
Date
Page

Slow



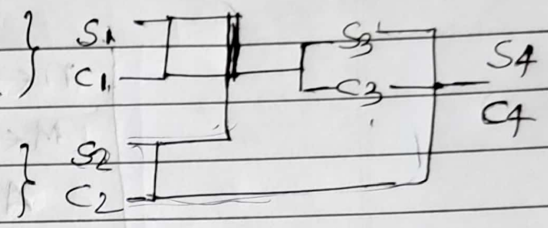
Fast



M → 1 0 1 1
Q → 1 1 0 1

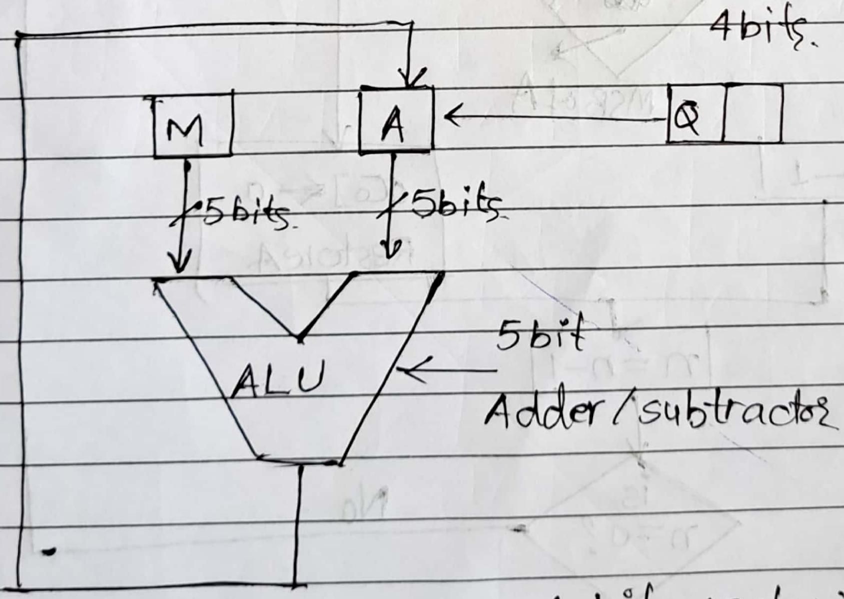
1 0 1 1 → A
0 0 0 0 + → B
1 0 1 1 + + → C
1 0 1 1 + + + → D

1 0 0 0 1 1 1 1



÷ Division :-

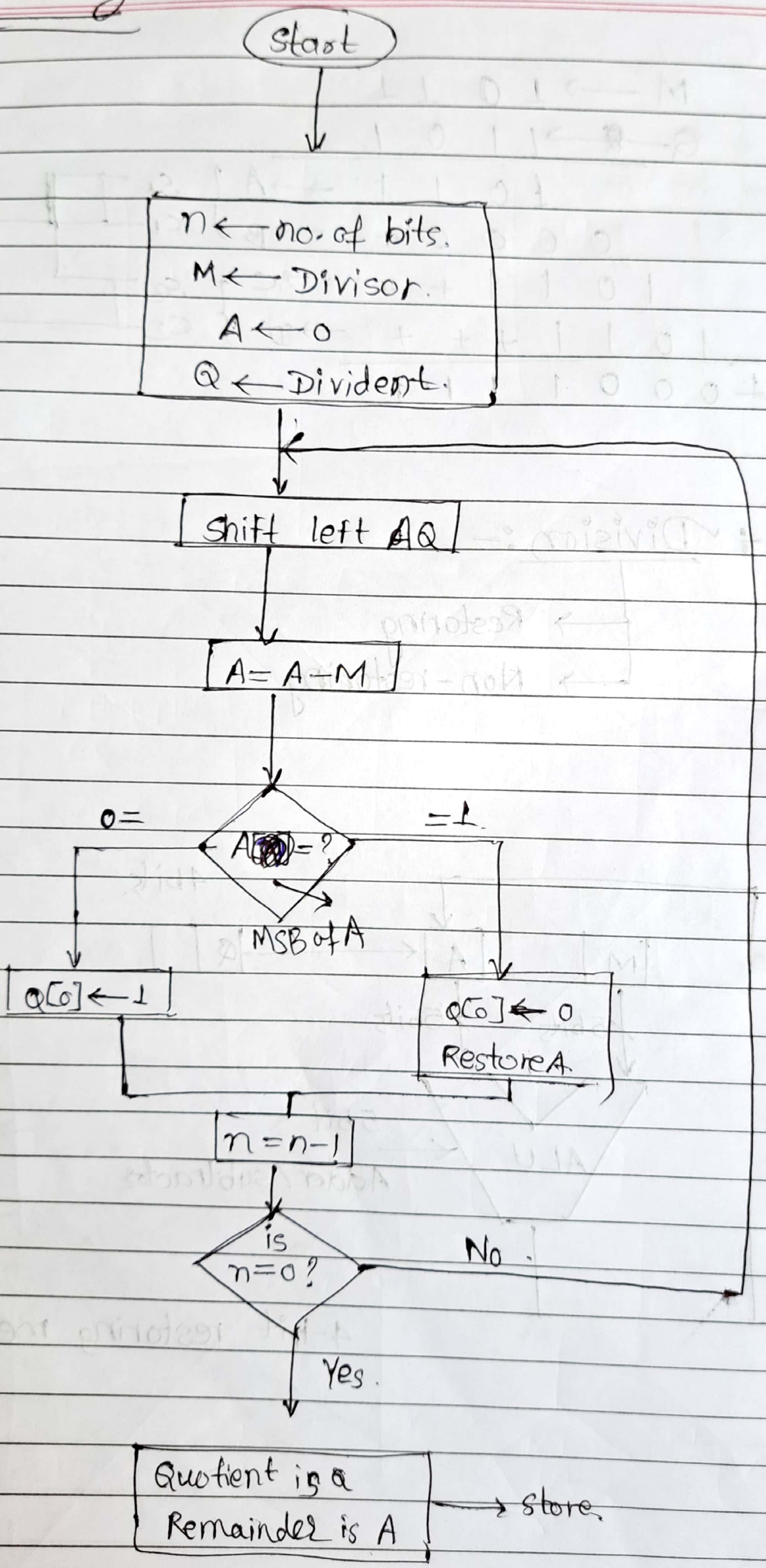
- → Restoring
- → Non-restoring



4-bit restoring method

classmate
Date
Page

Restoring



Scanned with OKEN Scanner

$$\begin{array}{r} 0\,0\,0\,0\,0\,2 \\ -\ 0\,0\,0\,1\,1 \\ \hline 1\,1\,1\,0\,1 \end{array}$$

classmate

Date ________

Page ________

Q) 11/3 ⇒ 3/2 A

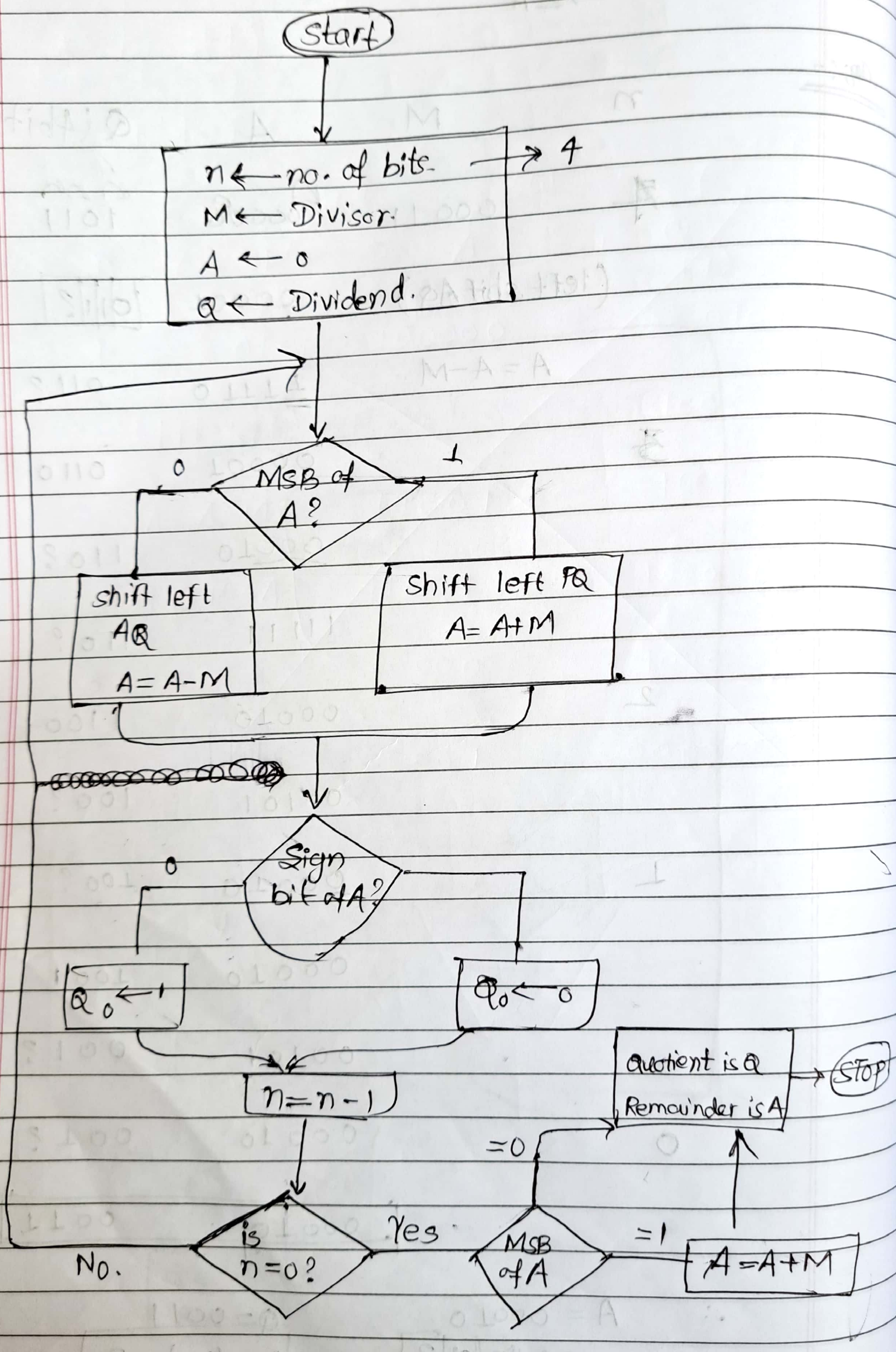
ans :-

n	M	A	Q (4bit)	
~~4~~	00011	00000	1011	
	(left shift AQ)	00001	011?	
	A = A − M	11110	011?	$Q_0 = 0$ Restore A
~~3~~		00001	0110	
		00010	110?	
		11111	110?	
2		00010	1100	
		00101	100?	
1		00010	100?	
		00010	1001	$Q_0 = 1$
		00101	001?	
0		00010	001?	
		00010	0011	

∴ A = 00010   Q = 0011

remainder = 2   quotient = 3

Non-restoring



Q. 11/3 ?

ans.

n	M	A	Q
4	00011	00000	1011
		00001	011 ?
3		11110	0110
		11100	110 ?
2		11111	1100
		11111	100 ?
1		00010	1001
		00101	001 ?
0	0	$\boxed{00010}$	$\boxed{0011}$

classmate

Date ________

Page ________

Scanned with OKEN Scanner