

Bitwise logic operations

$$P, R_1 \leftarrow R_1 \oplus R_2 \quad \text{XOR}$$

~~Input one goes to P and U1 Control Input~~

Let 16 Bit JSDS R1, R2 be $R_1 \oplus R_2$

$$\begin{array}{r} \text{Given bits } R_1 | 010 \\ \text{Given bits } R_2 | 100 \\ \hline \text{XOR} \end{array}$$

$$0110$$

XOR

For 16 Bits JSDS XOR, OR, And 1st operation pulse (i.e. let R_1 = Result)

$$P + Q: R_1 \leftarrow R_2 + R_3, R_4 \leftarrow R_5 \vee R_6$$

One input goes to P, Q and one input goes to R_1, R_4 (OR) gives R_1, R_4 result

\rightarrow Control Signals / Inputs (OR)

\rightarrow Registers (Addition / Plus)

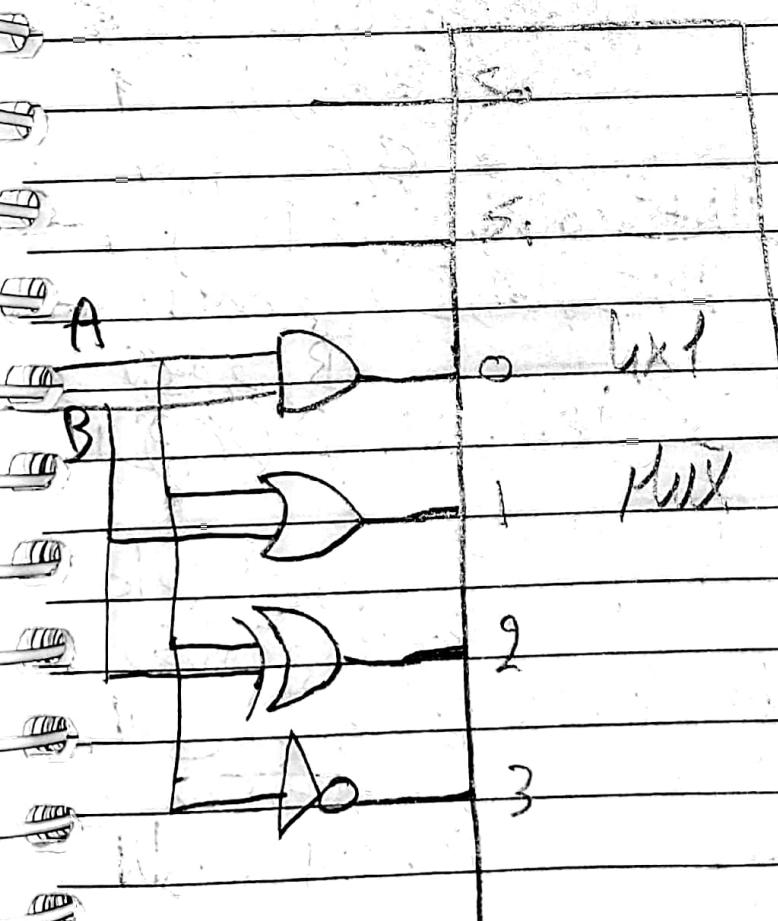
\rightarrow And

\rightarrow OR

\rightarrow XOR

\rightarrow NO

S_0	S_1	Output
0	0	$E = A \wedge B$ And
0	1	$E = A \vee B$ OR
1	0	$E = A \oplus B$ XOR
1	1	$E = \bar{A}$ NOT



Select 1 logic gate

All operations can be done

using

- [1] $S_0=0, S_1=0 \Rightarrow 0 \Rightarrow \text{And}$
- [2] $S_0=0, S_1=1 \Rightarrow 1 \Rightarrow \text{OR}$
- [3] $S_0=1, S_1=0 \Rightarrow 2 \Rightarrow \text{XOR}$
- [4] $S_0=1, S_1=1 \Rightarrow 3 \Rightarrow \text{NOT}$

Applications on Bitwise Operators

PAGE

DATE

Selective Ed

Selective Set ~~is Register~~ ~~is used~~
~~one bit~~ ~~in all~~ ~~bits~~
one ~~bit~~ ~~in~~ ~~all~~ ~~bits~~ ~~of~~ ~~the~~ ~~register~~
in ~~one~~ ~~bit~~ ~~in~~ ~~all~~ ~~bits~~ ~~of~~ ~~the~~ ~~register~~

Selective Complement

Selective Complement جزء باباً فيRegA جزء
المحض مع جزء مع جزء
جزء A ناتج وأكاد غير أقل وآخر جزء RegA
جزء B ناتج وأكاد غير أقل وآخر جزء ones
جزء C ناتج وأكاد غير أقل وآخر جزء B

Selective Class

Selective Clear Bits set by clear decimal
Reg A & One
if D1 is one, all clear after it will be 0
B is also set if A is set
for example

Mask A 3 Bits (Clear) \rightarrow clear few bits
B 11 (01101011) \rightarrow zero few bits
B 11 (01101011) \rightarrow one few bits

as done in C & AUDI

Insert depicts A 11 (Binary) \rightarrow one
bit value kept with original bits
as done above

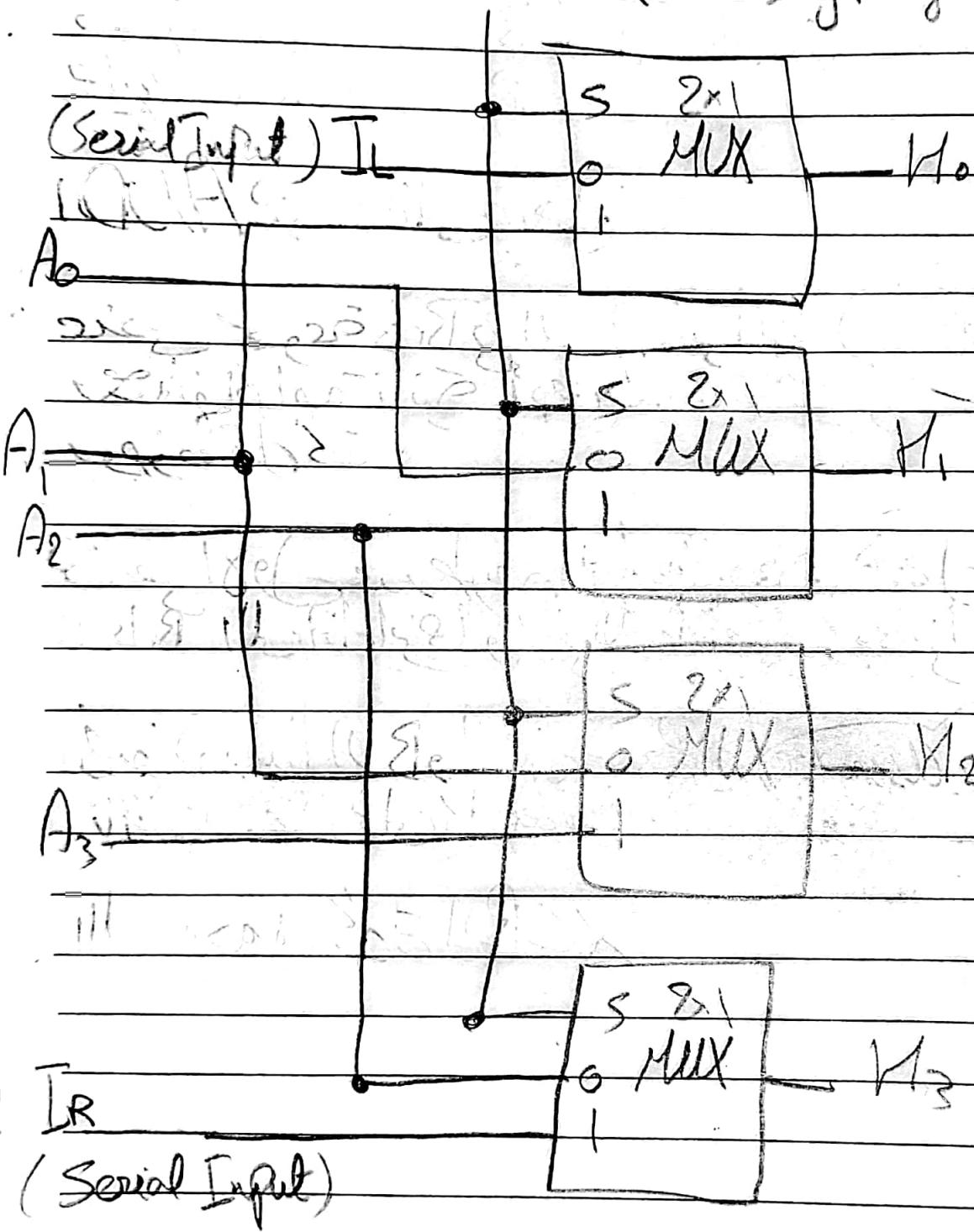
Match \rightarrow (01101011) Masking reflects 101 all
bits in A 11 (01101011) will be 101

Clear Using XOR \rightarrow A, B will give
zero as result because

XOR is as done in

4 Bit Shifter

(1 for Shift Left (down))
 Select (0 for Shift Right (up))



Select

Output

S

H₃H₂H₁H₀

0

I_RA₃A₂A₁

1

A₂A₁A₀I_L $(S=0 \rightarrow \text{Shift + Right})$ alg. B zero 1 bit input 11 \rightarrow MUX 11
~~11 is output 11~~H₃ \rightarrow I_RH₂ \rightarrow A₃H₁ \rightarrow A₂H₀ \rightarrow A₁

Input 11 and output 11 A 11

H₀ 11 is value 11 \rightarrow I_R and 11 is value(S = 1
shift left)one 11 bit input 11 \rightarrow MUX 11
H₁, H₂ 11 output valueH₃ \rightarrow A₂H₂ \rightarrow A₁H₁ \rightarrow A₀H₀ \rightarrow I_L2nd input 11 \rightarrow I_L \rightarrow 11 is value 11 is value

Shift Types

Logical

Arithmetic

Circular

~~Bit 11~~ Bit 11 ~~zero~~ When all ~~zero~~
 (IR/It) zero / one ~~zero~~ ~~sign~~ ~~positive~~

logical \rightarrow

zero \rightarrow Bit 11

IR = 0, It = 0

(unsigned Integer, sign bits) 11 ~~zero~~

Shift Right $\rightarrow \frac{\circ}{2}$, Floor

Integers

[ex, Floor(7.5) = 7]

Shift left $\rightarrow \times 2$ overflow

(A₃=1) 2nd Bit sign

Arithmetic

Shift Right (IR=A₃)

zero

Signed Int
Sign)

A₃ A₂ A₁ A₀

IR = 0

one

A₃ A₂ A₁ A₀

IR = 1

Shift left

$$I_L = \cancel{7.20}$$

Significant Figures in Arithmetic

Shift Right

~~Shift left~~

o 2 Flaschensplitting

~~X2 also~~

$$\text{floor}\left(\frac{-15}{2}\right) = \boxed{-7.5}$$

Overflow

~~Des Int~~

~~لابد من الـ وـ الـ وـ~~

~~mis Agh illal~~

Circulus

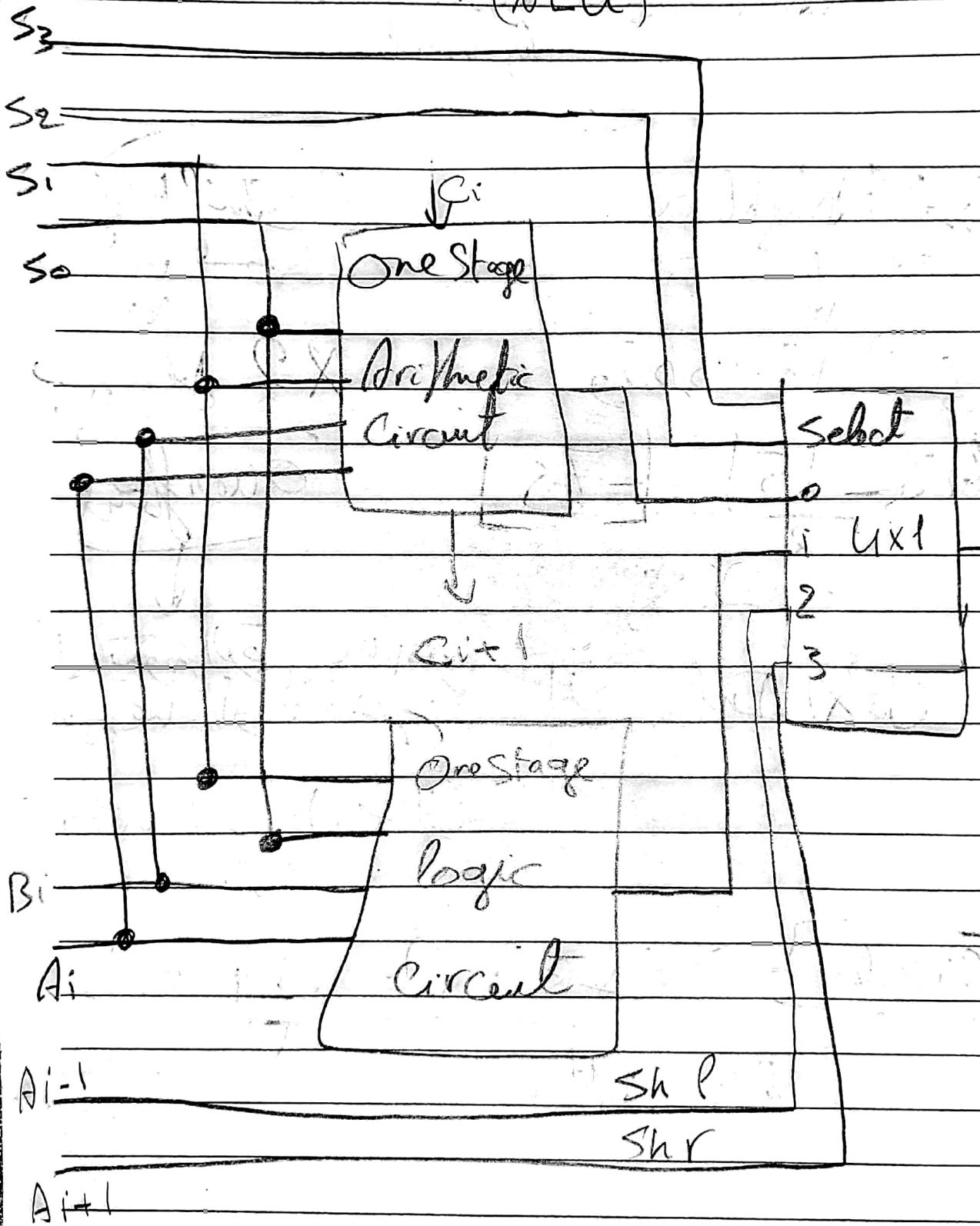
$$f_R = 10$$

$$\overline{J_1 = A_3}$$

Shift Right ↲

Shift left

Arithmetic Logic Shift Unit (ALU)



ACU Being An Input zip to Combinational
Fn output zip circuit

Output
F

Arithmetic opern (+, -, ×, ÷)
Logic opern (\bar{A} , V, +, etc)
Shift left/ Right

4 Bit Arithmetic dev (add, Is, reg, all carry)
Selects (Is, 1'st, etc) - carry, Subtract Add few.
D) Jiggle Output (cin, 1, 0, So, S)

MIX (A, B) Is 1'st Bit Jiggle if BC > 0 else
Bits 1 to 4 Full adder

logic opern (inv, CS, mul, Jiggle, all carry)
(Is, Jiggle, Selects) \rightarrow (Not, XOR, OR, AND, Inv)
Bits 1 to 4

Arithmetic logic is one bit dev circuit Jiggle
SS (Jiggle, Inv)

Bit Inv, Jiggle and output zip to MUX dev -
Result \rightarrow F1) Jiggle Result 1 zip to

4 bit width zip

32 Bit ALU

PAGE _____
DATE _____

1) $D_i \rightarrow \text{MUX}_1$
~~>Selects one stage of Arithmetic output circuit~~

Arithmetic Circuit of One Stage =

32 Bit ALU

2) $E_i \rightarrow \text{MUX}_2$
~~One Stage Bitwise logic circuit to outputs~~

3) Shift Left Output $\rightarrow \text{MUX}_3$

4) Shift Right Output $\rightarrow \text{MUX}_4$

Ex Selector Based on S_2, S_3 on MUX_1

Input of one Stage of logic circuit

and (S_2, S_3) of MUX_1 are E_i (One bit Input), B_i (One bit Input), S_1, S_0 of Input bit

Input a one Stage Arithmetic circuit

Input A, B, b_1, b_2, D as Input Output D, C, b_1
 ملخص i فال Input a being S_1, S_0 2 Selectors
 سیگنال داده Output $b_1, b_2, F.A$ j لی Carry C
 Index $i + 1$

Inputs $A, F.A$ و MUX میسر لیست Circuit
 2 Selectors, FA C لی Carry C A_i, B_i میسر
 MUX میسر b_1, b_2, D دیلیت Output D
 Next Stage N لی Carry C

Input a Output b Bit i
 A_{i-1} last Input b , Shift left

Input a Output b Bit i
 A_{i-1} last Input b , Shift Right

2 Circuits will output b (left) b (right)
 2 Selectors as Value S_1, S_0 2 Shifts
 MUX Input S_1, S_0, b, S_2, S_3 b (left)
 Output is b (right)