

Only math, No code

Lecture +1

Serial Communication Interfaces

- * Serial Interface → यह Data का transfer या, संकेत MSB Receiver → यह ३ राशिएँ → LSB प्रथम, Time after zero.
- * Parallel Interface → Data का transfer channel fixed & never change, Time का transfer.

- * Atmega328 has 3 types of serial communication interfaces:-

- ① Universal Synchronous Asynchronous Receiver and Transmitter (USART).
- ② Serial Peripheral Interface (SPI).
- ③ Two Wire Interface (TWI) / Inter-Integrated Circuit (I2C)

- * १st second → १ राशि bit transfer करेगी Or first १st band rate we कहे। (bps = bits per second)
- * Standard band rate := 2400, 4800, 9600, 19200, 38400... bps.
- * transmitter & receiver का band rate same होना To avoid communication error.
- * band rate का value → 2 \times n \times t₁ का product होता। OR
bps = १st band communication error rate,
- * Band width/error rate :- Standard band rate - Calculated band rate \times 100%
- * Standard band rate Question → जटिल 2022,

* Question → Given standard baud rate 7200 bps
 Calculated baud rate \approx ~~approx~~ standard baud rate
 use ~~2201~~

We know,
 Calculated baud rate = 5200
 \therefore Standard baud rate = 4800 (Nearest value)
 \therefore Baud error rate = $\frac{4800 - 5200}{4800} \times 100\% = -0.083\%$

\therefore There is no communication error.

* Band rate calculation \Rightarrow ~~for~~ to ~~for~~ operation mode ~~TMRI~~.

i) Asynchronous Normal Mode

ii) Asynchronous Double speed Mode

iii) Synchronous Master Mode.

Baud Rate (bps)	Operating Mode	Baud Rate Equation	Equation for UBRRn Values
f_{osc} (Hz)	Asynchronous Normal Mode	$\text{Baud Rate}_2 = \frac{f_{osc}}{16(\text{UBRR}_n + 1)}$	$\text{UBRR}_n = \frac{f_{osc}}{16 \times \text{Baud Rate}} - 1$
	Asynchronous Double speed Mode	$\text{Baud Rate} = \frac{f_{osc}}{8(\text{UBRR}_n + 1)}$	$\text{UBRR}_n = \frac{f_{osc}}{8 \times \text{Baud Rate}} - 1$
	Synchronous Master Mode	$\text{Baud rate} = \frac{f_{osc}}{2(\text{UBRR}_n + 1)}$	$\text{UBRR}_n = \frac{f_{osc}}{2 \times \text{Baud Rate}} - 1$

* Find the baud rate for the three operating modes when $f_{osc} = 1\text{MHz}$ and $UBRR_n = 25$. Calculate the baud error and comment whether there will be any communication error or not.

Soln:-

For asynchronous normal mode:-

$$\therefore \text{Baud rate} = \frac{f_{osc}}{16(UBRR_n + 1)} = \frac{1 \times 10^6}{16(25+1)} = 2409 \text{ bps}$$

$$\therefore \text{Baud error rate} = \frac{\text{standard baud rate} - \text{calculated baud rate}}{\text{standard baud rate}} \times 100\%$$

$$= \frac{2400 - 2409}{2400} \times 100\%$$

$$= -0.167\%$$

calculate time - 1
% error

\therefore There will be no communication error.

For asynchronous double speed mode:-

$$\therefore \text{Baud rate} = \frac{f_{osc}}{8(UBRR_n + 1)} = \frac{1 \times 10^6}{8(25+1)} = 9808 \text{ bps}$$

$$\therefore \text{Baud error rate} = \frac{9800 - 9808}{9800} \times 100\%$$

$$= -0.167\%$$

\therefore There will be no communication error.

for Synchronous master mode :-

$$\text{Baud rate} = \frac{f_{osc}}{2(CUBRR_n + 1)} = \frac{1 \times 10^6}{2(25+1)} = 19231 \text{ bps}$$

$$\text{Baud error rate} = \frac{19200 - 19231}{19200} \times 100\% \\ = 0.161\%$$

∴ there will be no communication error.

slide-14 (not important)

Lecture-2

Pulse Width Modulation (PWM)

* Non-Inverting Fast PWM Duty Cycle

$$OCRO_x = \frac{256D}{100} - 1 ; \text{ Here, } x = A \text{ or } B \mid D = \text{duty cycle.}$$

Q.1 A PWM signal is to have 75% duty cycle. Compute the value for OCROA.

Ans:-

$$\begin{aligned} OCROA &= \frac{256 \times 75}{100} - 1 \\ &= 192 - 1 \\ &= 191 \end{aligned}$$

* Inverting Fast PWM Duty Cycle

$$OCRO_x = 255 - \frac{256D}{100} ; \text{ Here, } x = A \text{ or } B \mid D = \text{duty cycle}$$

Q.1 A PWM signal is to have 75% duty cycle. Compute the value for

Ans:-

$$\begin{aligned} OCROA &= 255 - \frac{256 \times 75}{100} \\ &= 63 \end{aligned}$$

* Fast PWM Mode, $f_{OCAPWM} = \frac{f_{CK-10}}{N \times 256}$

* Phase Correct PWM Mode, $f_{OCAPCPWM} = \frac{f_{CK-10}}{N \times 510}$

Q.) Calculate the PWM frequency for the output when using Fast PWM Mode and Phase Connect PWM Mode When f_{CK-10} is 10 MHz and the Pre-scaler factors are 1, 8, 64, 256 or 1024. Comment on the result afterward.

Ans:-

The PWM frequency for the fast PWM Mode:-

$$f_{OCAPWM} = \frac{f_{CK-10}}{N \times 256} = \frac{10}{1024 \times 256} = 38.19 \text{ Hz} \quad | \text{ using Prescaler} = 1024$$

The PWM frequency for the phase correct PWM Mode:-

$$f_{OCAPCPWM} = \frac{f_{CK-10}}{N \times 510} = \frac{10}{1024 \times 510} = 19.19 \text{ Hz}$$

* Another type of fast PWM mode, if "OCROA" is given.

$$\text{: PWM frequency} = f_{OCOAPWM} = \frac{f_{CK-10}}{2N \times (1 + OCROA)}$$

Q.) Let's consider, $OCROA = 100$; Prescaler value, $N = 1$ and system clock frequency, $f_{CK-10} = 16 \text{ MHz}$

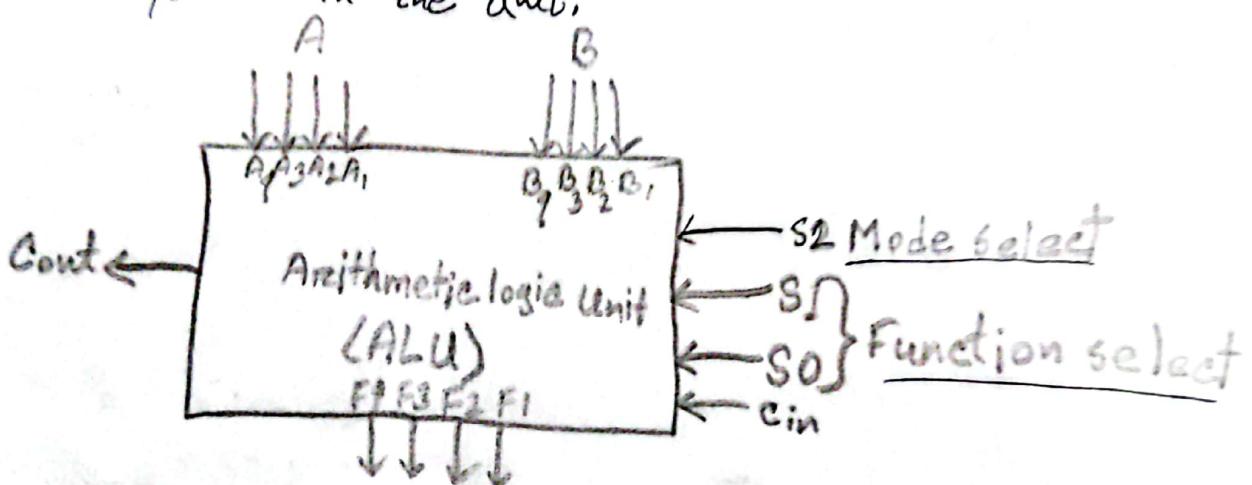
Ans:-

$$f_{OCOAPWM} = \frac{16 \times 10^6}{2 \times 1 \times (1 + 100)} = 79.2 \text{ kHz}$$

Lecture-3

Process Logic Unit

- ④ ALU can perform a set of basic arithmetic and logic operations.
- ④ The ALU ~~can~~ has several selection lines to select a particular operation in the unit.



⇒ S_2 distinguishes between arithmetic and logic operations.

⇒ $S_1 \& S_0$ specify the arithmetic and logic operations to be generated.

Design of Arithmetic Circuit

- ④ The basic component of the arithmetic section of an ALU is the parallel adder.

Lecture-6

slide-4

$$00 \rightarrow 0 \rightarrow 0$$

$$01 \rightarrow 1 \rightarrow 1$$

$$10 \rightarrow 2 \rightarrow S$$

$$11 \rightarrow 3 \rightarrow E$$

* If $S = 1$ Value 1 goes to load input to CAR
 " " " " " " " " increment CAR

* If $E = 1$ " " " " " " " " load input to CAR
 " " " " " " " " increment CAR

* ROM bits 13 & 14 no pin fill up extra pins

S use 130 130

E " " 131

S & E use no 131

0 & 0 use 131

Configuration

Table for the functions of the multiplexer select bits

ROM bits	13 14	MUX Select Function
0 0		Increment CAR
0 1		Load input to CAR
1 0		Load inputs to CAR if $S = 1$, increment CAR if $S = 0$
1 1		Load inputs to CAR if $E = 1$, increment CAR if $E = 0$

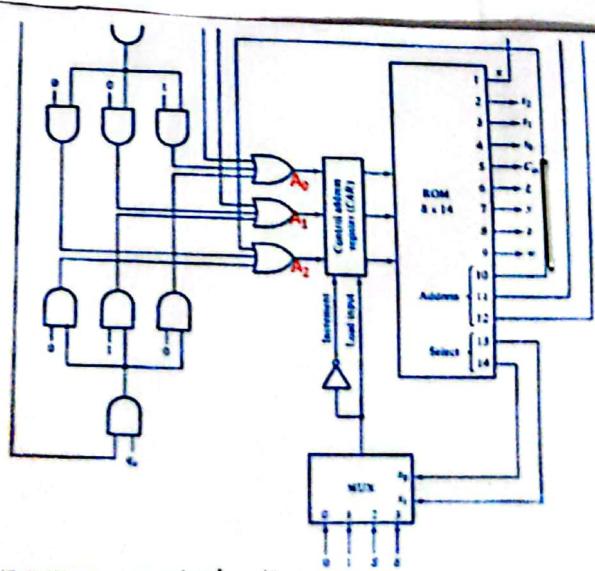


Fig. 10-10 Organization of the microprogram control unit

Microprogram for Control Memory

Table 10-2 Microprogram for Control Memory

ROM address	Microinstruction	Comments
0	$x = 1$, if ($q_s = 1$) then (go to 1), if ($q_o = 1$) then (go to 2), if ($q_s \wedge q_o = 0$) then (go to 0)	Load 0 or external address
1	$B_s \leftarrow \bar{B}_s$	$q_s = 1$, start subtraction
2	If ($S = 1$) then (go to 4)	$q_o = 1$, start addition
3	$A \leftarrow A + B, E \leftarrow C_{out}$ go to 0	Add magnitudes
4	$A \leftarrow A + \bar{B} + 1, E \leftarrow C_{out}$	Subtract magnitudes
5	If ($E = 1$) then (go to 0), $E \leftarrow 0$	Operation terminated if $E = 1$
6	$A \leftarrow \bar{A}$	$E = 0$, complement A
7	$A \leftarrow A + 1, A_s \leftarrow \bar{A}_s$, go to 0	Done, return to address 0

Microprogram sequence diagram:

T_0 : Initial state $x = 1$

T_1 : $B_s \leftarrow \bar{B}_s$

T_2 : nothing

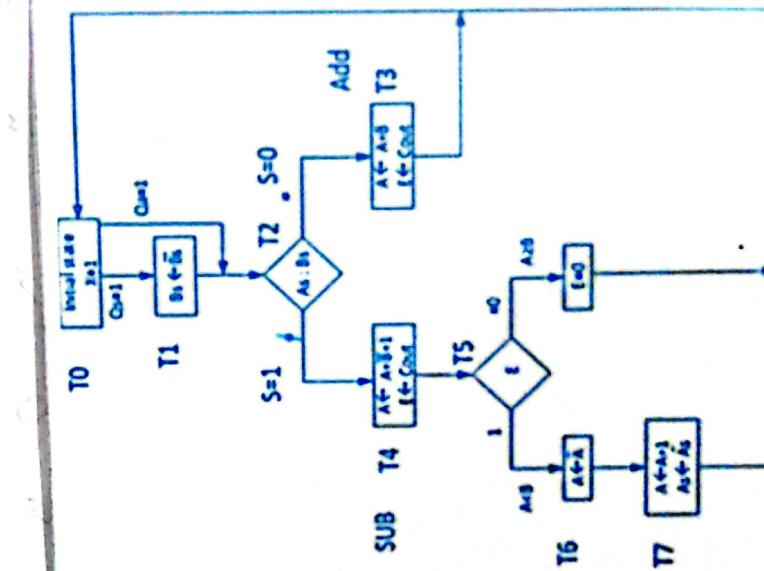
T_3 : $A \leftarrow A + B, E \leftarrow C_{out}$

T_4 : $A \leftarrow A + \bar{B} + 1, E \leftarrow C_{out}$

T_5 : $E \leftarrow 0$

T_6 : $A \leftarrow \bar{A}$

T_7 : $A \leftarrow A + 1, A_s \leftarrow \bar{A}_s$



Imp.

slide-15

④ Rom Address :-

000
001
011
100
101
110
111

- ④ Rom address $\rightarrow T_0, T_1, T_2, \dots$ এগৈতে সাথে হবে 'rom address' এর instruction \rightarrow যদি "go to 4" হলে তা $\rightarrow T_1$ এর rom address \rightarrow কোন address এর next address \rightarrow হবে। যদি "go to 20" হলে তা \rightarrow তা immediate address value \rightarrow next address \rightarrow হবে।
- ④ Select address \rightarrow value \rightarrow rom address রয়ে রয়ে কিন্তু গণ্য $S=1$
- ④ ~~select~~ \rightarrow এবত্বে স্মার্ট ডিপ্লায় select address \rightarrow 10 111 ৩ E21

ROM address	ROM outputs												Address		Select	
	x	s ₂	s ₁	s ₀	C _{in}	L	y	z	w	10	11	12	13	14		
000	0	0	0	0	0	0	0	0	0	0	0	0	0	1		
001	0	0	1	0	0	0	0	1	0	0	1	0	0	1		
011	0	1	0	0	0	0	0	0	0	1	0	0	0	1		
100	0	1	1	0	0	1	0	0	0	0	0	1	0	1		
101	1	0	0	0	1	0	0	0	1	0	0	1	0	1		
110	1	0	1	0	1	0	0	1	0	1	1	1	1	0		
111	1	1	0	0	1	0	1	0	0	0	0	0	0	0		

Slide - 24

④ यदि R_1 और R_2 दोनों प्रत्येक $R_1 \leftarrow R_1 + 1$ के बाद, R_2 तक समान हो जाए तो $R_2 = R_1$ मान दें।

Ex:

$$R_1 = 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1$$

$$\therefore R_2 (\text{if } R_2 = 0) = 9$$

$$R_2 (\text{if } R_2 = 1) = 5$$

* R_1 (Binary) के बाद carry वाली ओर R_2 (Decimal) - 1 1 का count हो।

* flowchart की रूपात एवं start पर इसे घटकों address देखा, then $R_2(D) = 0$ बाटे हुए, तो $C = 0$ करते हुए, एवं Diamond shape फिर यदि $R = 0$ तो Done or else $\neq 0$ होने आवश्यक Continue हो। $\neq 0$ होने R_1 के carry दिये करते हुए, यदि carry = 0 तो अब उसे आवश्यक R_1 दिये continue करते हुए तो $C = 1$ होने $R_2 + 1 \rightarrow R_2$ दिये तो यदि carry = 0 -> continue हो।

Imp.

R1 (Binary)	C	R2 (D)
0 0 1 1 0 1 0 0	0	0
0 0 0 0 1 1 0 1	1	1
0 0 0 0 0 1 1 0	1	2
0 0 0 0 0 0 1 1	0	2
0 0 0 0 0 0 0 1	1	3
0 0 0 0 0 0 0 0	1	4

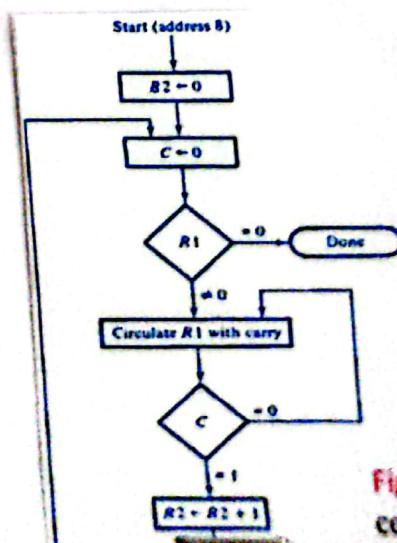


Fig.
cou

256 128 64 32 16 8 4 2 1

8.

Prepare a flow chart that will count the number of 0's in register, R2, and then store the counts in register R5.
 Determine the outputs of the R5 (in binary) and R2 (in decimal) registers as well as of the carry flag after each clock cycle or timing state. Determine the number of states that are required to complete the operation.

Timing States	R2								C	R5 (Decimal)	R5 (Binary)
T1	1	1	0	1	0	0	1	0	1	1	00000001
T2	1	1	1	0	1	0	0	1	0	2	00000010
T3	1	1	1	1	0	1	0	0	1	2	00000010
T4	1	1	1	1	1	0	1	0	0	3	00000011
T5	1	1	1	1	1	1	0	1	0	4	00000100
T6	1	1	1	1	1	1	1	0	1	4	00000100
T7	1	1	1	1	1	1	1	1	0	5	00000101
T8	1	1	1	1	1	1	1	1	1	5	00000101
											00000101

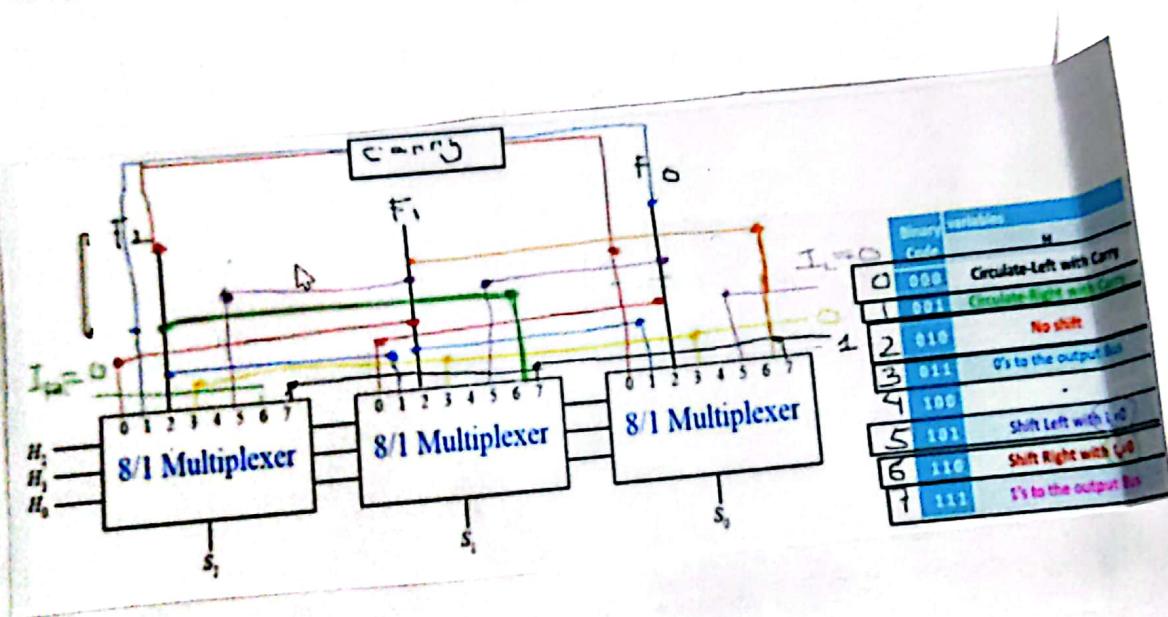
use multiple
sites

status register
3 bit shifter
control word

Lecture - 4

shifter

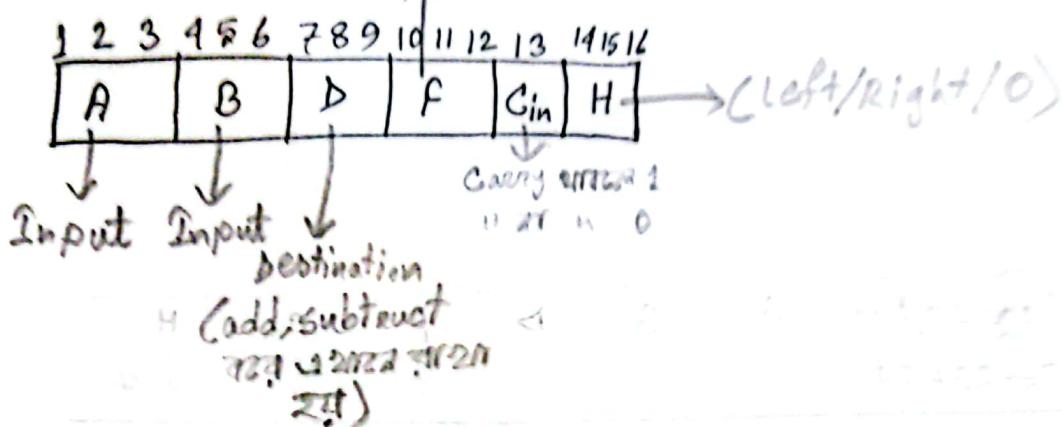
- ④ 3 bit shifter design for the shifting operations listed in the following Table.



- * No shift register always কার্য হবে। f ধারণা denote কার্য হবে।
 - * Circulate করা হবেন Pin to f করার হবে।
 - * Shift করা হবেন f to Pin করার হবে।
 - * 6f Multiplexer এর পরে একটি "carry" ফিল্ড হবে।

* Develop a control word from the table :-

control word, Function
(Input -> carry check)



Write a 16-bit control word for a micro-operation of adding two numbers stored in the registers R1 and R2 with carry and then storing the results in the R5 register after circulating it to the right with no external data.

Answer:

Microoperation:

$R5 \leftarrow R1 + R2 + Cin$

Control Word Format:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 odc

001 010 101 001 1 101

0 0 1 R1 R2 R3 R4 R5 R6 R7

0 1 0 R2 R3 R4 R5 R6 R7

0 1 1 R3 R4 R5 R6 R7

1 0 0 R4 R4 R4 R5 R6 R7

In Binary form: 001 010 101 0011 101b

001 010 101 0011 101b

In Hexadecimal form: 2A9Dh;

2A9Dh;

TABLE 9-8 Functions of control variables for the processor of Fig. 9-16

	Function of selection variables					
	A	B	D	F with $C_{in} = 0$	F with $C_{in} = 1$	H
001 010 101 001 1 101	Input data	Input data	None	$A, C \leftarrow 0$	$A + 1$	No shift
	R_1	R_1	R_1	$A + B$	$A + B + 1$	Shift-right, $J_R = 0$
	R_2	R_2	R_2	$A - B - 1$	$A - B$	Shift-left, $J_L = 0$
	R_3	R_3	R_3	$A - 1$	$A, C \leftarrow 1$	0's to output bus
	R_4	R_4	R_4	$A \vee B$	—	—
	R_5	R_5	R_5	$A \oplus B$	—	Circulate-right with C
	R_6	R_6	R_6	$A \wedge B$	—	Circulate-left with C
	R_7	R_7	R_7	\bar{A}	—	—

TABLE 9-9 Examples of microoperations for processor

Microoperation	Control word						Function
	A	B	D	F	C_{in}	H	
$R1 \leftarrow R1 - R2$	001	010	001	010	1	000	Subtract R2 from R1
$R3 \leftarrow R4$	011	100	000	010	1	000	Compare R3 and R4
$R5 \leftarrow R4$	100	000	101	000	0	000	Transfer R4 to R5
$R6 \leftarrow \text{Input}$	000	000	110	000	0	000	Input data to R6
$\text{Output} \leftarrow R7$	111	000	000	000	0	000	Output data from R7
$R1 \leftarrow R1, C \leftarrow 0$	001	000	001	000	0	000	Clear carry bit C
$R3 \leftarrow \text{shl } R3$	011	011	011	100	0	010	Shift-left R3 with $J_L = 0$
$R1 \leftarrow \text{crc } R1$	001	001	001	100	0	101	Circulate-right R1 with carry
$R2 \leftarrow 0$	000	000	010	000	0	011	Clear R2

bit Control word

* Shift, Circulate or clear input ~~and~~ H এর Value কৈবল্য,

011 000 011 0

5. Develop the control words in binary and hexadecimal formats using the information provided in Table 2 for the following micro-operations:

- $R7 \leftarrow R3 + R4$
- $R3 \leftarrow SHR R3$
- $RS \leftarrow R1$
- $R2 \leftarrow SHR R5$
- $R3 \leftarrow CRC R7$

Table 2: Functions of control variables

Binary Code	Functions of selection variables					
	A	B	D	F with $C_{in} = 0$	F with $C_{in} = 1$	H
000	Input Data	Input Data	None	A+1	A	1's to the output Bus
001	R1	R1	R1	A+B	A+B+1	Shift Left with $I_L = 0$
010	R2	R2	R2	A-B-1	A-B	No Shift
011	R3	R3	R3	A	A+1	Circulate Left with Carry
100	R4	R4	R4	A	X	0's to the output Bus
101	R5	R5	R5	ANOR B	X	-
110	R6	R6	R6	AAND B	X	Circulate Right with Carry
111	R7	R7	R7	AORB	X	Shift Right with $I_R = 0$

Microoperation	A	B	D	F	C_{in}	H
$R7 \leftarrow R3 + R4$	011	100	111	001	0	000
$R3 \leftarrow SHL R3$	011	011	011	011	0	001
$RS \leftarrow R1$	001	000	101	011	0	000
$R2 \leftarrow SHR R5$	101	101	010	111	0	111
$R3 \leftarrow CRC R7$	111	111	011	111	0	110

*CRC, CLC or SHL instead AORB 221 F-21

④ $C_{in} = 0$ Priority*. $C_{in} = 0$ even if $C_{in} = 1$.