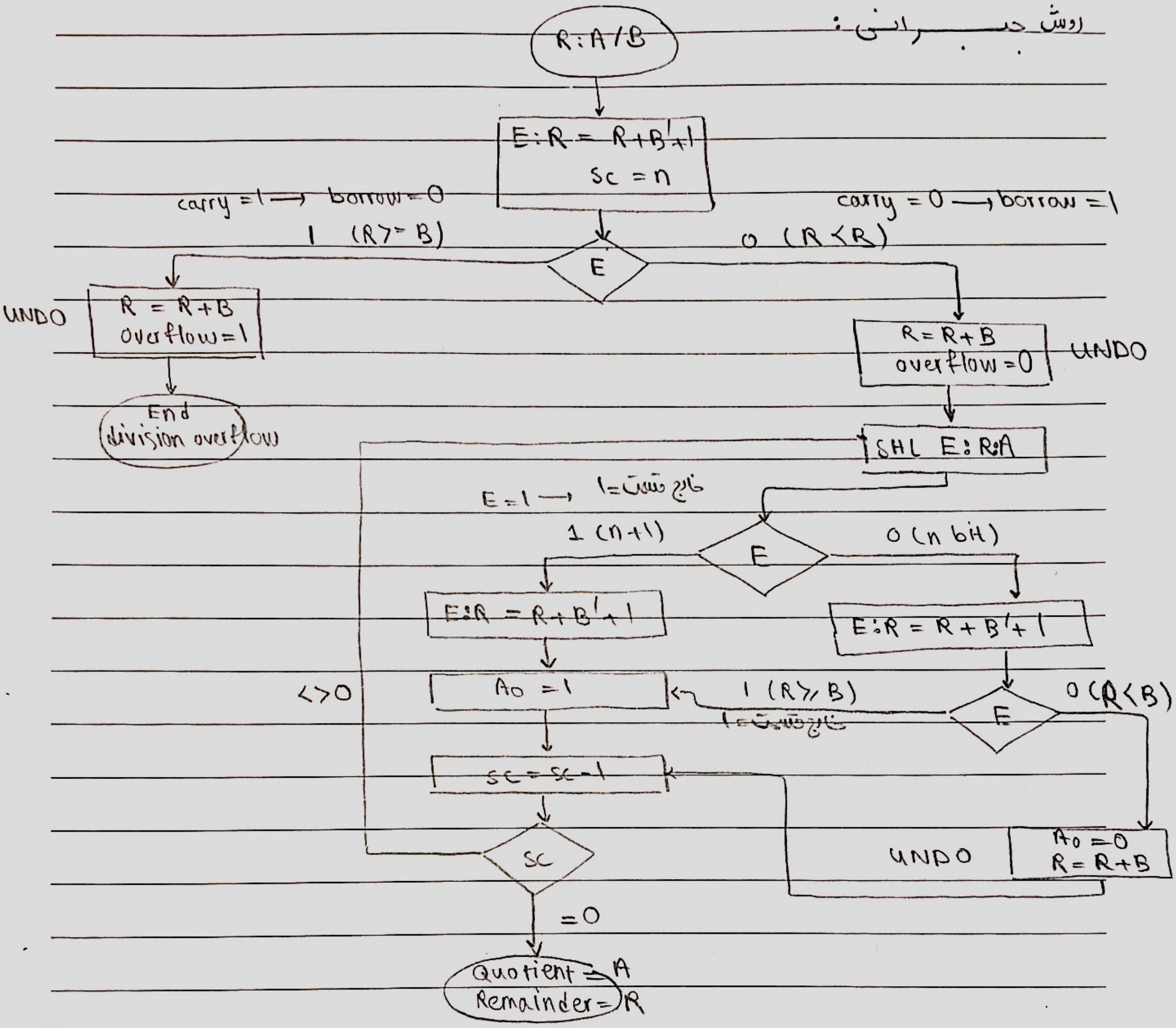
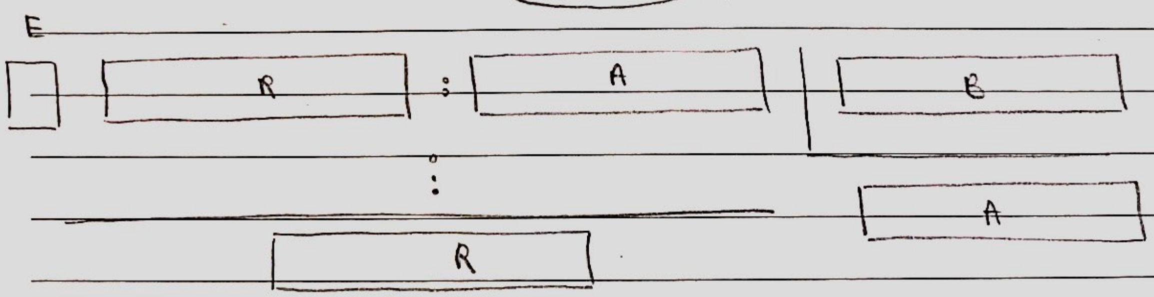
مینا بیکی - کد: ۹۸۳۱۰۷۵ - استاد: دکتر زرندی

Subject: تمرین سری هشتم درس معماری کامپیوتر

۱. تقسیم ۲۷ بر ۵ را به دو روش جبرانی و غیرجبرانی انجام دهید و مراحل انجام کار را گام به گام توضیح دهید و فلوچارت هر دو روش را رسم کنید.

روش جبرانی:





Subject:

E	R	A	B
0	0 1 1	0 1 1	1 0 1

$27 \xrightarrow{\text{binary}} = 11011$ $n = 3$ bits تقسیم به روش جبرانی :

$5 \longrightarrow = 101$

check overflow:

E	R	A	B
0	0 1 1	0 1 1	1 0 1

$$\begin{array}{r} 011 \\ +\ 010 \\ 1 \\ \hline 110 \end{array}$$

$E:R = R+B'+1 \longrightarrow$

E	R
0	1 1 0

$E=0 \longrightarrow carry=0 \longrightarrow borrow=1 \longrightarrow R<B \longrightarrow$ overflow نداریم و تقسیم قابل انجام است. ✓

$UNDO \longrightarrow R = R+B \longrightarrow$ 0 1 1

$$\begin{array}{r} 110 \\ +\ 101 \\ \hline 1011 \end{array}$$

$\boxed{n=3}$:

SHL $E:R:A \longrightarrow$

E	R	A
0	1 1 0	1 1 0

$E=0 \longrightarrow$ باید برای پیدا کردن خارج قسمت مقایسه انجام شود $\longrightarrow E:R = R+B'+1$

$$\begin{array}{r} 110 \\ +\ 010 \\ 1 \\ \hline 1\ 001 \end{array}$$

E	R
1	0 0 1

$\longrightarrow E=1 \longrightarrow A_0=1 \longrightarrow$

E	R	A
1	0 0 1	1 1 1

$\boxed{n=2}$:

SHL $E:R:A \longrightarrow$

E	R	A
0	0 1 1	1 1 0

$E=0 \longrightarrow$ باید برای پیدا کردن خارج قسمت مقایسه انجام شود $\longrightarrow E:R = R+B'+1$

$$\begin{array}{r} 011 \\ +\ 010 \\ 1 \\ \hline 0\ 110 \end{array}$$

E	R
0	1 1 0

$\longrightarrow E=0 \xrightarrow{R<B}$ UNDO: $R=R+B$, $A_0=0$

E	R	A
0	0 1 1	1 1 0

$$\begin{array}{r} 110 \\ +\ 101 \\ \hline 1011 \end{array}$$

$\boxed{n=1}$:

SHL $E:R:A \longrightarrow$

E	R	A
0	1 1 1	1 0 0

$E=0 \longrightarrow$ باید برای پیدا کردن خارج قسمت مقایسه انجام شود $\longrightarrow E:R = R+B'+1$

$$\begin{array}{r} 111 \\ +\ 010 \\ 1 \\ \hline 1\ 010 \end{array}$$

E	R
1	0 1 0

$\longrightarrow E=1 \xrightarrow{R \geq B} A_0 = 1$

E	R	A
0	0 1 0	1 0 1

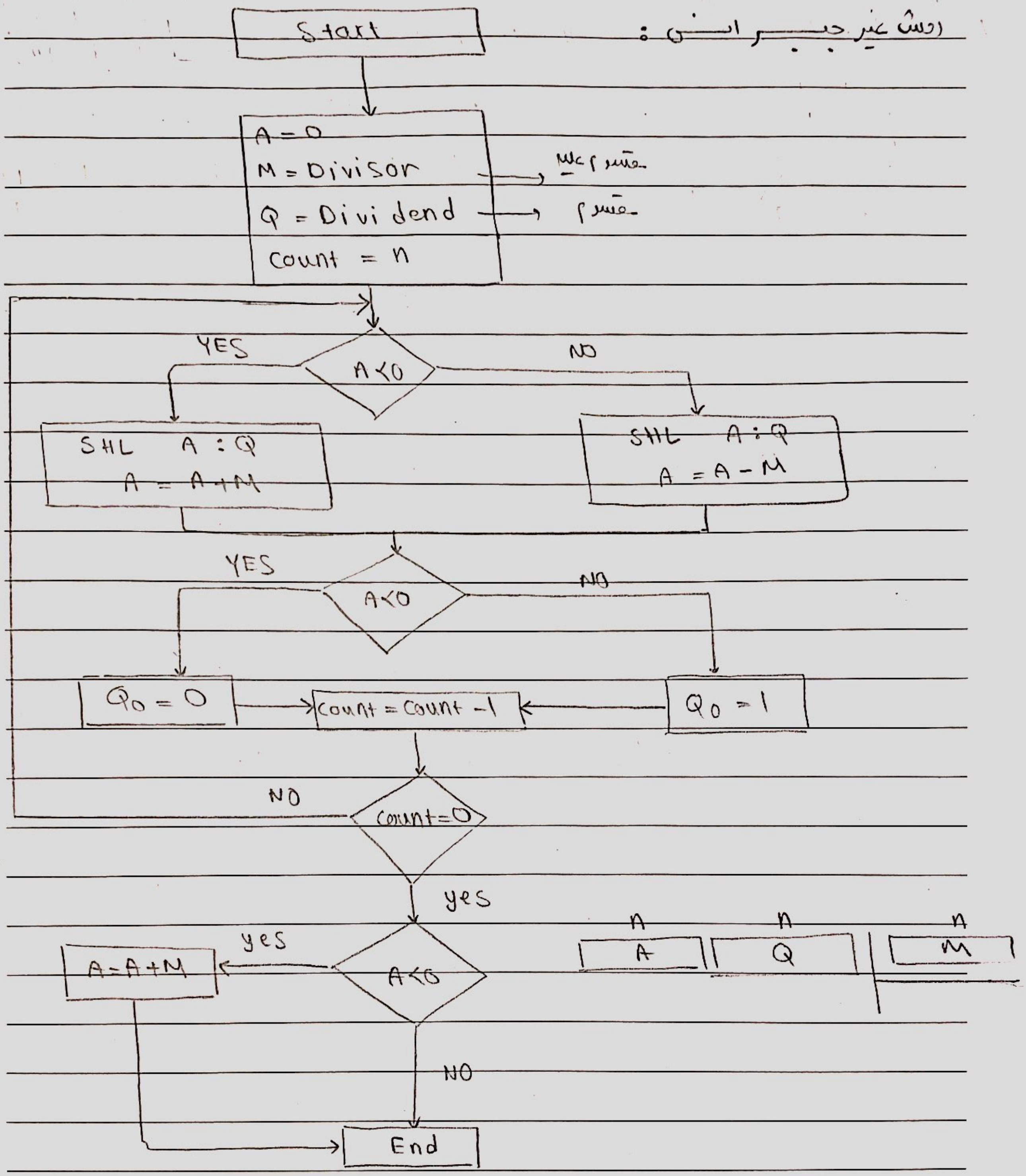
IDEA

$\Longrightarrow$ ✓ $101 = A =$ خارج قسمت

✓ $010 = R =$ باقی مانده

Subject:

روش غیر جبرانی :



(۲۷)$_{۱۰}$ = (۱۱۰۱۱)$_{۲}$

(۵)$_{۱۰}$ = (۱۰۱)$_{۲}$

Subject:

انجام تقسیم به روش غیرجبرانی :

A	Q	M
0 0 0 0 0	1 1 0 1 1	0 0 1 0 1

$\boxed{n=5}$: $A=0 \longrightarrow \cancel{A<0} \longrightarrow$ SHL A:Q , $A = A-M$

A	Q
0 0 0 0 1	1 0 1 1 0

$\longrightarrow A = A-M = A+M'+1$

	0 0 0 0 1
+	1 1 0 1 0
	1
	1 1 1 0 0

A	Q
1 1 1 0 0	1 0 1 1 0

$\longrightarrow$ sign bit of $A=1 \longrightarrow A<0 \longrightarrow Q_0 = 0$ (.است 'n) $\longrightarrow n = 5-1 = 4$

$\boxed{n=4}$: $A<0 \longrightarrow$ SHL A:Q , $A = A+M$

A	Q
1 1 0 0 1	0 1 1 0 0
1 1 1 1 0	0 1 1 0 0

	1 1 0 0 1
+	0 0 1 0 1
	1 1 1 1 0

$A<0 \longrightarrow Q_0 = 0$ ✓ $\longrightarrow n = 3$

$\boxed{n=3}$: $A<0 \longrightarrow$ SHL A:Q , $A = A+M$

A	Q
1 1 1 0 0	1 1 0 0 0
0 0 0 0 1	1 1 0 0 $\cancel{0}$ 1

	1 1 1 0 0
+	0 0 1 0 1
	1 0 0 0 0 1

$\longrightarrow \cancel{A<0} \longrightarrow Q_0 = 1$

$\boxed{n=2}$: $\cancel{A<0} \longrightarrow$ SHL A:Q , $A = A-M$

A	Q
0 0 0 1 1	1 0 0 1 0
1 1 1 1 0	1 0 0 1 0

	0 0 0 1 1
+	1 1 0 1 0
	1
	1 1 1 1 0

$A<0 \longrightarrow Q_0 = 0$ ✓

$\boxed{n=1}$: $A<0 \longrightarrow$ SHL A:Q , $A = A+M$

A	Q
1 1 1 0 1	0 0 1 0 0
0 0 0 1 0	0 0 1 0 $\cancel{0}$ 1

	1 1 1 0 1
+	0 0 1 0 1
	1 0 0 0 1 0

$\longrightarrow \cancel{A<0} \longrightarrow Q_0 = 1 \longrightarrow$ Count = 0 $\longrightarrow \cancel{A<0}$ end

Subject:

$\Longrightarrow$ Remainder $= A = (00010)_2 = (۲)_{10}$ ✓

Quotient $= Q = (00101)_2 = (۵)_{10}$ ✓

۲. تأخیر حاصل در ضرب کننده آرایه‌ای n در n را حساب کنید و مراحل کار را ذکر کنید. تأخیر همه گیت‌ها باهم برابر و مساوی d نانوثانیه است.

منطق عملکرد ضرب آرایه‌ای : (مثال برای دو عدد ۴ بیتی)

$$\begin{array}{ccccc} & x_3 & x_2 & x_1 & x_0 \\ \times & y_3 & y_2 & y_1 & y_0 \end{array}$$

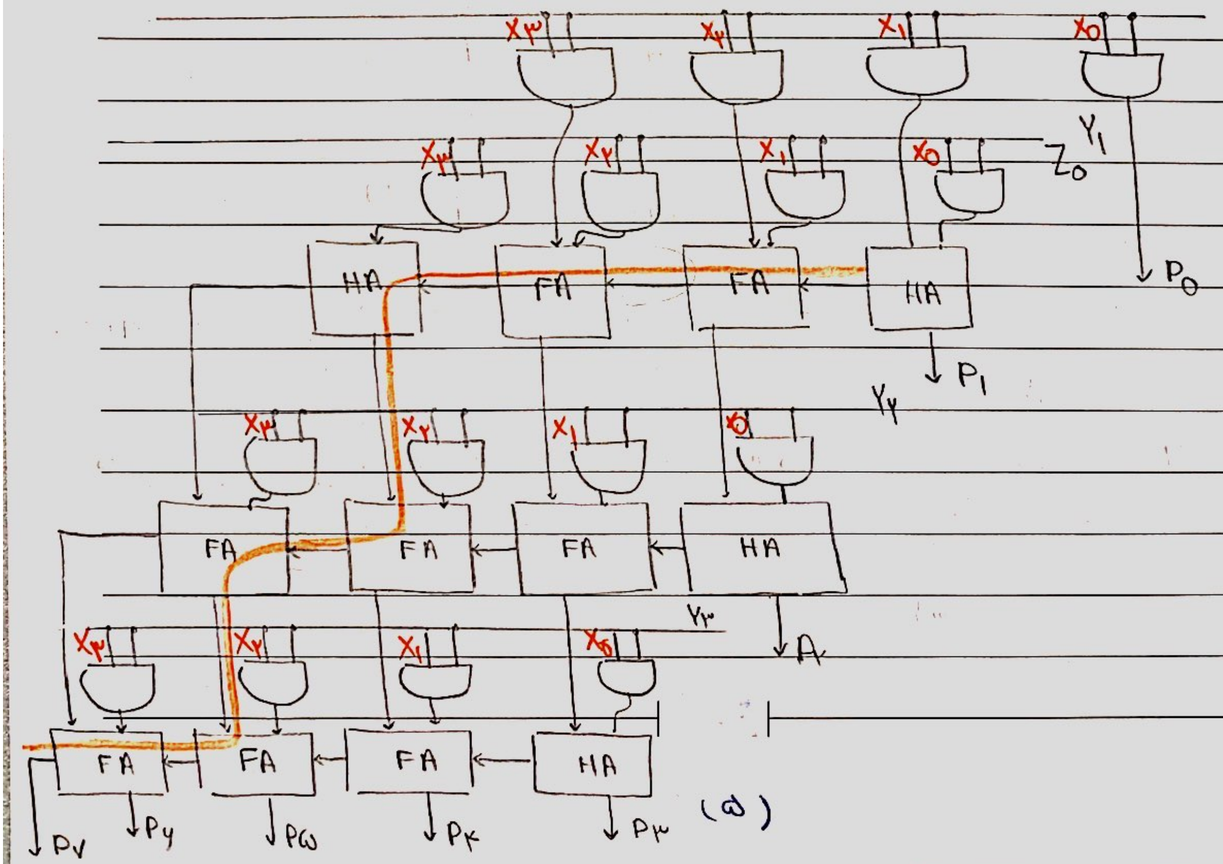
				$x_3 \cdot y_0$	$x_2 \cdot y_0$	$x_1 \cdot y_0$	$x_0 \cdot y_0$
			$x_3 y_1$	$x_2 y_1$	$x_1 y_1$	$x_0 \cdot y_1$	
		$x_3 y_2$	$x_2 y_2$	$x_1 y_2$	$x_0 y_2$		
	$x_3 y_3$	$x_2 y_3$	$x_1 y_3$	$x_0 y_3$			
P_7	P_6	P_5	P_4	P_3	P_2	P_1	P_0

+ ← Partial Products

منطق ضرب دقیقاً مانند نحوه‌ی ضرب‌های عادی باشد.

به این معنا که در مرحله i ام، عنصر ۰ ام از عدد دوم را در عنصر ۰ ام از عوامل ضرب می‌کنیم، سپس عنصر ۰ ام از عدد دوم را در عنصر اول عوامل ضرب می‌کنیم به همین ترتیب ادامه می‌دهیم تا partial product ها کامل شود

برای سخت افزار ۲ حالت می‌توانیم داشته باشیم. می‌توانیم از HA و FA استفاده کنیم و یا فقط از FA ها استفاده کنیم. اگر از HA ها هم استفاده کنیم:



Subject:

با توجه به شکل بالا برای ضرب 4bit x 4bit می توان گفت که برای ضرب n bit x n bit :

n-1 ردیف جمع کننده داریم که در هر کدام n تا جمع کننده وجود دارد. (ترکیبی از HA ها و FA ها)

برای به دست آوردن Delay، مسیر بحرانی را در نظر می گیریم، مسیری که با رنگ نارنجی در شکل نشان داده شده است.

محاسبه تاخیر: ① برای گیت های and که به طور موازی کار می کنند: d

② تاخیرهای مسیر:

$$t_{carry\,HA} + (n-2)\, t_{carry\,FA} + t_{sum\,HA} + \underbrace{(n-2)}_{\text{تعداد خطوط میانی}} (t_{carry\,FA} + t_{sum\,FA})$$

$$+ \underbrace{2 t_{carry\,FA}}_{\text{خط آخر}}$$

$$= t_{carry\,FA}(\underbrace{n-2+n-3+2}_{2n-3}) + t_{carry\,HA} + t_{sum\,HA} + (n-3)\, t_{sum\,FA}$$

$$= 2d(2n-3) + d + d + d(n-3) = d(\underbrace{4n-6+2+n-3}_{5n-7}) = (5n-7)d$$

$$\Rightarrow T = \underset{①}{d} + \underset{②}{(5n-7)d} = \boxed{(5n-6)d}$$

حالت دوم: اگر همه ی جمع کننده ها را FA فرض کنیم:

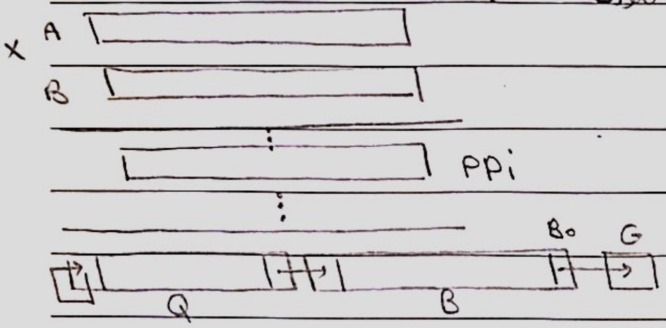
$$T = d + (\underbrace{2d(2n-3) + 2d + d + (n-3)d}_{(4n-6+3+n-3)d \,=\, 5n}) = (5n-5)d$$

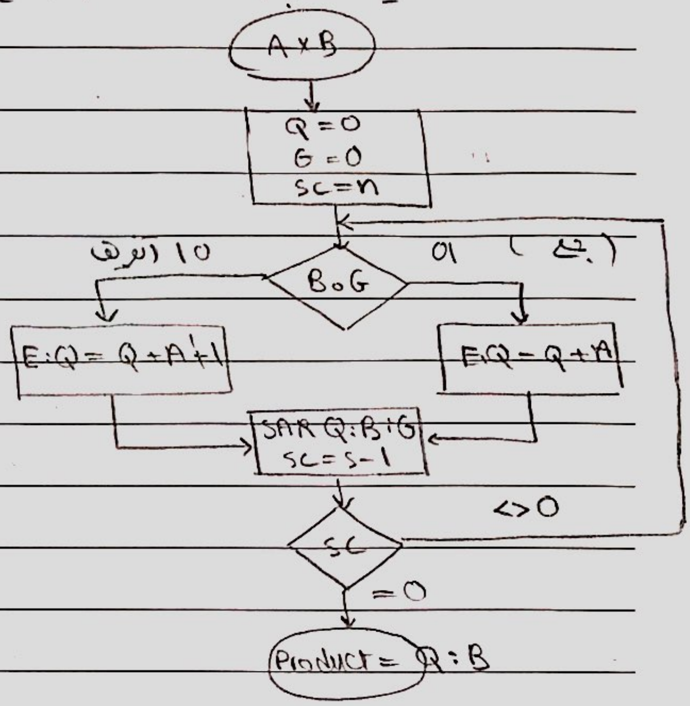
IDEA

Subject:

۳. در یک ضرب بوث ۳۲ بیت در ۳۲ بیت، حداکثر تعداد جمع و همچنین حداکثر تعداد تفریق را که ممکن است رخ دهد را به دست آورید و توضیح دهید چگونه به جواب رسیدید.

همانطور که می دانیم در ضرب بوث عملکرد ضرب به این صورت است:





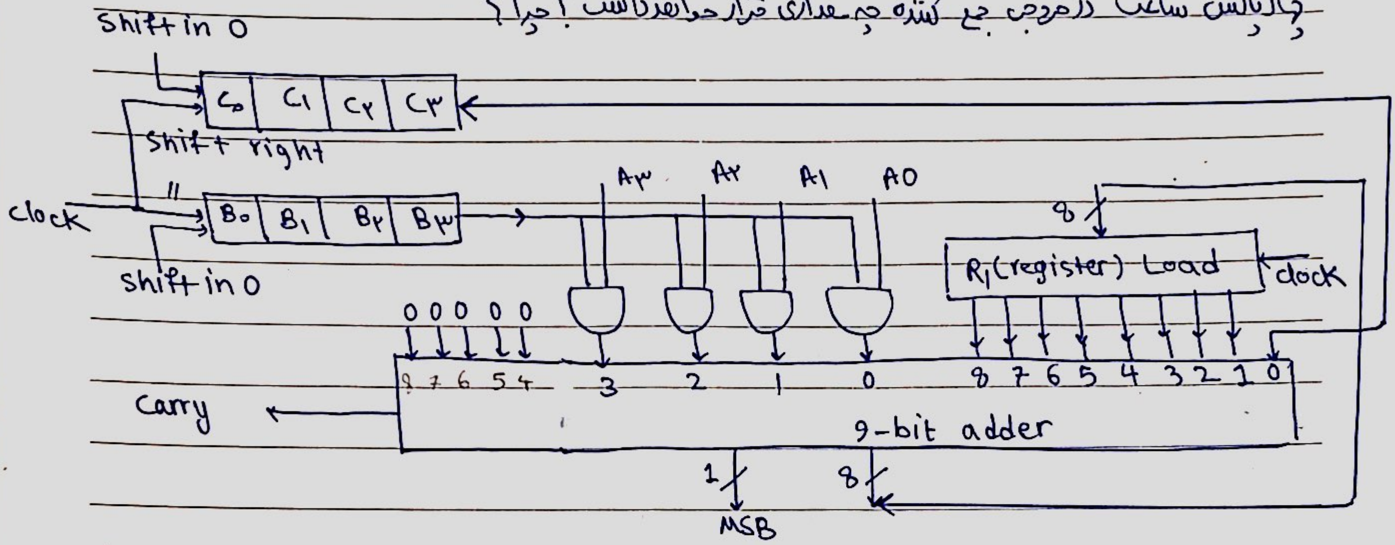
پس هر بار به BoG نگاه می کنیم

پس از آن جایی که در هر مرحله به BoG نگاه می کنیم و اگر 01 باشد عمل جمع و اگر 10 باشد تفریق انجام می دهیم، در اولین بار که $G=0$ است پس $B_0=1$ می گذاریم تا $B_0G=10$ و عمل تفریق انجام شود. سپس اگر $B_1=0$ باشد پس از SAR، $B_0=0$ می شود و $B_0G=01$ و عمل جمع انجام می شود به همین ترتیب ادامه می دهیم. چون یکی در میان عمل جمع و تفریق انجام می شود پس هر کدام به تعداد حداکثر خود می رسند

⟸ حداکثر تعداد تفریق = حداکثر تعداد جمع $= \frac{۳۲}{۲} = \boxed{۱۶}$

Subject:

۴. در شکل زیر سه عدد چهار بیتی $A_3A_2A_1A_0$ و $B_3B_2B_1B_0$ و $C_3C_2C_1C_0$ موجود است و مقدار اولیه ثبات R_1 برابر با صفر است. محتوای اولیه شیفت دهنده را نیز داده ایم. بعد از گذشت چهار پالس ساعت در خروجی جمع کننده چه مقداری قرار خواهد داشت؟ چرا؟



* Clock 1 : first we shift B & C : $\underline{0}\ \underline{c_0}\ \underline{c_1}\ \underline{c_2} \longrightarrow c_3$

$\underline{0}\ \underline{b_0}\ \underline{b_1}\ \underline{b_2} \longrightarrow b_3$

$\longrightarrow$ 9 bit adder

$$\begin{array}{r} 00000\ \ b_3a_3\ \ b_3a_2\ \ b_3a_1\ \ b_3a_0 \\ +\ 00000\ \ 0\ \ 0\ \ 0\ \ c_3 \\ \hline 0\ \underbrace{0000\ \ b_3a_3\ \ b_3a_2\ \ b_3a_1\ \ b_3a_0+c_3}_{\text{new } R_1} \end{array}$$

* Clock 2 :

shift B & C : $\underline{0}\ \underline{0}\ \underline{c_0}\ \underline{c_1} \longrightarrow c_2$

$\underline{0}\ \underline{0}\ \underline{b_0}\ \underline{b_1} \longrightarrow b_2$

$\longrightarrow$ 9 bit adder :

$$\begin{array}{r} 0000\ \ 0\ \ b_2a_3\ \ b_2a_2\ \ b_2a_1\ \ b_2a_0 \\ +\ 0000\ \ b_3a_3\ \ b_3a_2\ \ b_3a_1\ \ b_3a_0+c_3\ \ c_2 \\ \hline \underbrace{0000\ \ b_3a_3\ \ \underset{b_2a_3+b_3a_2}{\wedge}\ \ \underset{b_3a_1+b_2a_2}{\wedge}\ \ b_2a_1+b_3a_0+c_3\ \ c_2+a_0b_2}_{\text{new } R_1} \end{array}$$

Subject:

* Clock 3 : shift B & C :

0	0	0	c_0	$\rightarrow c_1$
0	0	0	b_0	$\rightarrow b_1$

$\rightarrow$ 9 bit adder :

0	0	0	0	0	b_1a_3	b_1a_2	b_1a_1	b_1a_0
+ 0	0	0	a_3b_3	$a_2b_3+a_3b_2$	$a_1b_3+a_2b_2$	$c_3+a_0b_3+a_1b_2$	$c_2+a_0b_2$	c_1
0	0	0	a_3b_3	$a_2b_3+a_3b_2$	$a_1b_3+a_2b_2+a_3b_1$	$c_3+a_0b_3+a_1b_2+a_2b_1$	$c_2+a_0b_2+a_1b_1$	$c_1+a_0b_1$

new R_1

* Clock 4 : shift B & C :

0	0	0	0	$\rightarrow c_0$
0	0	0	0	$\rightarrow b_0$

$\rightarrow$ ~~9 bit adder:~~

0	0	0	0	0	b_0a_3	b_0a_2	b_0a_1	b_0a_0
+ 0	0	a_3b_3	$a_2b_3+a_3b_2$	$a_1b_3+a_2b_2+a_3b_1$	$c_3+a_0b_3+a_1b_2+a_2b_1$	$c_2+a_0b_2+a_1b_1$	$c_1+a_0b_1$	c_0
0	0	a_3b_3	$a_2b_3+a_3b_2$	$a_1b_3+a_2b_2+a_3b_1$	$c_3+a_0b_3+a_1b_2+a_2b_1+a_3b_0$	$c_2+a_0b_2+a_1b_1+b_0a_2$	$c_1+a_0b_1+a_1b_0$	$c_0+a_0b_0$

$\boxed{\text{result} = (a \times b) + c}$

با توجه به عبارت می‌توان متوجه شد که حاصل شده حاصل ضرب آرایه‌های a و b است که به بیت اول c نیز اضافه شده است.

Scanned with CamScanner