

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

راهنمای جامع مسئله های معماری کامپیوتر

موریس مانو

به پیوست مجموعه تست های برگزیده فصل بندی شده و پاسخ تشریحی

تالیف و تدوین :

محمدجواد مروتی

سعید کاظمی

تقدیم به ...

تقدیم به ساحت مقدس حضرت ولی عصر (عج) و همه کسانی که به رسم ادب این
اوراق را تورق می کنند.
گرچه از آن مستغنی اند

مروتی

تقدیم به خانواده ام

و هر آنکه لبخند با لبش عقد بسته.

کاظمی

فهرست مطالب

۵	حمایت داوطلبانه (Donation)
۶	پیشگفتار نویسندگان
۸	فصل اول : مدارهای منطقی دیجیتال
۲۶	فصل دوم: قطعات دیجیتال
۴۱	فصل سوم: نمایش داده‌ها
۵۶	فصل چهارم: انتقال ثبات‌ها و ریزعمل‌ها
۷۴	فصل پنجم: سازمان و طراحی یک کامپیوتر پایه
۹۵	فصل ششم: برنامه‌نویسی کامپیوتر پایه
۱۲۰	فصل هفتم: کنترل ریزبرنامه‌نویسی شده
۱۳۶	فصل هشتم: واحد مرکزی پردازش
۱۶۲	فصل نهم: پردازش خط لوله‌ای و برداری
۱۷۳	فصل دهم: معماری کامپیوتر و الگوریتم‌های حسابی
۲۱۵	فصل یازدهم: سازمان ورودی- خروجی
۲۳۵	فصل دوازدهم: سازمان حافظه
۲۵۶	تست‌های فصل ۴ «انتقال ثبات‌ها و ریزعمل‌ها»
۲۵۸	حل تست‌های فصل ۴
۲۵۹	تست‌های فصل ۵ «سازمان و طراحی یک کامپیوتر پایه»
۲۶۲	حل تست‌های فصل ۵
۲۶۴	تست‌های فصل ۶ «برنامه‌نویس کامپیوتر پایه»
۲۶۶	حل تست‌های فصل ۶
۲۶۷	تست‌های فصل ۷ «کنترل ریز برنامه‌نویسی شده»

۲۶۹	حل تست‌های فصل ۷
۲۷۰	تست‌های فصل ۸ «واحد مرکزی پردازش»
۲۷۳	حل تست‌های فصل ۸
۲۷۵	تست‌های فصل ۹ «پردازش خط لوله‌ای و برداری»
۲۷۷	حل تست‌های فصل ۹
۲۷۹	تست‌های فصل ۱۰ «معماری کامپیوتر و الگوریتم‌های حسابی»
۲۸۲	حل تست‌های فصل ۱۰
۲۸۴	تست‌های فصل ۱۱ «سازمان ورودی - خروجی»
۲۸۶	حل تست‌های فصل ۱۱
۲۸۷	تست‌های فصل ۱۲ «سازمان حافظه»
۲۸۹	حل تست‌های فصل ۱۲
۲۹۱	منابع

حمایت داوطلبانه (Donation)

خواننده گرامی

تالیف، تدوین، آماده سازی محتوا، تایپ، صفحه آرایی و ویرایش‌های
انجام شده کتاب حاضر، با هزینه نویسندگان انجام شده است.
در صورت تمایل به حمایت از این مجموعه، پیوند یا کد QR زیر را دنبال
کنید.

<http://hpcc.ir/downloads/computer-architecture-solution/>



پیشگفتار نویسندگان

- درس معماری کامپیوتر به عنوان یک مبحث پایه‌ای رشته مهندسی و علوم کامپیوتر و سایر رشته‌های مرتبط مطرح است. به طوری که روند طراحی و ساخت سیستم‌های کامپیوتری را از ترکیب مباحث نرم‌افزاری و برنامه‌نویسی، و مباحث سخت‌افزاری، مدارهای منطقی و دیجیتال بازگو می‌نماید.
- کتاب حاضر به بررسی و حل مسائل درس معماری کامپیوتر، براساس کتاب معماری کامپیوتر «پروفسور موریس مانو»، ترجمه‌ی «دکتر قدرت سپیدنام»، می‌پردازد. در ضمن، سؤالات چهار گزینه‌ای برگزیده فصل‌های کتاب معماری کامپیوتر به همراه پاسخ تشریحی آنها نیز به کتاب پیوست شده است.
- سه فصل ابتدایی کتاب معماری کامپیوتر موریس مانو، مشتمل بر مباحث مدارهای منطقی و دیجیتال است که کتاب حاضر به تشریح مسائل این سه فصل نیز پرداخته است. بنابراین این کتاب می‌تواند برای مرور و تسلط بر مباحث مدارهای منطقی و دیجیتال به عنوان مقدمه و پیش‌نیاز طراحی و معماری کامپیوتر استفاده گردد.
- منبع اصلی مجموعه حاضر، جزوه راهنمای حل مسئله (Solution Manual) پروفسور موریس مانو است. اگرچه در بسیاری از مسائل و راه‌حل‌ها، این راهنما دارای نقص، اشتباه‌های محاسباتی و اشتباه‌های منطقی بوده و یا اینکه تنها دارای جواب آخر بوده‌اند؛ که در کتاب حاضر اقدام به رفع این اشتباه‌ها شده و توضیحات لازم در جواب‌ها داده شده و الگوریتم‌ها، روش‌های حل و شکل‌های ابتکاری بکار برده‌ایم. البته امیدواریم خوانندگان محترم با مطالعه دقیق و کامل کتاب، نویسندگان این مجموعه را نسبت به نقص‌ها، کمبودها، اشتباه‌های احتمالی و نامفهوم بودن مطالب آگاه نمایند.

با تشکر و آرزوی موفقیت، مروتی - کاظمی

فصل اول : مدارهای منطقی دیجیتال

۱-۱ با استفاده از جدول درستی، صحت قضیه دمورگان را برای سه متغیر

بررسی کنید.

$$(ABC)' = A' + B' + C'$$

حل: طبق جدول درستی برای عبارت فوق داریم:

A	B	C	A.B.C	(A.B.C)'	A'	B'	C'	A'+B'+C'
۰	۰	۰	۰	۱	۱	۱	۱	۱
۰	۰	۱	۰	۱	۱	۱	۰	۱
۰	۱	۰	۰	۱	۱	۰	۱	۱
۰	۱	۱	۰	۱	۱	۰	۰	۱
۱	۰	۰	۰	۱	۰	۱	۱	۱
۱	۰	۱	۰	۱	۰	۱	۰	۱
۱	۱	۰	۰	۱	۰	۰	۱	۱
۱	۱	۱	۱	۰	۰	۰	۰	۰

۱-۲- جدول درستی تابع OR انحصاری (تابع فرد) سه متغیره را بسازید:

$$x = A \oplus B \oplus C$$

حل: ابتدا OR انحصاری A و B را بدست آورده و سپس OR انحصاری حاصل آن

را با C محاسبه می‌کنیم.

$$A \oplus B \oplus C = (A \oplus B) \oplus C$$

A	B	C	$A \oplus B$	$A \oplus B \oplus C$
---	---	---	--------------	-----------------------

۰	۰	۰	۰
۰	۰	۱	۰
۰	۱	۰	۱
۰	۱	۱	۰
۱	۰	۰	۱
۱	۰	۱	۰
۱	۱	۰	۰
۱	۱	۱	۱

۱-۳- عبارات زیر را با جبر بول ساده کنید:

- الف) $A+AB$ ب) $AB+AB'$
 ج) $A'BC+AC$ د) $A'B+ABC'+ABC$

حل:

الف) $A+AB=A(1+B)=A$

ب) $AB+AB'=A(B+B')=A$

ج) $A'BC+AC=C(A'B+A)=C(A'+A)(B+A)=(A+B)C$

د) $A'B+ABC'+ABC=A'B+AB(C'+C)=A'B+AB=B(A'+A)=B$

۱-۴- عبارت های زیر را با جبر بول ساده کنید:

- الف) $AB+A(CD+CD')$ ب) $(BC'+A'D)(AB'+CD')$

حل:

الف) $AB+A(CD+CD')=AB+AC(D+D')=A(B+C)$

ب) $(BC'+A'D)(AB'+CD')=\frac{ABB'C}{0}+\frac{A'AB'D}{0}+\frac{BCC'D'}{0}+\frac{A'CD'D}{0}=0$

۱-۵- با قضیه دمورگان نشان دهید:

- الف) $(A+B)'(A'+B')'=0$ ب) $A+A'B+A'B'=1$

حل:

الف) $(A+B)'(A'+B')' = (A'B')(AB) = 0$

ب) $A + A'B + A'B' = A + A'(B+B') = A + A' = 1$

۱-۶ - با توجه به عبارت بولی $F = x'y + xyz'$

الف) یک عبارت جبری برای متهم، F' بدست آورید.

ب) نشان دهید که $FF' = 0$

ج) نشان دهید $F + F' = 1$

حل:

الف) $F = x'y + xyz'$

$F' = (x+y')(x'+y'+z) = x'y' + xy' + y' + xz + y'z = y'(1+x'+x+z) + xz = y' + x$

ب) $F.F' = (x'y + xyz')(y' + xz) = 0 + 0 + 0 + 0 = 0$

ج) $F + F' = x'y + xyz' + y' + xz = x'y + xy(z' + z) + y'(1 + xz)$

$= x'y + xy + y' = y(x' + x) + y' = y + y' = 1$

۱-۷ - با توجه به تابع بولی $F = xy'z + x'y'z + xyz$

الف) جدول درستی تابع را بدست آورید.

ب) نمودار منطقی را برای عبارت بولی اصلی رسم کنید.

ج) عبارت جبری را با جبر بول ساده کنید.

د) جدول درستی را با عبارت ساده شده تابع تشکیل داده و نشان دهید که

همان جدول بخش (الف) است.

ه) نمودار منطقی را با عبارت ساده شده بدست آورده و تعداد کل گیت‌ها را

با نمودار بخش (ب) مقایسه کنید.

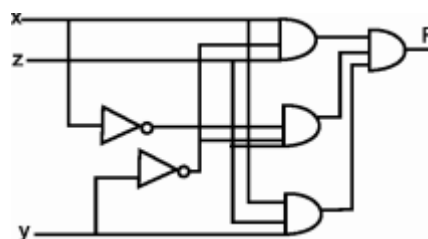
حل:

الف)

ب) $F = xy'z + x'y'z + xyz$

X	y	z	F
---	---	---	---

۰	۰	۰	۰
۰	۰	۱	۱
۰	۱	۰	۰
۰	۱	۱	۰
۱	۰	۰	۰
۱	۰	۱	۱
۱	۱	۰	۰
۱	۱	۱	۱



۶ = تعداد گیت‌های تشکیل دهنده

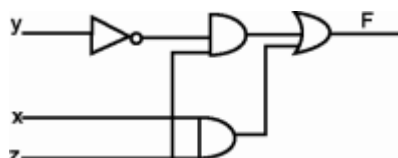
(ج)

$$\begin{aligned}
 F &= xy'z + x'y'z + xyz \\
 &= y'z(x + x') + xz(y + y') \\
 &= y'z + xz
 \end{aligned}$$

(د)

x	y	z	$y'z$	xz	$y'x + xz$
۰	۰	۰	۰	۰	۰
۰	۰	۱	۱	۰	۱
۰	۱	۰	۰	۰	۰
۰	۱	۱	۰	۰	۰
۱	۰	۰	۰	۰	۰
۱	۰	۱	۱	۱	۱
۱	۱	۰	۰	۰	۰
۱	۱	۱	۰	۱	۱

(هـ)



۴ = تعداد گیت‌های تشکیل دهنده

۸-۱ - توابع بولی زیر را با نقشه سه متغیره ساده کنید.

$F(x,y,z)$

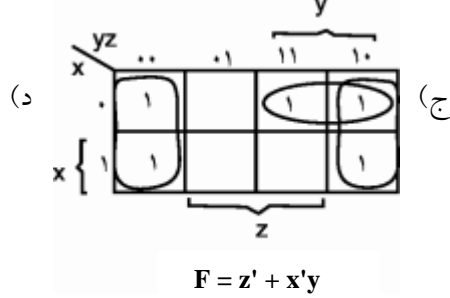
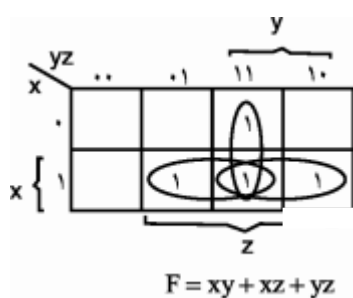
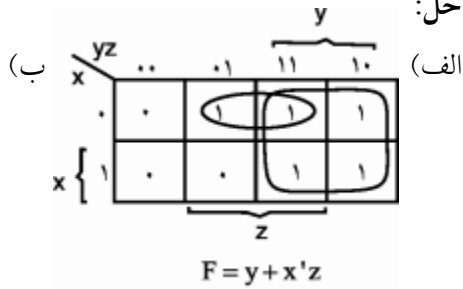
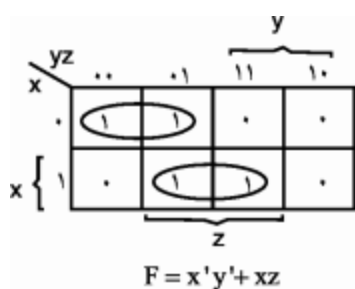
ب) $F = \Sigma(1,2,3,6,7)$

الف) $F = \Sigma(0,1,5,7)$

د) $F = \Sigma(0,2,3,4,6)$

ج) $F = \Sigma(3,5,6,7)$

حل:



۹-۱ - توابع بولی زیر را با نقشه چهار متغیره ساده کنید.

$F(A,B,C,D)$

ب) $F = \Sigma(3,7,11,13,14,15)$

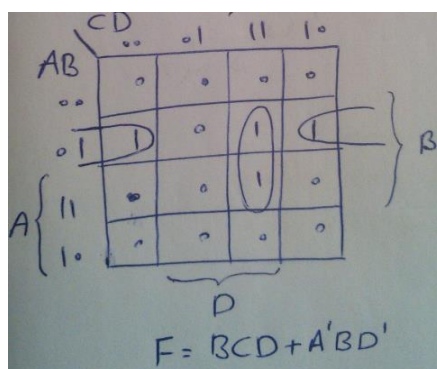
الف) $F = \Sigma(4,6,7,15)$

د) $F = \Sigma(0,2,4,5,6,7,8,10,13,15)$

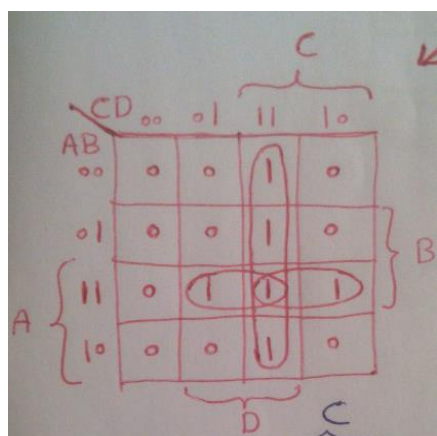
ج) $F = \Sigma(0,1,2,4,5,7,11,15)$

حل:

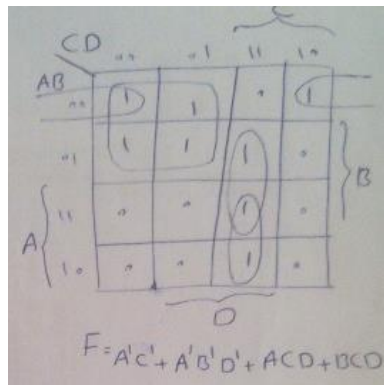
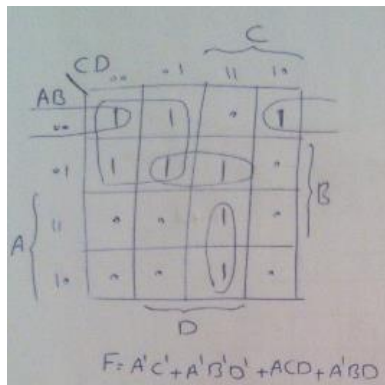
(الف)



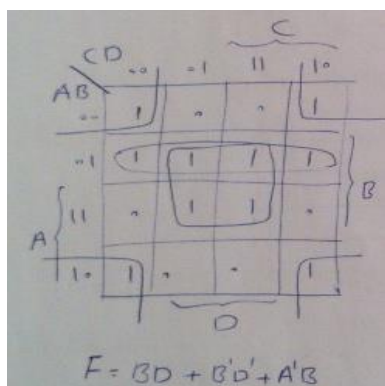
(ب)



$$F = CD + ABC + ABD$$



(ج)



(د)

۱-۱۰- عبارات زیر را (۱) به صورت جمع حاصلضربها و (۲) به صورت ضرب

حاصل جمعها ساده کنید:

الف) $x'z' + y'z' + yz' + xy$ ب) $AC' + B'D + A'CD + ABCD$

حل: ابتدا جدول کارنو هر کدام از توابع را رسم کرده، سپس حاصلضربها و

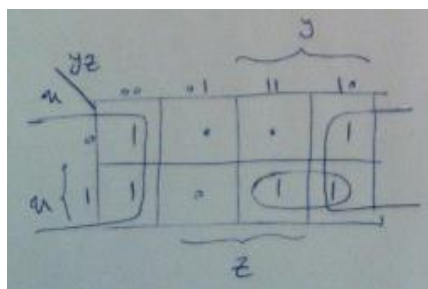
حاصل جمعها را بدست می آوریم.

الف)

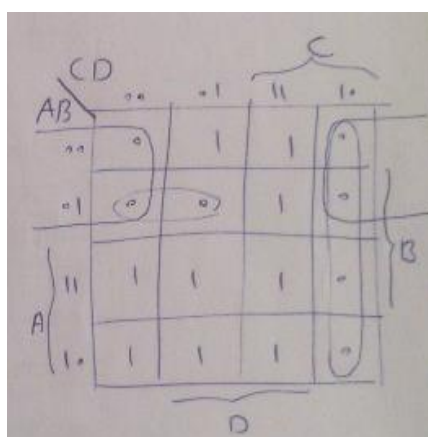
حاصل جمع حاصل ضرب $F = xy + z'$

حاصل ضرب حاصل جمع $F' = x'z + y'z$

$\Rightarrow F = (x + z')(y + z')$



(ب)



حاصل ضرب $F = AC' + CD + B'D$ حاصل جمع ضرب

حاصل ضرب $F = (A + D)(C' + D)(A + B' + C)$ حاصل جمع

توضیحات: در جدول کارنو عبارت (الف) و (ب)، دسته‌بندی‌ها، حاصل‌جمع حاصلضرب خود تابع را می‌سازد و دسته‌بندی ۰ها حاصل‌جمع حاصلضرب متمم عبارت را می‌سازد که با استفاده از قضیه دمورگان، حاصلضرب حاصل‌جمع عبارت معین می‌شود.

۱-۱۱- توابع بولی زیر را به شکل جمع حاصلضرب‌ها با نقشه چهار متغیره ساده

کنید. نمودار منطقی را

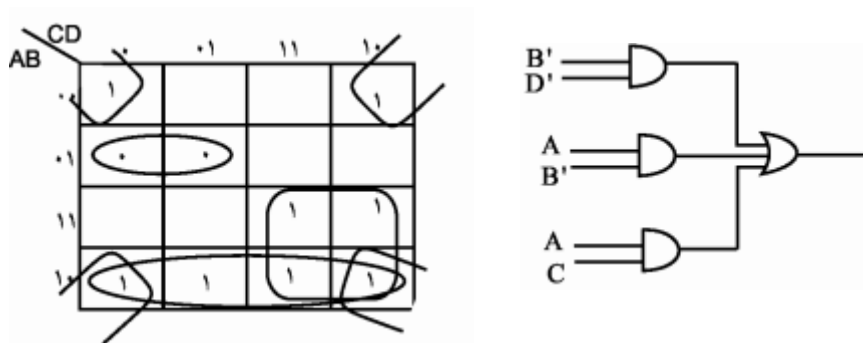
(ب) با گیت‌های NAND رسم کنید.

(الف) با گیت‌های AND-OR

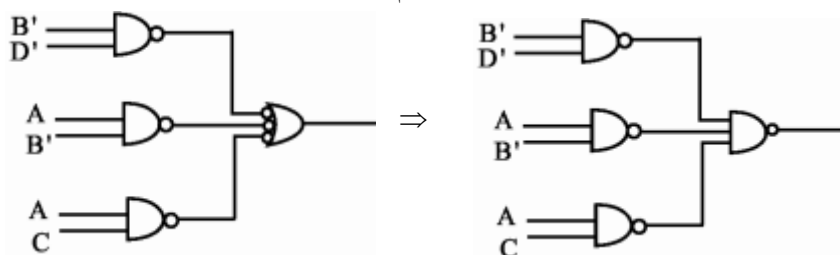
$$F = \Sigma(0, 2, 8, 9, 10, 11, 14, 15)$$

حل:

(الف)



(ب) ابتدا AND ها را به NAND تبدیل می کنیم:



۱-۱۲- تابع بولی زیر را به شکل ضرب حاصل جمع ها با نقشه چهار متغیره

ساده کنید. نمودار منطقی را

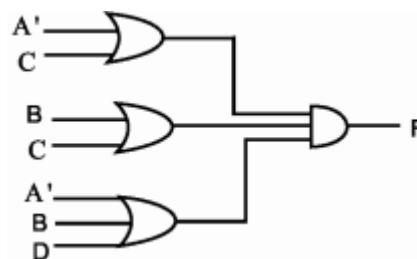
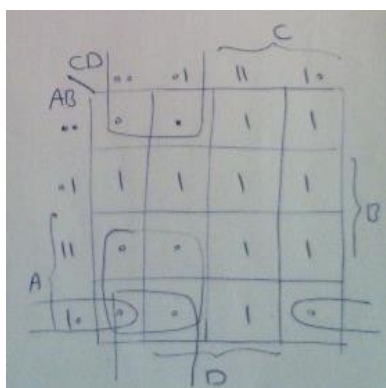
(ب) با گیت های NOR رسم کنید.

(الف) با گیت های OR-AND

$$F = \Sigma(2,3,4,5,6,7,11,14,15)$$

حل:

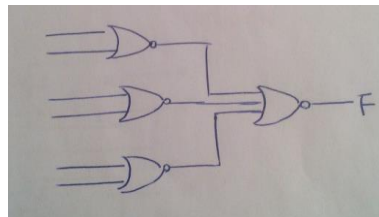
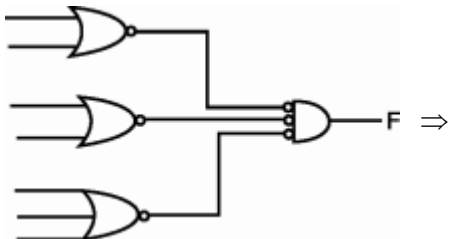
(الف)



$$F' = AC' + B'C' + AB'D'$$

$$F = (A' + C)(B + C)(A' + B + D)$$

ب) ابتدا ORها را به NOR تبدیل می کنیم:



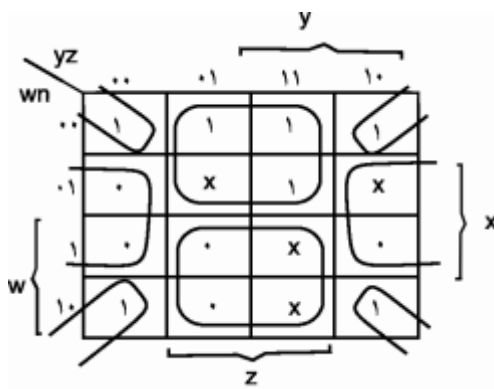
۱-۱۳- تابع بولی F را همراه با حالات بی اهمیت d به شکل (۱) جمع حاصل ضرب ها و (۲) ضرب حاصل جمع ها ساده کنید.

$$F(w, x, y, z)$$

$$F = \sum m(0, 1, 2, 3, 7, 8, 10)$$

$$F = \sum d(5, 6, 11, 15)$$

حل: با توجه به توضیحات سوال ۱۰ همین فصل داریم:



$$۱) \text{ جمع حاصل ضربها : } F = x'z' + w'z$$

$$۲) \text{ ضرب حاصل جمعها : } F = (x' + z)(w' + z')$$

۱۴-۱- با جدول ۱-۲ عبارت بولی S (جمع) خروجی یک تمام جمع کننده به شکل جمع حاصل ضربها را بدست آورید. سپس با دستکاریهای جبری نشان دهید که S می تواند به صورت OR انحصاری سه متغیره بیان شود.

$$S = x \oplus y \oplus z$$

حل: با توجه به جدول درستی NOR انحصاری،

$$S = x'y'z + x'yz' + xy'z' + xyz$$

$$= x'(y'z + yz') + x(y'z' + yz)$$

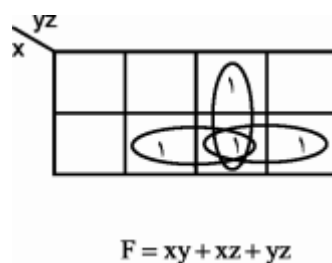
$$= x'(y \oplus z) + x(y \oplus z)'$$

$$= x \oplus y \oplus z$$

۱۵-۱- یک مدار ترتیبی به نام تابع اکثریت به این صورت تعریف می شود که هر گاه مقدار متغیرهای ورودی ۱، بیشتر از ورودیهای ۰ باشند، خروجی ۱ است، در غیر این صورت خروجی ۰ است. تابع اکثریت سه ورودی را طراحی کنید.

حل: جدول درستی را تشکیل داده و مقادیر مورد نیاز را به تابع تخصیص می دهیم سپس ساده سازی انجام می دهیم.

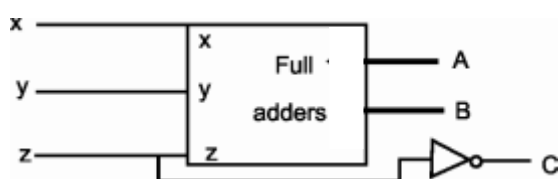
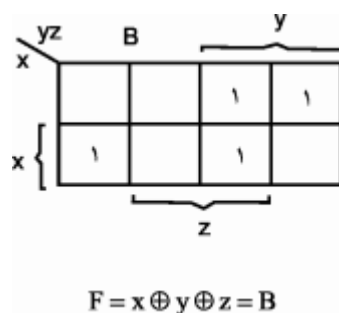
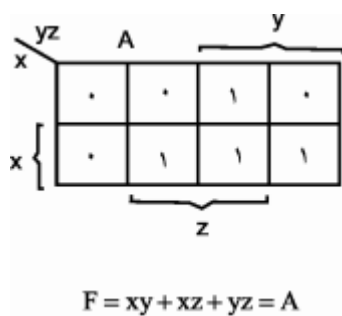
X	y	z	F
۰	۰	۰	۰
۰	۰	۱	۰
۰	۱	۰	۰
۰	۱	۱	۱
۱	۰	۰	۰
۱	۰	۱	۱
۱	۱	۰	۱
۱	۱	۱	۱



۱-۱۶. یک مدار ترکیبی با سه متغیر x و y و z و سه خروجی A و B و C طراحی کنید. اگر ورودی‌های دودویی ۰، ۱، ۲ یا ۳ باشد، خروجی یکی بیشتر از ورودی است. وقتی ورودی‌ها ۴ و ۵ و ۶ و ۷ باشند، خروجی دودویی یکی کمتر از ورودی باشد.

حل:

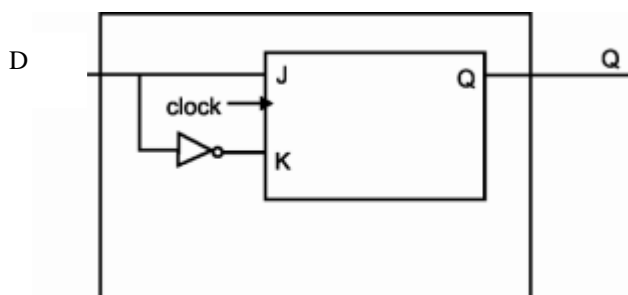
x	y	z	A	B	C
۰	۰	۰	۰	۰	۱
۰	۰	۱	۰	۱	۰
۰	۱	۰	۰	۱	۱
۰	۱	۱	۱	۰	۰
۱	۰	۰	۰	۱	۱
۱	۰	۱	۱	۰	۰
۱	۱	۰	۱	۰	۱
۱	۱	۱	۱	۱	۰



با دقت در توابع A و B تشکیل شده در می‌یابیم A و B ، خروجی‌های Full adder می‌باشند.

۱۷-۱- نشان دهید که یک فلیپ فلاپ JK را با قرار دادن یک وارونگر بین ورودی‌های J و K می‌توان به یک فلیپ فلاپ D تبدیل کرد.
حل:

طبق جداول تحریک فلیپ فلاپ‌ها و با توجه به شکل زیر:



داریم:

$$D=0 \Rightarrow J=0, K=1 \Rightarrow Q \rightarrow 0$$

$$D=1 \Rightarrow J=1, K=0 \Rightarrow Q \rightarrow 1$$

که زمانی

۱۸-۱ - با اطلاعات موجود در جدول مشخصه فلیپ فلاپ JK در جدول شکل

۱-۲ (ب) جدول تحریک را برای فلیپ فلاپ JK به دست آورده و جواب خود را با جدول ۱-۳ مقایسه کنید.

حل: با توجه به جدول مشخصه فلیپ فلاپ JK جدول درستی را به شکل زیر

تشکیل می دهیم:

J	K	Q(t+1)		J	K	Q(t)	Q(t+1)
۰	۰	Q(t)		۰	۰	۰	۰
۰	۱	۰	\Rightarrow	۰	۰	۱	۱
۱	۰	۱		۰	۱	۰	۰
۱	۱	Q'(t)		۰	۱	۱	۰
				۱	۰	۰	۱
				۱	۰	۱	۱
				۱	۱	۰	۱
				۱	۱	۱	۰

سپس با توجه به حالات $Q(t), Q(t+1)$ در جدول درستی بالا، مقادیر J و k مناسب

را استخراج کرده تا جدول تحریک JK بدست آید:

Q(t)	Q(t+1)	J	K
۰	۰	۰	x
۰	۱	۱	x
۱	۰	x	۱
۱	۱	x	۰

که با جدول ۱-۳ برابر است.

۱۹-۱- یک مدار ترتیبی دارای دو فلیپ فلاپ A و B، دو ورودی x و y و یک خروجی z است. معادله های ورودی فلیپ فلاپ و نیز خروجی مدار به قرار زیرند:

$$D_A = x'y + xA$$

$$D_B = x'B + xA$$

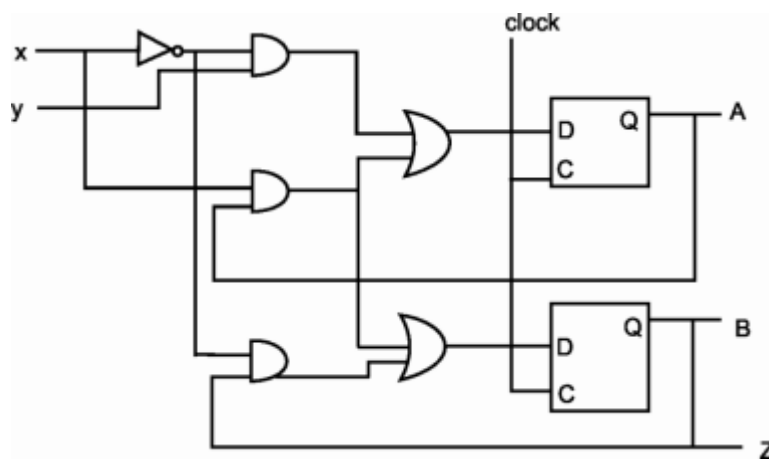
$$z = B$$

الف) دیاگرام منطقی مدار را رسم کنید.

ب) جدول حالت را برپا کنید.

حل:

الف) D_A, D_B مشخص می کنند که فلیپ فلاپ پیشنهادی از نوع D است.



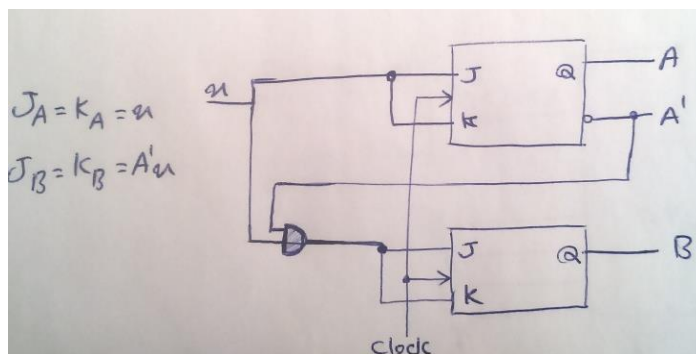
ب)

حالت فعلی		ورودی ها		حالت بعدی		خروجی
A	B	x	y	A	B	Z
.

۰	۰	۰	۱	۱	۰	۰
۰	۰	۱	۰	۰	۰	۰
۰	۰	۱	۱	۰	۰	۰
۰	۱	۰	۰	۰	۱	۱
۰	۱	۰	۱	۱	۱	۱
۰	۱	۱	۰	۰	۰	۱
۰	۱	۱	۱	۰	۰	۱
۱	۰	۰	۰	۰	۰	۰
۱	۰	۰	۱	۱	۰	۰
۱	۰	۱	۰	۱	۱	۰
۱	۰	۱	۱	۱	۱	۰
۱	۱	۰	۰	۰	۱	۱
۱	۱	۰	۱	۱	۱	۱
۱	۱	۱	۰	۱	۱	۱
۱	۱	۱	۱	۱	۱	۱

۱-۲۰. یک پایین شمارنده دو بیتی طراحی کنید. این مدار ترتیبی دارای دو فلیپ فلاپ و یک ورودی x است. وقتی $x=0$ باشد، حالت فلیپ فلاپ تغییر نمی‌کند. وقتی $x=1$ باشد، ترتیب حالات ۱۱، ۱۰، ۰۱، ۰۰ و تکرار آن است. حل: از فلیپ فلاپ JK استفاده می‌کنیم که هر دو حالت toggle و no-change را دارد.

شمارنده پایین رونده (down counter)



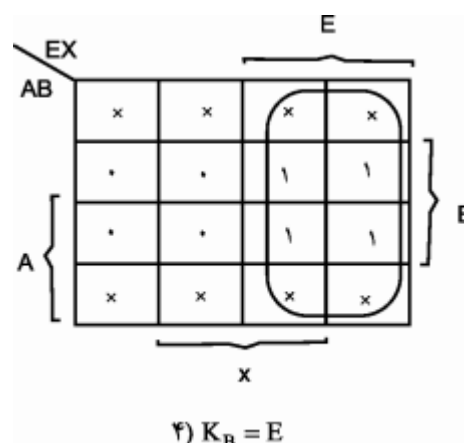
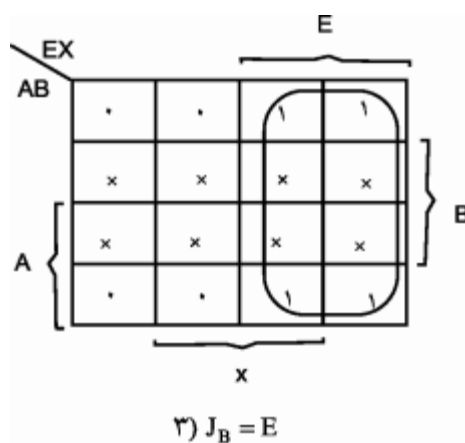
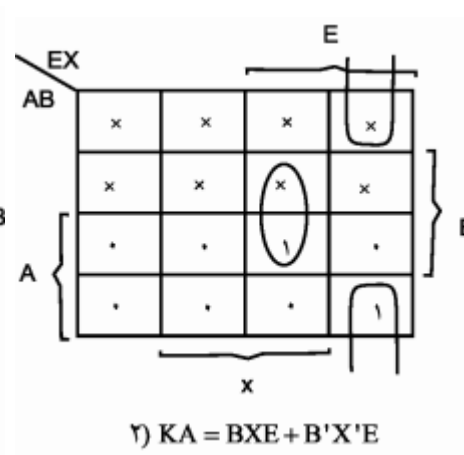
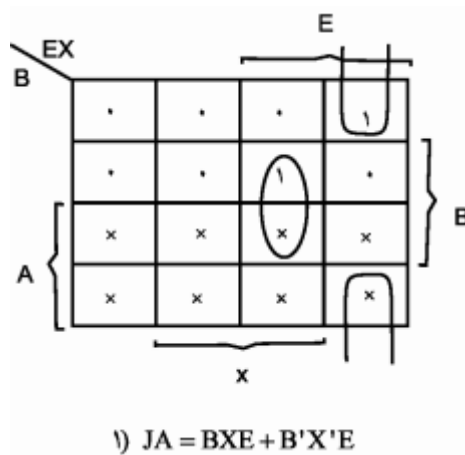
۲۱-۱. یک مدار ترتیبی با دو فلیپ فلاپ JK و دو ورودی E و x طراحی کنید. اگر $E=0$ باشد، مدار بدون توجه به مقدار x در همان حالت می ماند. وقتی $E=1$ و $x=1$ است، مدار به دنباله حالات از ۰۰ به ۰۱، به ۱۰، به ۱۱ و ۰۰ و تکرار آن وارد می شود. وقتی $E=1$ و $x=0$ باشد، مدار به حالت انتقالی ۰۰ به ۱۱، به ۰۱، به ۰۱ و ۰۰ و تکرار آن وارد می شود.

حل: این مدار یک شمارنده پایین و بالارونده است (up-down counter). E در اینجا نقش فعال کننده (Enable) را دارد.

حالت فعلی		ورودی ها		حالت بعدی		ورودی فلیپ فلاپ ها			
						J_A	K_A	J_B	K_B
۰	۰	۰	۰	۰	۰	۰	x	۰	x
۰	۰	۰	۱	۰	۰	۰	x	۰	x
۰	۰	۱	۰	۱	۱	۱	x	۱	x
۰	۰	۱	۱	۰	۱	۰	x	۱	x
۰	۱	۰	۰	۰	۱	۰	x	x	۰
۰	۱	۰	۱	۰	۱	۰	x	x	۰
۰	۱	۱	۰	۰	۰	۰	x	x	۱
۰	۱	۱	۱	۱	۰	۱	x	x	۱
۱	۰	۰	۰	۱	۰	x	۰	۰	x
۱	۰	۰	۱	۱	۰	x	۰	۰	x
۱	۰	۱	۰	۰	۱	x	۱	۱	x
۱	۰	۱	۱	۱	۱	x	۰	۱	x
۱	۱	۰	۰	۱	۱	x	۰	x	۰

۱	۱	۰	۱	۱	۱	x	۰	x	۰
۱	۱	۱	۰	۱	۰	x	۰	x	۱
۱	۱	۱	۱	۰	۰	x	۱	x	۱

حال توابع ورودی فلیپ فلاپ‌ها را به ترتیب بدست می‌آوریم:



فصل دوم: قطعات دیجیتال

۱- قطعه های TTL SSI بیشتر به صورت مدارهای مجتمع ۱۴ پایه عرضه می شوند. دو پایه برای منبع تغذیه در نظر گرفته شده و بقیه برای پایانه های ورودی و خروجی به کار می روند. چند مدار از نوع های زیر را در چنین بسته ای می توان گذاشت؟

الف) معکوس کننده:

ب) گیت OR انحصاری دو ورودی:

ج) گیت OR سه ورودی:

د) گیت AND چهار ورودی:

ه) گیت NOR پنج ورودی:

و) گیت NAND هشت ورودی:

ز) فلیپ فلاپ JK ساعت دار با پاک کننده غیر همزمان:

حل:

الف) معکوس کننده:

گیت $6 = \frac{12}{2} \Rightarrow$ هر معکوس کننده ۲ پین دارد.

ب) گیت OR انحصاری دو ورودی:

گیت $4 = \frac{12}{3} \Rightarrow$ XOR دو ورودی ۳ پین دارد.

ج) گیت OR سه ورودی:

گیت $3 = \frac{12}{4} \Rightarrow$ OR سه ورودی ۴ پین دارد.

د) گیت AND چهار ورودی:

گیت $2 = \frac{12}{5} \Rightarrow$ AND چهار ورودی ۵ پین دارد.

ه) گیت NOR پنج ورودی:

گیت $\frac{12}{6}=2 \Rightarrow \text{NOR}$ پنج ورودی ۶ بین دارد.

و) گیت NAND هشت ورودی:

گیت $\frac{12}{9}=1 \Rightarrow \text{NAND}$ هشت ورودی ۹ بین دارد.

ز) فلیپ فلاپ JK ساعت‌دار با پاک‌کننده غیر همزمان:

با توجه به غیر همزمان بودن، پایه‌های Clock و Reset جداگانه برای هر فلیپ فلاپ در نظر گرفته می‌شود پس:

فلیپ فلاپ $\frac{12}{6}=2 \Rightarrow$ هر فلیپ فلاپ JK ۶ بین

۲- تراشه‌های MSI موجود توابع دیجیتال ساده‌ای مانند دیکدر مولتی پلکسر، ثبات‌ها و شمارنده‌ها را ممکن می‌سازند. تراشه‌های زیر مدارهای مجتمع از نوع TTL هستند که این توابع را تولید می‌کنند. ویژگی‌های آنها را در کتاب راهنما یافته و آنها را با تراشه‌های متناظر ارائه شده در این فصل مقایسه کنید.

الف) IC نوع ۷۴۱۵۵، دیکدر دوتایی ۲ به ۴

ب) IC نوع ۷۵۱۵۷، مولتی پلکسر چهار تایی ۲ به ۱ خطی

ج) IC نوع ۷۴۱۹۴، شیفت رجیستر چهار بیت دو طرفه با بار شدن موازی

د) IC نوع ۷۴۱۶۳، شمارنده دو دویی چهار بیت با بار شدن موازی و پاک شدن

همزمان

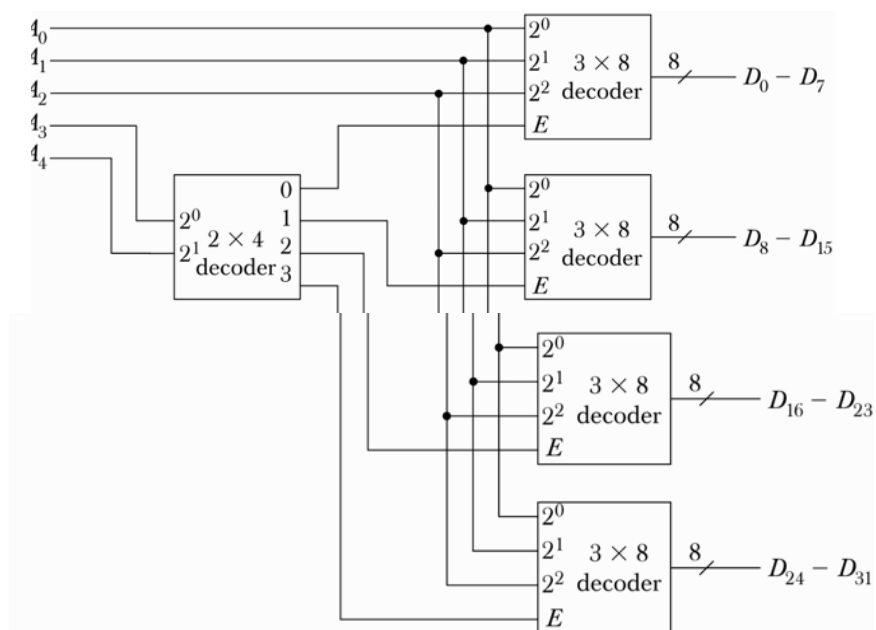
حل: جستجو و مقایسه با مدل‌های شکل‌های فصل ۲ کتاب

۳- یک دیکدر ۵ به ۳۲ را با چهار دیکدر ۳ به ۸ دارای ورودی تواناساز و یک

دیکدر ۲ به ۴ بسازید. از بلاک دیاگرام مشابه شکل ۲-۴ استفاده کنید.

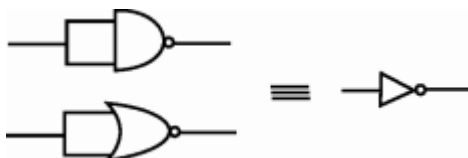
حل: از دیکدر ۲ به ۴ به منظور فعال‌سازی هر کدام از دیکدرهای ۳ به ۸ استفاده

می‌شود.

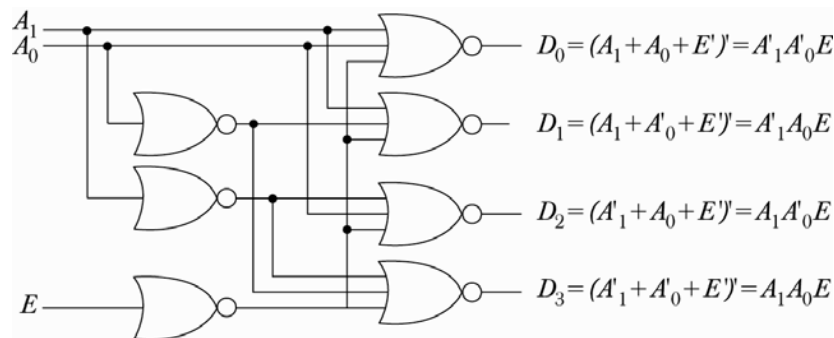


۴- نمودار منطقی یک دیکدر ۲ به ۴ را فقط با گیت NOR رسم کنید. ورودی تواناساز را نیز در نظر بگیرید.

حل: در حالت کلی، برای ساخت گیت NOT از گیت‌های NAND و NOR، ورودی به پایه‌های متصل به هم این گیت‌ها متصل می‌شود.



برای دیاگرام منطقی دیکدر ۲ به ۴ داریم :



۵- دیکدر شکل ۲-۳ را تغییر دهید طوری که مدار اگر $E=1$ باشد فعال و اگر $E=0$ باشد، غیرفعال باشد. جدول درستی را برای نوع تغییر یافته رسم کنید.

حل: در صورت سوال، بایست شکل ۲-۳ ذکر شود که اشتباهاً ۲-۲ ذکر شده بود.

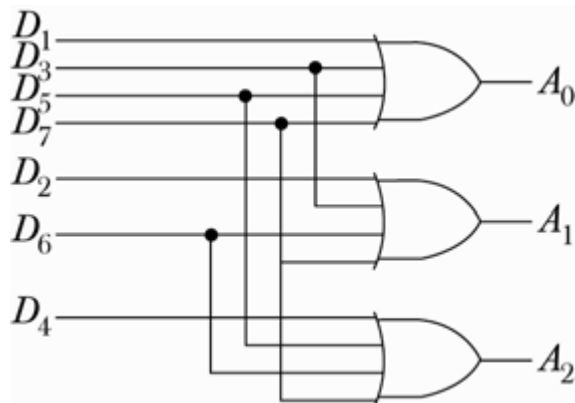
این کار با حذف گیت NOT متصل به ورودی E ، امکان‌پذیر است.

جدول درستی به شکل زیر است:

E	A_1	A_0	D_0	D_1	D_2	D_3
۰	X	X	۱	۱	۱	۱
۱	۰	۰	۰	۱	۱	۱
۱	۰	۱	۱	۰	۱	۱
۱	۱	۰	۱	۱	۰	۱
۱	۱	۱	۱	۱	۱	۰

۶- نمودار منطقی یک انکدر با هشت ورودی و سه خروجی، که جدول درستی آن در جدول ۲-۲ آورده شده را رسم کنید. وقتی تمام ورودی‌ها ۰ باشند خروجی چیست؟ اگر فقط ورودی D برابر ۰ باشد، خروجی چیست؟ روشی پیشنهاد دهید که این دو حالت را متمایز کند.

حل: با توجه به جدول ۲-۲ داریم:

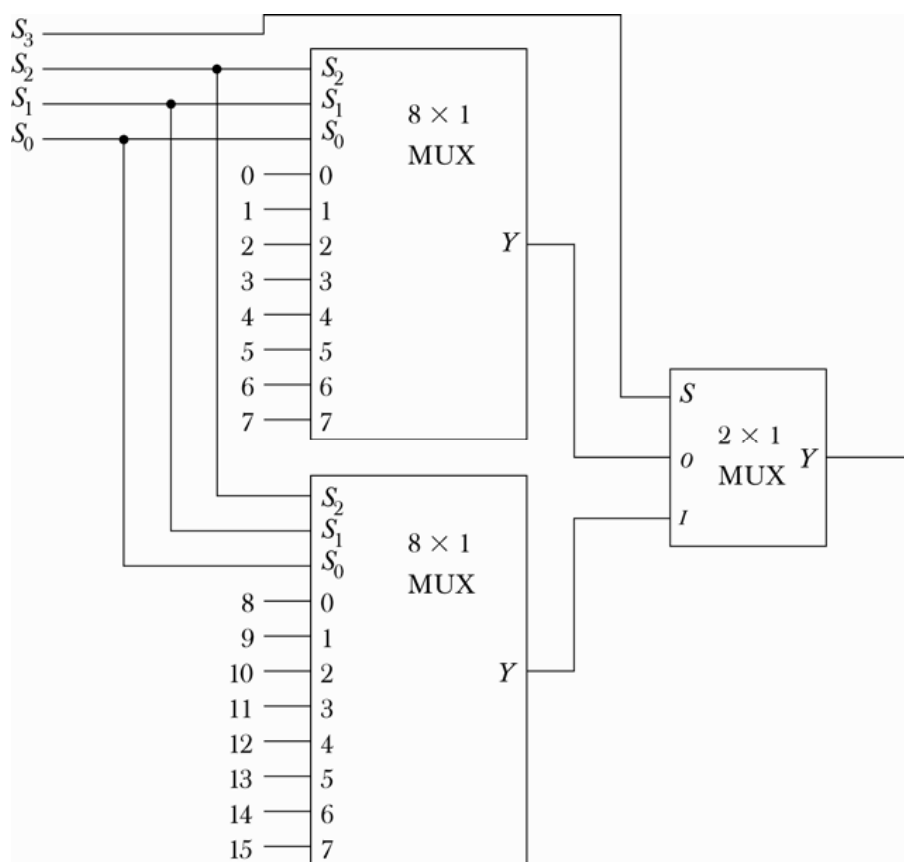


اگر همه‌ی ورودی‌ها برابر ۰ و یا اگر فقط $D_0 = 1$ باشد، خروجی‌های $A_2A_1A_0$ برابر ۰۰۰ خواهد بود.

برای تمایز بین این دو حالت، می‌بایست یک خروجی اضافه برای تشخیص حالتی که همه ورودی‌ها ۰ باشند در نظر بگیریم و آن را بیت **invalid** بنامیم. این بیت را می‌توان از طریق **NOR** همه ورودی‌ها بدست آورد.

۷- یک مولتی پلکسر ۱۶ به ۱ را با دو مولتی پلکسر ۸ به ۱ و یک مولتی پلکسر ۲ به ۱ بسازید. برای هر سه مولتی پلکسر از بلاک دیاگرام استفاده کنید.

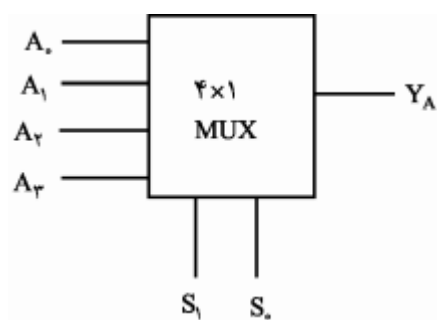
حل: ۴ بیت ورودی را $S_3S_2S_1S_0$ می‌نامیم. از سه بیت $S_2S_1S_0$ برای انتخاب ورودی متناظر در مولتی پلکسر ۸ به ۱ استفاده می‌کنیم. از بیت S_3 برای انتخاب ورودی مورد نظر به وسیله‌ی مولتی پلکسر ۲ به ۱ استفاده می‌کنیم. پس داریم:



۸ - بلاک دیاگرام یک مولتی پلکسر را رسم کنید و کارکرد آن را با جدول تابع

توضیح دهید.

حل: برای مولتی پلکسر ۴ به ۱ داریم:

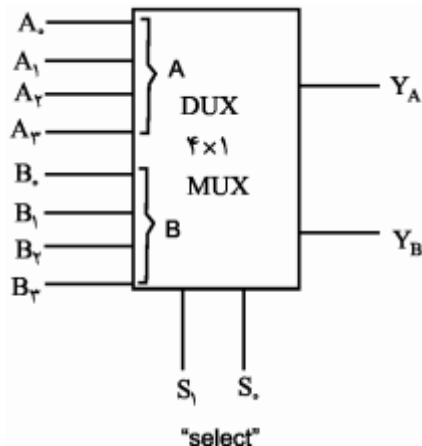


S_1	S_0	Y_A
۰	۰	A_0
۰	۱	A_1
۱	۰	A_2
۱	۱	A_3

(select)

جدول تابع

البته در صورت سوال کتاب لاتین، بلاک دیاگرام یک مولتی پلکسر دوگانه (Dual

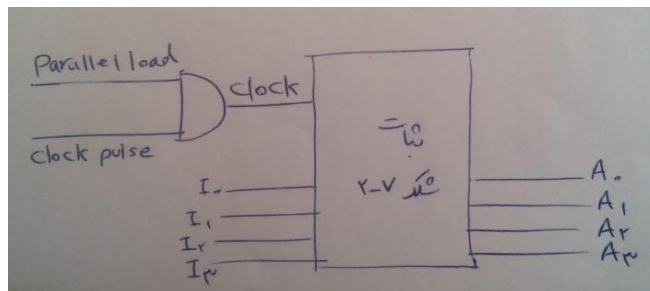


(MUX) درخواست شده است که داریم:

S_1	S_0	Y_A	Y_B
۰	۰	A_0	B_0
۰	۱	A_1	B_1
۱	۰	A_2	B_2
۱	۱	A_3	B_3

۹- یک گیت AND دو ورودی را در ثبات شکل ۷-۲ جای دهید و خروجی این گیت را به ورودی‌های ساعت همه فلیپ فلاپ‌ها وصل کنید. یکی از ورودی‌های گیت AND پالس‌های ساعت را از مولد پالس ساعت دریافت می‌کند. ورودی دیگر گیت AND، کنترل بار شدن موازی را فراهم می‌کند. کارکرد ثبات جدید را شرح دهید.

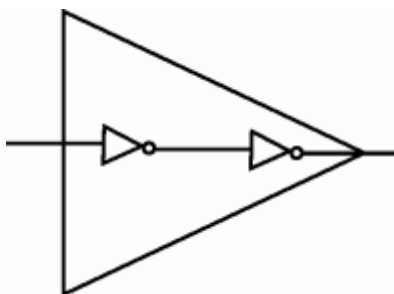
حل:



زمانی که ورودی "parallel load"، ۱ است، امکان بار موازی رجیستر فراهم می‌آید و زمانی که این ورودی، ۰ است، خروجی گیت AND ۰ شده و اطلاعات رجیستر به طور تضمین شده، نگهداری خواهد شد. یعنی با تغییر احتمالی و اشتباهی ورودی‌های $(I_0 - I_3)$ ، بارگذاری صورت نمی‌گیرد.

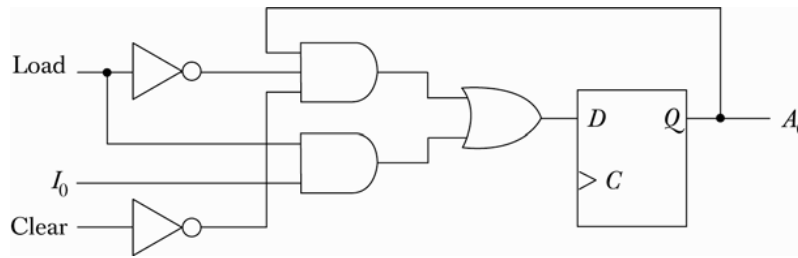
۱۰- هدف گیت بافر در ورودی ساعت ثبات شکل ۸ - ۲ چیست؟

حل: با توجه به توضیحات صفحه‌ی ۵۲ کتاب، این گیت توان دریافتی لازم را از مولد ساعت کاهش می‌دهد. هر گاه پالس ساعت فقط به ورودی یک گیت در عوض چند گیت وصل شود، توان کمتری لازم است، ولی چنانچه این گیت بافر مورد استفاده قرار نگرفته و پالس ساعت مستقیماً به هر چهار ورودی وصل شود. توان بیشتری از مولد پالس اخذ خواهد شد. ساختمان این بافر می‌تواند به صورت زیر باشد.



۱۱- به ثبات دارای امکان بار شدن موازی در شکل ۸ - ۲، امکان ورودی پاک‌کننده را اضافه کنید.

حل:



جدول تابع به صورت زیر است:

پاک‌کننده بارگذاری

عمل	D	Clear	Load
بلا تغییر	Q(t)	۰	۰
پاک کردن به ۰	۰	۱	۰
بار کردن I ₀	I ₀	X	۱

۱۲- مقدار اولیه محتوای یک ثبات ۱۱۰۱ است. ثبات شش بار با ورودی سری

۱۰۱۱۰۱ به راست شیفت داده می‌شود. محتوای ثبات پس از هر شیفت چیست؟

حل:

رجیستر ۴ بیتی	بیت‌های ورودی
۱۱۰۱	۱
۱۱۱۰	۰
۰۱۱۱	۱
۱۰۱۱	۱
۱۱۰۱	۰
۰۱۱۰	۱
۱۰۱۱	

۱۳- فرق بین انتقال سری و موازی چیست؟ با یک شیفت رجیستر دارای بار

شدن موازی توضیح دهید چگونه می‌توان ورودی سری را به خروجی موازی و برعکس تبدیل کرد.

حل:

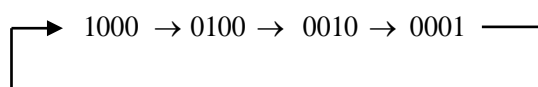
انتقال سریال: در هر زمان یک بیت انتقال، به وسیله شیفت

انتقال موازی: هر تعداد بیت به صورت همزمان

ورودی سری از طریق شیفت به خروجی موازی تبدیل شده و ورودی موازی از طریق بار موازی به خروجی سری از طریق شیفت تبدیل می‌شود. نیاز به یک شمارنده برای تعیین تعداد بیت «مثلاً ۸ بیت در هر کلمه» وجود دارد.

۱۴- شمارنده حلقوی یک شیفت رجیستر شکل ۹-۲ می‌باشد که خروجی سری‌اش به ورودی سری آن وصل شده است. با شروع از حالت اولیه ۱۰۰۰، رشته حالات چهار فلیپ فلاپ را پس از هر شیفت نشان دهید.

حل: اگر شیفت رجیستر به راست فرض شود داریم:



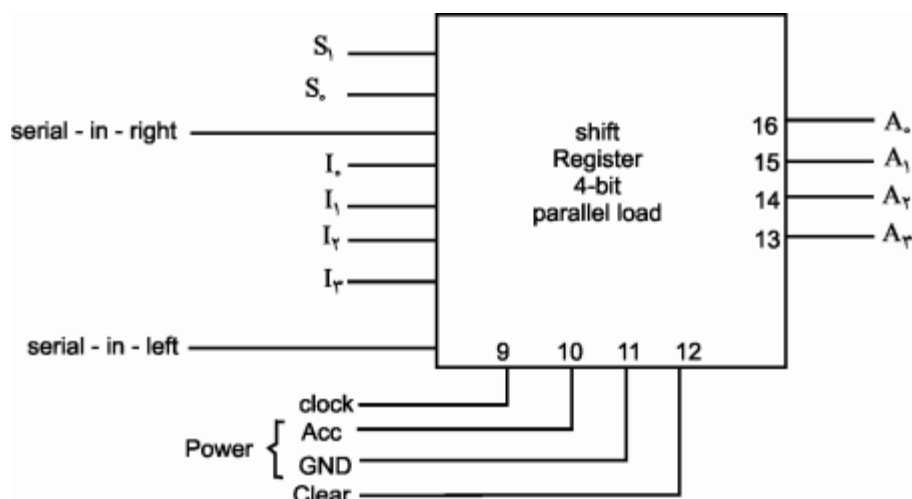
۱۵- یک شیفت رجیستر چهار بیتی دو طرفه با ویژگی بار شدن موازی طبق شکل ۱۰-۲ در یک مدار مجتمع، بسته‌بندی شده است.

الف) بلاک دیاگرام مدار مجتمع را با تمام ورودی‌ها و خروجی‌ها رسم کنید. دو پایه برای منبع تغذیه قرار دهید.

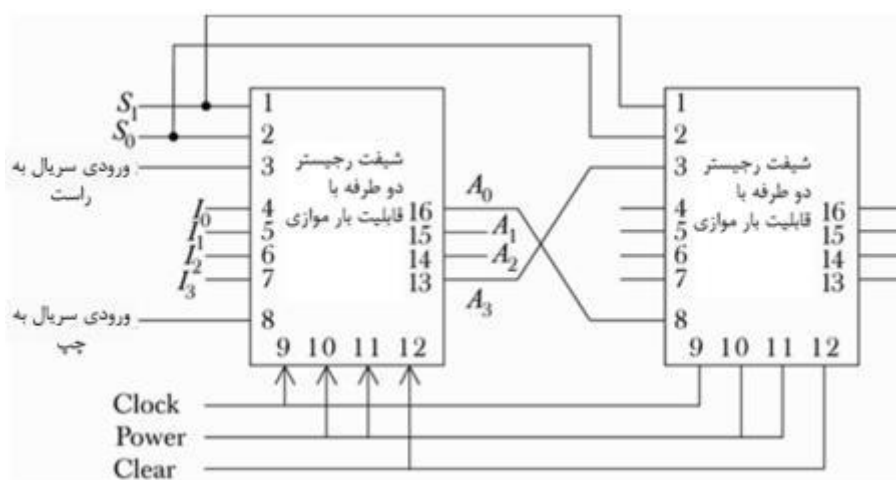
ب) با دو مدار مجتمع بلاک دیاگرام یک شیفت رجیستر ۸ بیتی با ویژگی بار شدن موازی را رسم کنید.

حل:

الف)



(ب)



۱۶- در شمارنده دودویی ده بیتی، چند فلیپ فلاپ در شمارش بعدی متمرکز می‌شوند.

می‌شوند.

(ب) ۰۰۱۱۱۱۱۱۱۱

(الف) ۱۰۰۱۱۰۰۱۱۱۱

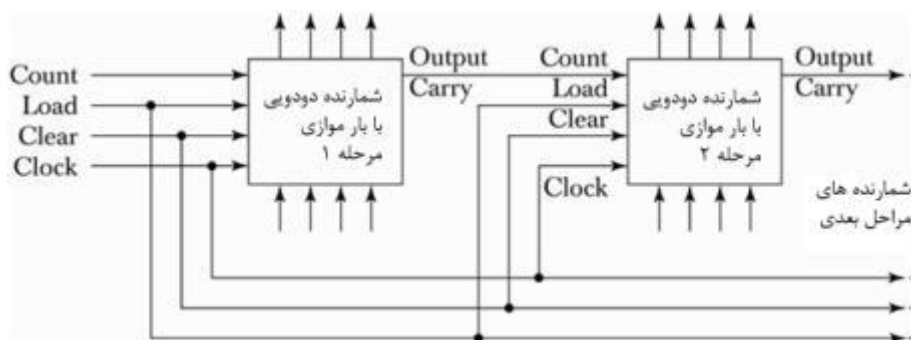
حل: در هر مورد، عدد با عدد ۰۰۰۰۰۰۰۰۱ جمع دودویی می‌شود و تعداد متمم‌ها حساب می‌شود.

$$\text{الف) } ۴ \text{ مورد } ۱۰۰۱۱۰۱۰۰۰ + ۰۰۰۰۰۰۰۰۱ = ۱۰۰۱۱۰۱۱۱$$

$$\text{ب) } ۹ \text{ مورد } ۰۱۰۰۰۰۰۰۰ + ۰۰۱۱۱۱۱۱۱ = ۰۱۱۱۱۱۱۱۱$$

۱۷- اتصالات لازم بین چهار شمارنده دودویی چهار بیتی با بار شدن موازی (شکل ۲-۱۲) برای ایجاد یک شمارنده دودویی ۱۶ بیتی با بار شدن موازی را نشان دهید. برای هر شمارنده چهار بیتی از یک بلاک دیاگرام استفاده کنید.

حل: نکته مهم در این سؤال، آن است که در رابطه با گسترش شمارنده‌ها بیت نقلی خروجی هر مرحله (شمارنده) به فعال‌ساز شمارش مرحله «شمارنده» بعدی متصل می‌شود و بیت‌های Load، Clear و Clock در همه شمارنده‌ها مشترک هستند.



۱۸- نشان دهید که چگونه یک شمارنده دودویی با بار شدن موازی شکل ۲-۱۲ را می‌توان به یک شمارنده تقسیم بر N تبدیل کرد.

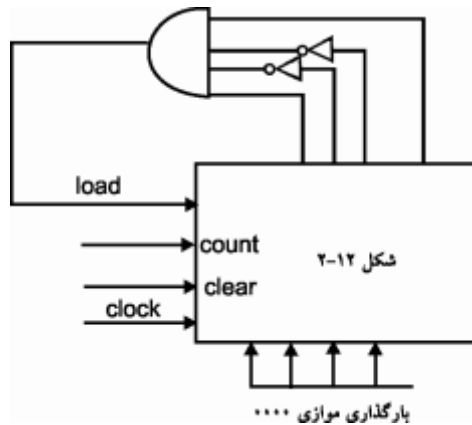
(یعنی شمارنده‌ای که از ۰۰۰۰ تا N شمرد و به ۰۰۰۰ باز گردد).

به عنوان یک حالت خاص، مدار یک تقسیم بر ۱۰ را با شکل ۲-۱۲ و یک گیت AND خارجی رسم کنید.

حل: زمانی که لفظ تقسیم بر N گفته می‌شود. مفهوم این است که محتوای شمارنده

از ۰۰۰۰ تا خود N-1 پیش رفته و بعد از آن ۰۰۰۰ به رجیستر Load می‌شود.
طبق این توضیحات، در این سؤال داریم: شمارنده از ۰۰۰۰ تا ۱۰۰۱ پیش رفته و بعد از آن دارای ۰۰۰۰ می‌شود.

پس:



۱۹- واحدهای حافظه زیر با تعداد کلمات در تعداد بیت‌ها مشخص شده‌اند. در هر مورد چند خط آدرس و چند خط داده ورودی - خروجی نیاز است.

الف) $2k \times 16$ ۲) $64k \times 8$

ج) $16M \times 32$ ۴) $4G \times 64$

حل:

الف) $16 \times 2^{11} = 2K \times 16$ ۱۱ خط آدرس ۱۶ خط داده

ب) $8 \times 2^{16} = 64K \times 8$ ۱۶ خط آدرس ۸ خط داده

ج) $32 \times 2^{24} = 16M \times 32$ ۲۴ خط آدرس ۳۲ خط داده

د) $64 \times 2^{32} = 4G \times 64$ ۳۲ خط آدرس ۶۴ خط داده

۲۰- تعداد بایت‌هایی را که می‌توان در حافظه‌های مسئله ۱۹-۲ ذخیره کرد

مشخص کنید.

حل:

الف) $2K \times 16 = 4K \text{ byte} = 4096 \text{ bytes}$

ب) $64K \times 8 = 64K \text{ byte} = 2^{16} \text{ bytes}$

ج) $16M \times 32 = 2^{24} \times 4 = 2^{26} \text{ bytes}$

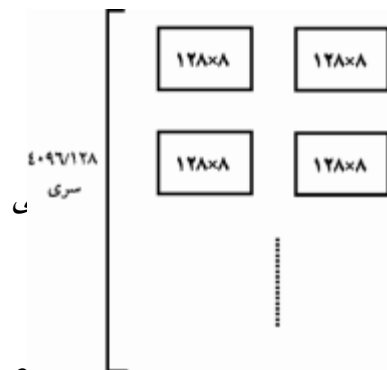
د) $4G \times 64 = 2^{32} \times 8 = 2^{35} \text{ bytes}$

۲۱- چند تراشه حافظه 128×8 برای ساخت یک حافظه 4096×16 نیاز است؟

حل: در این مسائل، ابتدا تعیین می‌کنیم که شکل حافظه نهایی براساس حافظه در

دسترس چگونه است. با توجه به تعداد بیت‌ها:

$$\Rightarrow \frac{4096 \times 16}{128 \times 8} = \frac{2^{12} \times 2^4}{2^7 \times 2^3} = 2^6 = 64 \text{ تراشه}$$



۲۲- با داشتن یک ROM 32×8 بیتی و یک و

لازم برای ساخت یک ROM 128×8 و یک دیک

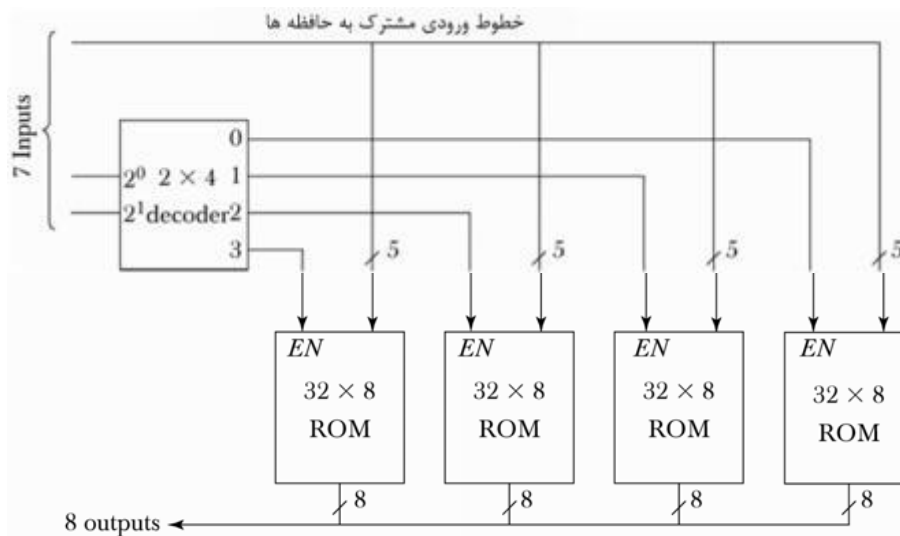
حل:

۸ خط داده ۵ خط آدرس $32 \times 8 = 2^5 \times 8$

۸ خط داده ۷ خط آدرس $128 \times 8 = 2^7 \times 8$

با استفاده از یک دیکدر 2×4 و دو بیت با ارزش آدرس ROM 128×8 ، فعال‌سازی

هر کدام از ROM های 32×8 را انجام می‌دهیم.



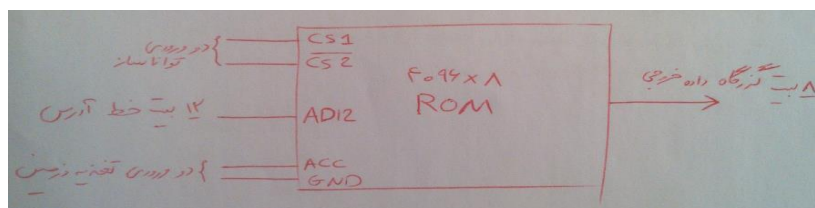
۲۳- یک تراشه ROM 4096×8 بیتی دارای دو ورودی تواناساز بوده و با منبع تغذیه ۵ ولت کار می‌کند. چند پایه برای بسته مدار مجتمع نیاز است؟ بلاک دیاگرام آن را رسم و پایانه‌های ورودی و خروجی را نام‌گذاری کنید.

حل:

۸ خط داده «خروجی» + ۱۲ خط آدرس $4096 \times 8 = 2^{12} \times 8$

پین ۲۴ = دو پین منبع تغذیه برای ACC و زمین + دو ورودی تواناساز

بدین ترتیب بسته مدار مجتمع را رسم و پایانه‌ها را نامگذاری می‌کنیم:



فصل سوم: نمایش داده‌ها

۱- اعداد دودویی زیر را به دهدهی تبدیل کنید.

حل: با توجه به ارزش رقم‌ها در مبنای ۲:

$$(101110)_2 = 32 + 8 + 4 + 2 = 46$$

$$(1110101)_2 = 64 + 32 + 16 + 4 + 1 = 117$$

$$(110110100)_2 = 256 + 128 + 32 + 16 + 4 = 436$$

۲- اعداد زیر را، که پایه‌هایشان $۵, ۱۰, ۲۰, ۴۰$ به دهدهی تبدیل کنید.

$$(50)_4$$

حل:

با توجه به ارزش ارقام در پایه مربوطه:

$$(12121)_3 = 3^4 + 2 \times 3^3 + 3^2 + 2 \times 3 + 1 = 81 + 54 + 9 + 6 + 1 = 151$$

$$(4310)_5 = 4 \times 5^3 + 3 \times 5^2 + 5 = 500 + 75 + 5 = 580$$

$$(50)_4 = 5 \times 7 = 35$$

$$(198)_{12} = 12^2 + 9 \times 12 + 8 = 144 + 108 + 8 = 260$$

۳- اعداد دهدهی زیر را به دودویی تبدیل کنید.

حل:

$$(1231)_{10} = 1024 + 128 + 64 + 8 + 4 + 2 + 1 = 2^{10} + 2^7 + 2^6 + 2^3 + 2^2 + 2^1 + 2^0$$

بیت‌های متناظر با توان‌هایی از ۲ که ظاهر شده‌اند، ۱ و بقیه صفر هستند.

$$(1231)_{10} = 2^{10} + 2^7 + 2^6 + 2^3 + 2^2 + 2^1 + 2^0 = (10011001111)_2$$

پس داریم: روش دیگر استفاده از تقسیم متوالی است که در آن معکوس دنباله‌ی باقی‌مانده‌ها،

معادل عدد در مبنای دلخواه (در اینجا ۲) است.

قسمت ۱

تقسیم را تا جایی ادامه می‌دهیم که خارج قسمت از مبنای مورد نظر (۲) کوچکتر شود.

دنباله : ۱۱۱۱۰۰۱۱۰۰۱

معکوس دنباله : ۱۰۰۱۱۰۰۱۱۱۱

همانند مثال اول داریم:

$$(673)_{10} = 512 + 128 + 32 + 1 = 2^9 + 2^7 + 2^5 + 2^0 = (1010100001)_2$$

$$(1998)_{10} = 1024 + 512 + 256 + 128 + 64 + 8 + 4 + 2$$

$$= 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^3 + 2^2 + 2^1$$

$$= (11111001110)_2$$

همان‌طور که دیدید روش اول یک روش سرانگشتی برای محاسبه راحت‌تر است

ولی روش دوم، یک الگوریتم دقیق است.

۴- اعداد دهمی زیر را به مبنای مشخص شده تبدیل کنید.

الف) ۷۵۶۲ به هشت هشتی ب) ۱۹۳۸ به شانزده شانزدهی

ج) ۱۷۵ به دودویی

حل: می‌توان از همان روش تقسیم متوالی استفاده کرد با در نظر گرفتن این که ارقام

مبنای ۱۶، اعداد ۱۰ (A) تا ۱۵ (F) را در بر می‌گیرند.

پس داریم:

$$(7562)_{10} = (16612)_8$$

$$(1938)_{10} = (792)_{16}$$

$$(175)_{10} = (10101111)_2$$

۵- عدد شانزده شانزدهی F3A7C2 را به دودویی و هشت هشتی تبدیل نمایید.

حل: با توجه به این که ۱۶ توان چهارم ۲ است، هر رقم مبنای ۱۶ معادل ۴ رقم مبنای ۲ است که با جایگزینی ارقام عدد با معادل دودویی آن داریم:

$$F3A7C2 = (1111\ 0011\ 1010\ 0111\ 1100\ 0010)$$

همچنین با توجه اینکه ۸ توان سوم ۲ است، هر سه رقم مبنای ۲ معادل یک رقم مبنای ۸ است پس با جایگزینی از سمت راست داریم:

$$111\ 100\ 111\ 010\ 011\ 111\ 000\ 010$$

که معادل است با

$$74723702$$

۶- اگر جواب معادله درجه دوم $x^2 - 10x + 31 = 0$ برابر $x = 5$ و $x = 8$ باشد،

پایه عددها چیست؟

حل:

$$(x^2 - 10x + 31)_r = [(x - 5)(x - 8)]_{10}$$

$$= x^2 - (5 + 8)_{10}x + (40)_{10}$$

با معادل قرار دادن ضرایب داریم:

$$(10)_r = (13)_{10}$$

$$0 + r = 13 \Rightarrow r = 13$$

$$(31)_r = (40)_{10} \Rightarrow 1 + 3 \times r = 40 \Rightarrow 3r = 39 \quad \text{همچنین:}$$

$$\Rightarrow r = 13$$

$$x^2 - (r + 0)x + 3r + 1 = 0 \Rightarrow 25 - 5r + 3r + 1 = 0 \quad \text{یا } (x = 5):$$

$$\Rightarrow 2r = 26$$

$$r = 13$$

۷- مقدار همه بیت‌های یک ثبات ۱۲ بیتی که عدد ۲۱۵ دهدهی را نگه می‌دارد:

الف) به دودویی:

$$(215)_{10} = 128 + 64 + 16 + 4 + 2 + 1 =$$

$$(11010111)_2$$

$$\Rightarrow 000011010111$$

ب) به هشت هشتی کد شده با دودویی:

$$000 \ 011 \ 010 \ 111$$

$$0 \quad 3 \quad 2 \quad 7$$

ج) به شانزده شانزدهی کد شده با دودویی:

$$0000 \ 1101 \ 0111$$

$$0 \quad D \quad 7$$

د) به دهدهی کد شده با دودویی:

$$0010 \quad 0001 \quad 0101$$

$$2 \quad 1 \quad 5$$

همان‌طور که می‌بینیم هشت هشتی و شانزده شانزدهی کد شده با دودویی یک عدد، با دودویی آن عدد معادل است.

۸- آرایش یک ثبات ۲۴ بیتی را که محتوایش عدد دهدهی ۲۹۵ است:

$$\text{الف) به دودویی: } (295)_{10} = 0000 \ 0000 \ 0000 \ 0001 \ 0010 \ 0111$$

$$\text{ب) به BCD: } 0000 \ 0000 \ 0000 \ 0010 \ 1001 \ 0101$$

ج) به ASCII با استفاده از هشت هشتی با توازن زوج:

$$10110010 \quad 00111001 \quad 00110101$$

در ۳ عدد دودویی بالا، بیت توازن زوج به با ارزشترین بیت توجه شود.

۹- نام خود را با یک کد ۸ بیتی به ASCII نوشته و بیت سمت چپ را ۰ در

نظر بگیرید. بین نام و نام خانوادگی خود از فضای خالی استفاده کنید و در صورت

استفاده از حرف اول نام، پس از آن نقطه بگذارید.

حل: Morris Mano

M(01001101) o(01101111) r(01110010) r(01110010) i(01101001) s(01110011)
(01000000) M(01001101) a(01100001) n(01101110) o(01101111)

۱۰- کد ASCII زیر را رمزگشایی کنید:

1001010 1001111 1001000 1001110 0100000 1000100 1001111 1000101

حل: به ترتیب داریم

J O H N ' ' D O E
JOHN DOE

۱۱- متمم ۹ اعداد دهدهی هشت رقمی زیر را بدست آورید:

12349876 ; 00980100 ; 90009951 ; 00000000

حل: به ترتیب داریم

87650123 ; 99019899 ; 09990048 ; 99999999

۱۲- متمم ۱۰ اعداد دهدهی شش رقمی زیر با دست آورید.

123900 ; 090657 ; 100000 ; 000000

حل: به ترتیب داریم

876100 ; 909343 ; 900000 ; 000000

۱۳- متممهای ۱ و ۲ اعداد دودویی هشت رقمی زیر را به دست آورید.

حل:

10101110 , 10000001 , 10000000 , 00000001 , 00000000

متمم ۱: 01010001 , 01111110 , 01111111 , 11111110 , 11111111

متمم ۲: 01010010 , 01111111 , 10000000 , 11111111 , 00000000

برای متمم ۱ به جای ۰، ۱ و به جای ۱، ۰ جایگزین می‌کنیم.

برای متمم ۲، از سمت راست شروع کرده، تا رسیدن به اولین ۱ صفرهای عدد را

بدون تغییر می‌گذاریم، همچنین اولین ۱ را بدون تغییر می‌گذاریم و تمام بیت‌های سمت

چپ اولین یک (از سمت راست) را متمم ۱ می‌کنیم.

۱۴- تفریق را با اعداد دهدهی بدون علامت زیر و با متمم ۱۰ مفروق انجام

دهید:

الف) ۱۳۲۱ - ۵۲۵۰

$$\begin{array}{r} 5250 \\ + 8679 \\ \hline 13929 \end{array}$$

که متمم ۱۰ آن، 6887- است.

ب) ۸۶۴۰ - ۱۷۵۳

صفر بودن رقم نقلی به معنی منفی بودن حاصل است که برای بدست آمدن حاصل

باید نتیجه را متمم ۱۰ کنیم.

$$\begin{array}{r} 1753 \\ + 1360 \\ \hline 03113 \end{array}$$

که متمم ۱۰ آن، 080- است.

ج) ۱۰۰ - ۲۰

مانند قسمت ب داریم:

$$\begin{array}{r} 020 \\ + 900 \\ \hline 0920 \end{array}$$

د) ۲۵۰ - ۱۲۰۰

$$\begin{array}{r} 1200 \\ + 9750 \\ \hline 10950 \end{array}$$

۱۵- تفریق را با اعداد دودویی بدون علامت زیر و با متمم ۲ مفروق انجام

دهید.

الف) ۱۰۰۰۰ - ۱۱۰۱۰

$$\begin{array}{r} 11010 \\ +10000 \\ \hline 101010 \end{array} \quad 26-16=10$$

ب) $11010 - 11010$

$$\begin{array}{r} 11010 \\ +10011 \\ \hline 101101 \end{array} \quad 26-13=13$$

ج) $100 - 110000$

$$\begin{array}{r} 000100 \\ +010000 \\ \hline 1010100 \end{array}$$

$$(4 - 48 = -44)$$

همانند سؤال قبل نیاز به متمم ۲ داریم. پس متمم ۲ می شود -101100

د) $1010100 - 1010100$

$$\begin{array}{r} 1010100 \\ +0101100 \\ \hline 10000000 \end{array}$$

$$84-84=0$$

۱۶- اعمال حسابی $(-13) + (+42)$ و $(-13) - (-42)$ را در دودویی و با متمم ۲ علامت دار برای اعداد منفی انجام دهید.

حل:

$+42 = 0101010$	$+13 = 0001101$
$-42 = 1010110$	$-13 = 1110011$
$(+42) \ 0101010$	$(-42) \ 1010110$
$(-13) \ 1110011$	$(+13) \ 0001101$
$\hline (+29) \ 0011101$	$\hline (-29) \ 1100011$

همان‌طور که می‌دانیم در صورتی که حاصل منفی باشد، به صورت متمم ۲ بدست می‌آید.

۱۷- اعمال حسابی $(+80) + (+70)$ و $(-80) + (-70)$ را با نمایش اعداد دودویی به شکل متمم ۲ علامت‌دار انجام دهید. هشت بیت برای نمایش هر عدد به همراه علامت آن بکار برید. نشان دهید که در هر دو مورد سرریز رخ می‌دهد، دو نقلی آخر نامساوی اند و علامت وارون می‌شود.

حل:

+70	01000110	-70	10111010
+80	01010000	-80	10110000
+150	10010110	-150	010101010

در محاسبه سمت چپ، جواب بزرگتر از +127 است. پس جواب منفی و سرریز رخ داده است.

در محاسبه سمت راست، جواب کوچکتر از -128 است. پس جواب مثبت و سرریز رخ داده است.

۱۸- اعمال حسابی زیر را برای اعداد دهدهی مشخص شده با استفاده از نمایش متمم ۱۰ علامت‌دار برای اعداد منفی انجام دهید.

الف) $(-638) + (+785)$

(-638)	9362
$(+785)$	<u>+0785</u>
$(+147)$	0147

ب) $(-638) - (+185)$

(-638)	9362
(-185)	<u>+9815</u>
(-823)	9177

که نیاز به متمم گیری دارد.

۱۹- یک عدد دودویی ممیز شناور ۳۶ بیتی دارای ۸ بیت همراه علامت برای نما و ۲۶ بیت همراه علامت برای مانتیس است. مانتیس یک کسر نرمال شده است. اعداد مانتیس و نما به فرم مقدار علامت دار هستند. بزرگترین و کوچکترین عدد مثبتی که با این شکل می توان نشان داد، به جز صفر، کدامند؟

$$\text{بزرگترین: } 2^{+255} \times (1 - 2^{-26})$$

$$\text{بزرگترین: } 2^{-255} \times (2^{-1})$$

۲۰- عدد $(+46,5)_{10}$ را به صورت یک عدد دودویی ممیز شناور ۲۴ بیتی نشان دهید. مانتیس که کسر نرمال شده است ۱۶ بیت و نما ۸ بیت دارند.

$$46,5 = 32 + 8 + 4 + 2 + 0,5 = 2^5 + 2^3 + 2^2 + 2^1 + 2^{-1} = (101110,1)_2$$

پس داریم:

۰۱۰۱۱۱۰۱۰۰۰۰۰۰۰۰

و چون کسر را نرمال کرده ایم نما برابر است با:

$$00000110(+6)$$

۸ بیت نما

۲۱- کد گری گاهی کد بازتابی گفته می شود زیرا مقدار بیت ها در دو طرف هر یک از توان های ۲ ، بازتاب یکدیگرند، مثلاً همانگونه که در جدول ۵ -۳ ملاحظه می شود، مقدار سه بیت کم ارزش در دو طرف خطی که بین ۷ و ۸ کشیده می شود بازتاب یکدیگرند. با استفاده از این ویژگی کد گری به دست آورید:

الف) اعداد کد گری برای ۱۶ تا ۳۱ به عنوان ادامه جدول ۳-۵

ب) کد گری افزونی ۳ برای اعداد دهدهی ۱۰ تا ۱۹ به عنوان ادامه جدول ۶-۳

حل:

الف:

Decimal	Gray Code
16	11000
17	11001
18	11011
19	11010
20	11110
21	11111
22	11101
23	11100
24	10100
25	10101
26	10111
27	10110
28	10010
29	10011
30	10001
31	10000

نکته: یک روش ساده برای بدست آوردن کدگری یک عدد دودویی به این صورت است:

از سمت چپ شروع کرده، اولین رقم را عیناً می‌نویسیم و برای رقم‌های بعدی OR انحصاری رقم عدد اصلی و رقم سمت چپ آن را در نظر می‌گیریم.

مثال:

10110 1011001

دودویی

↓

11101 1110101

گری

پس ابتدا افزونی ۳ را محاسبه کرده و سپس کدگری آن را برای ستون بعد بدست

می آوریم.

دهدہی	افزونی ۳	کدگری افزونی ۳
-------	----------	-------------------

۹	۱۱۰۰	۱۰۱۰
۱۰	۱۱۰۱	۱۰۱۱
۱۱	۱۱۱۰	۱۰۰۱
۱۲	۱۱۱۱	۱۰۰۰
۱۳	۱۰۰۰۰	۱۱۰۰۰
۱۴	۱۰۰۰۱	۱۱۰۰۱
۱۵	۱۰۰۱۰	۱۱۰۱۱
۱۶	۱۰۰۱۱	۱۱۰۱۰
۱۷	۱۰۱۰۰	۱۱۱۱۰
۱۸	۱۰۱۰۱	۱۱۱۱۱
۱۹	۱۰۱۱۰	۱۱۱۰۱
۲۰	۱۰۱۱۱	۱۱۱۰۰
	۱۱۰۰۰	۱۰۱۰۰

۲۲- عدد دهدهی ۸۶۲۰ را به شکل‌های زیر بدست آورید.

الف) BCD: 1000 0110 0010 0000

ب) افزودنی ۳: 1011 1001 0101 0011

ج) کد ۲۴۲۱: 1110 1100 0010 0000

د) دودویی: $8192+256+128+32+8+4$

$$=10000110101100$$

۲۳- ده رقم BCD را با توازن زوج در سمت چپ آن (در کل پنج بیت در هر

رقم) نشان دهید. این کار را با توازن فرد تکرار کنید.

حل:

BCD با توازن فرد BCD با توازن زوج دهدهی

۰	۰۰۰۰۰	۱۰۰۰۰
۱	۱۰۰۰۱	۰۰۰۰۱
۲	۱۰۰۱۰	۰۰۰۱۰
۳	۰۰۰۱۱	۱۰۰۱۱
۴	۱۰۱۰۰	۰۰۱۰۰
۵	۰۰۱۰۰۱	۱۰۱۰۱
۶	۰۰۱۱۰	۱۰۱۱۰
۷	۱۰۱۱۱	۰۰۱۱۱
۸	۱۱۰۰۰	۰۱۰۰۰
۹	۰۱۰۰۱	۱۱۰۰۱

توازن فرد، عکس توازن زوج است.

۲۴- عدد دهدهی ۳۹۸۴ را به صورت کد ۲۴۲۱ جدول ۶-۳ نشان دهید. تمام بیت‌های عدد کد شده را متمم کنید و نشان دهید که نتیجه همان متمم نه عدد ۳۹۸۴ با کد ۲۴۲۱ است.

حل:

$$3984 = 0011 \ 1111 \ 1110 \ 0100$$

که متمم آن می‌شود

$$1100 \ 0000 \ 0001 \ 1011 = 6015 \rightarrow$$

که همان متمم ۹ عدد ۳۹۸۴ است (این موضوع به دلیل خاصیت خود متممی کد ۲۴۲۱ است).

۲۵- نشان دهید که تابع OR انحصاری $x = A \oplus B \oplus C \oplus D$ یک تابع فرد است. یکی راه این است که جدول درستی $y = A \oplus B$ و $z = C \oplus D$ را بدست آورده و سپس جدول درستی $x = y \oplus z$ را بدست آورید. نشان دهید که $x = 1$ است به

شرطی که تعداد ۱ ها در A، B، C و D فرد باشد.

حل:

A B	$Y = A \oplus B$
0 0	0
0 1	1
1 0	1
1 1	0

C D	$Z = C \oplus D$
0 0	0
0 1	1
1 0	1
1 1	0

y	z	$x = y \oplus z$
0	0	0
0	1	1 ← $\begin{cases} AB=00 \text{ یا } 11 \\ CD=01 \text{ یا } 10 \end{cases}$
1	0	1 ← $\begin{cases} AB=01 \text{ یا } 10 \\ CD=00 \text{ یا } 11 \end{cases}$
1	1	0

ABCD
0001, 0010, 1101, 1110
0100, 0111, 1000, 1011

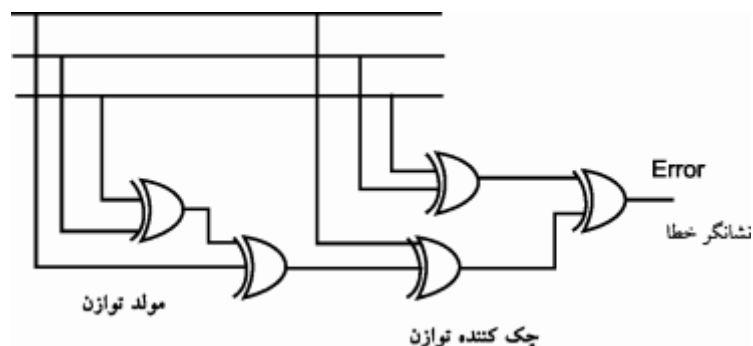
همواره تعداد فردی 1 وجود دارد

۲۶- مدارهای یک سازنده توازن سه بیتی و بررسی کننده توازن چهار بیتی را با استفاده از بیت توازن زوج بدست آورید (مدارهای شکل ۳-۳ توازن فرد را به کار برده‌اند).

حل:

$$p = x \oplus y \oplus z \text{ و } \bar{p} = x \oplus y \oplus z \oplus 1$$

این مدارها همانند شکل ۳-۳ هستند با این تفاوت که حباب‌های متمم‌ساز را در خروجی خود ندارند.



فصل چهارم: انتقال ثبات‌ها و ریز عمل‌ها

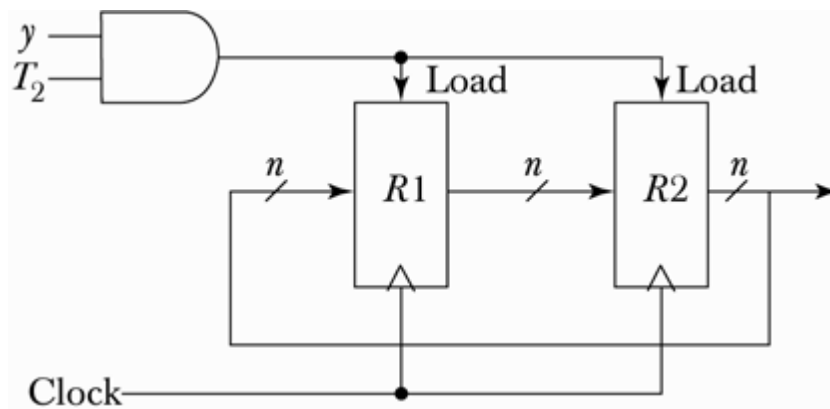
۱- بلاک دیاگرام سخت‌افزاری مشابه شکل ۴-۲ (الف) که عبارت انتقال ثباتی

زیر را پیاده‌سازی می‌کند رسم کنید:

$$yT_2 : R_2 \leftarrow R_1, R_1 \leftarrow R_2$$

حل: عبارت yT_2 نشان‌دهنده‌ی لزوم فعال بودن همزمان y و T_2 برای انجام

عملیات‌ها است. پس:



توضیح آنکه در ارتباط با صحت داده‌های موجود در رجیسترها در عملیات جابجایی (swap)، این صحت با توجه به مفاهیم setup ، hold time، propagation time time مربوط به سیگنال clock تفسیر می‌گردد.

۲- خروجی چهار ثبات R_0, R_1, R_2, R_3 با یک مولتی پلکسر 4×1 به

ورودی‌های ثبات پنجم R_5 وصل شده‌اند. هر ثبات هشت بیتی است. انتقال‌های

لازم توسط چهار متغیر زمان‌بندی T_0 تا T_3 به صورت زیر تعیین می‌شود:

$$T_0 : R_5 \leftarrow R_0$$

$$T_1 : R_5 \leftarrow R_1$$

$$T_2 : R_5 \leftarrow R_2$$

$$T_3 : R_5 \leftarrow R_3$$

متغیرهای زمان دو به دو منحصراند. یعنی در هر زمان معین فقط یک متغیر ۱ است، در حالی که سه متغیر دیگر ۰ هستند. بلاک دیاگرامی رسم کنید که نشان‌دهنده پیاده‌سازی سخت‌افزاری انتقال ثباتی باشد. اتصالات لازم برای چهار متغیر زمان‌بندی به خطوط انتخاب مولتی پلکسر و ورودی بار کردن ثبات R_5 را نیز نشان دهید.

حل: جدول درستی را با توجه به توضیحات سؤال آماده می‌کنیم: S_0 و S_1

ورودی‌های انتخاب MUX هستند.

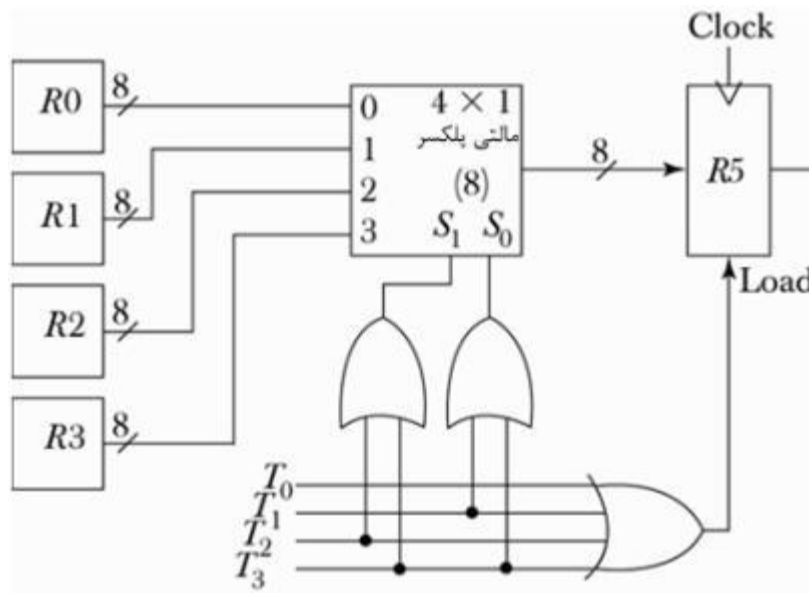
T_0	T_1	T_2	T_3	S_1	S_0	Load
۰	۰	۰	۰	X	X	۰
۱	۰	۰	۰	۰	۰	۱
۰	۱	۰	۰	۰	۱	۱
۰	۰	۱	۰	۱	۰	۱
۰	۰	۰	۱	۱	۱	۱

$$\Rightarrow S_1 = T_2 + T_3$$

$$S_0 = T_1 + T_3$$

$$\text{Load} = T_0 + T_1 + T_2 + T_3$$

پس مدار به شکل زیر خواهد بود:



۳- عبارت کنترل شرطی زیر را با دو عبارت انتقال ثباتی با توابع کنترل نشان

دهید:

$\text{IF}(P=1)\text{ then } (R_1 \leftarrow R_2) \text{ elseif } (Q=1)\text{ then } (R_1 \leftarrow R_3)$

حل:

$P: R_1 \leftarrow R_2$

$P'Q: R_1 \leftarrow R_3$

دلیل استفاده از P' در عبارت دوم، لفظ “else” در عبارت کنترل صورت سؤال است.

۴- برای اینکه سیستم گذرگاه شکل ۳-۴ داده را از هر ثبات به هر ثبات دیگر

منتقل کند چه باید کرد؟ به ویژه اتصالاتی را که باید برای فراهم کردن یک مسیر از

خروجی‌های ثبات C به ورودی‌های ثبات A وصل نمود نشان دهید.

حل: برای این کار، ۴ خط گذرگاه مشترک را به ورودی ۴ رجیستر وصل می‌نماییم.

همچنین به منظور انتخاب رجیستر مقصد، با استفاده از یک دیکدر 2×4 و ورودی‌های $S_0, S_1, Load$ ، رجیستر مقصد را فعال کرده تا داده از روی گذرگاه مشترک بر روی رجیستر مقصد بارگذاری شود.

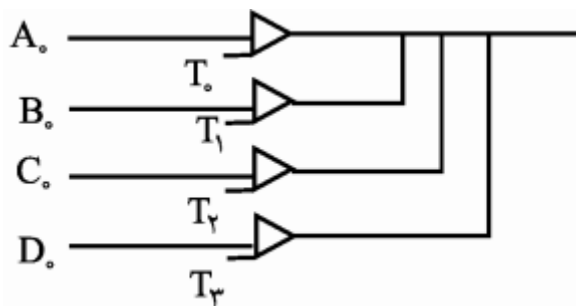
برای نمونه، برای انتقال محتویات رجیستر C به رجیستر A، $S_1 S_0 = 10$ را اعمال کرده تا رجیستر C مبدأ قرار گیرد. سپس Load رجیستر A را فعال کرده و کلاک پالس به منظور انتقال «بارگذاری» اعمال می‌کنیم.

۵ - یک سیستم گذرگاه مانند شکل ۳-۴ رسم کنید، ولی از بافرهای سه حالت و دیکدر به جای مولتی پلکسر استفاده کنید.

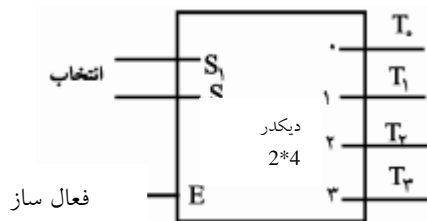
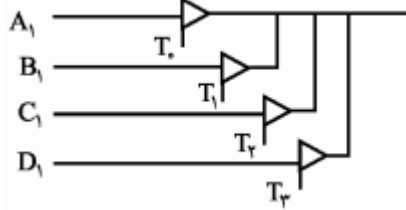
حل: دو روش طراحی با استفاده از پافر سه حالت می‌توان ارائه داد.

روش اول، به این صورت است که از بافرهای سه حالتی برای انتخاب هر بیت از بیت‌های ۰ تا ۳۱، دوم و سوم در همه رجیسترها استفاده می‌شود. به شکل زیر:

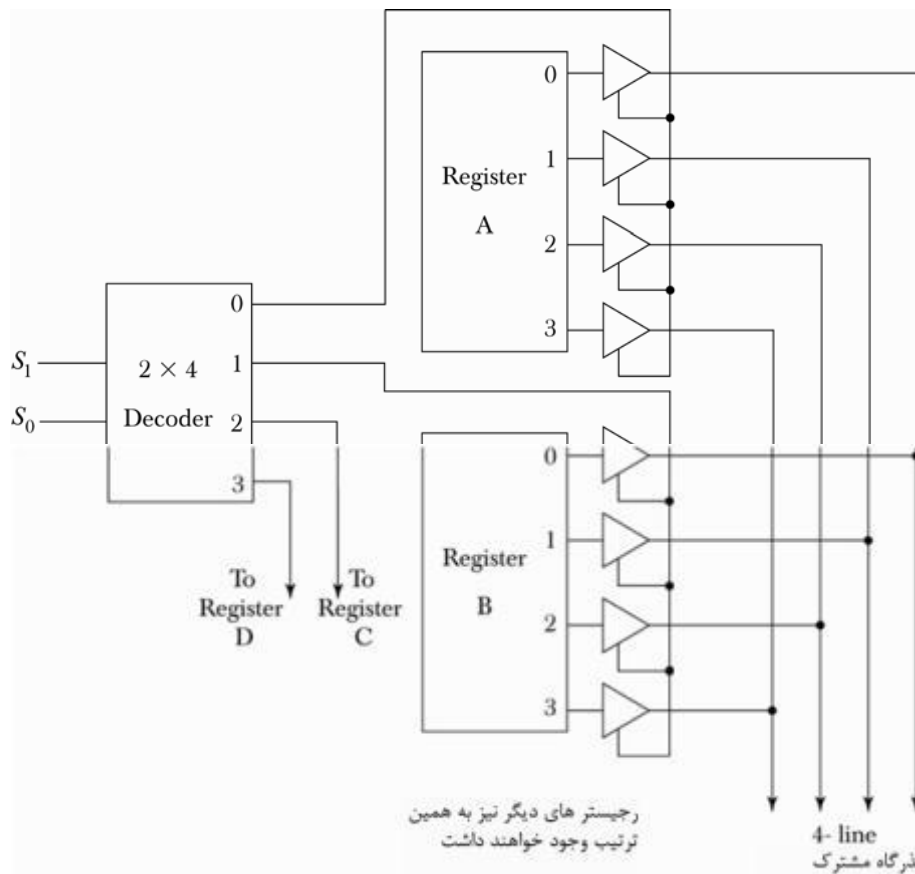
بیت • برای گذرگاه



بیت ۱ برای گذرگاه



تولید بیت‌های ۲ و ۳ برای گذرگاه مانند بالا طراحی شود.
 روش دوم، به این صورت است که از بافرهای سه حالت، برای فعال یا امپدانس بالا کردن یک رجیستر کامل استفاده می‌شود. به شکل زیر:



- ۶- یک کامپیوتر دیجیتال دارای سیستم گذرگاه مشترک برای ۱۶ ثابت ۳۲ بیتی است. گذرگاه با مولتی پلکسر ساخته شده است.
- الف) در هر مولتی پلکسر چند ورودی انتخاب وجود دارد.
- ب) اندازه مولتی پلکسر چیست.
- ج) چند مولتی پلکسر در گذرگاه است.

حل:

الف) ۴ خط ورودی برای مولتی پلکسر برای انتخاب یکی از ۱۶ رجیستر

ب) مولتی پلکسر 16×1

ج) ۳۲ مالتی پلکسر

۷- عبارت های زیر انتقال را در یک حافظه را مشخص می کنند. در هر حالت

عمل حافظه را توضیح دهید.

الف) $R_2 \leftarrow M[AR]$ ب) $M[AR] \leftarrow R_3$ ج) $R_5 \leftarrow M[R_5]$

حل:

الف) محتوای موجود در حافظه به آدرس AR خوانده شده و در R_2 نوشته

می شود.

ب) محتوای رجیستر R_3 در خانه حافظه به آدرس AR نوشته می شود.

ج) محتوای موجود در حافظه به آدرس موجود در R_5 ، خوانده شده و در R_5

نوشته می شود. در ضمن مقدار قبلی R_5 «آدرس مورد نظر از حافظه» از بین می رود.

۸- بلاک دیاگرام سخت افزاری را رسم کنید که عبارت های زیر را پیاده سازی

می کند.

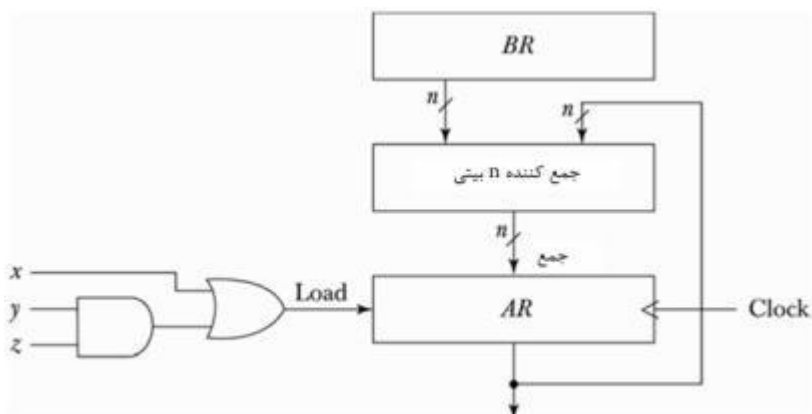
$$x + yz : AR \leftarrow AR + BR$$

که در آن AR و BR دو ثبات n بیتی، و x و y و z متغیر های کنترلی هستند.

گیت های منطقی را برای توابع کنترلی قرار دهید. (توجه شود سمبل + در بخش

کنترل، عمل OR، و در ریز عمل جمع به معنی بعلاوه است)

حل:

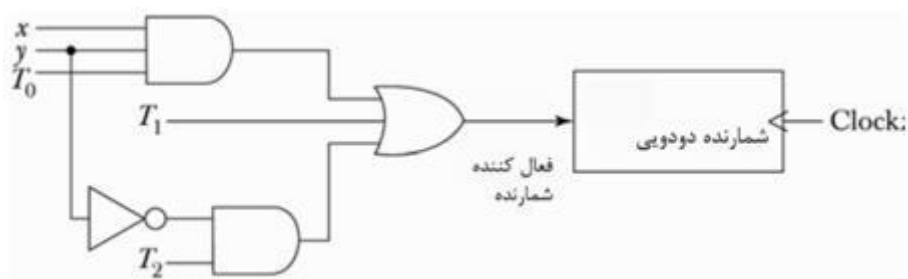


۹- سخت‌افزاری را نشان دهید که عبارت زیر را پیاده‌سازی کند. برای تابع کنترل گیت‌های منطقی و برای شمارنده دودویی با ورودی تواناساز شمارش، بلاک دیاگرام را قرار دهید.

$$xyT_0 + T_1 + y'T_2 : AR \leftarrow AR + 1$$

حل:

می‌توان عمل $AR \leftarrow AR + 1$ را با شمارش دودویی انجام داد. پس:



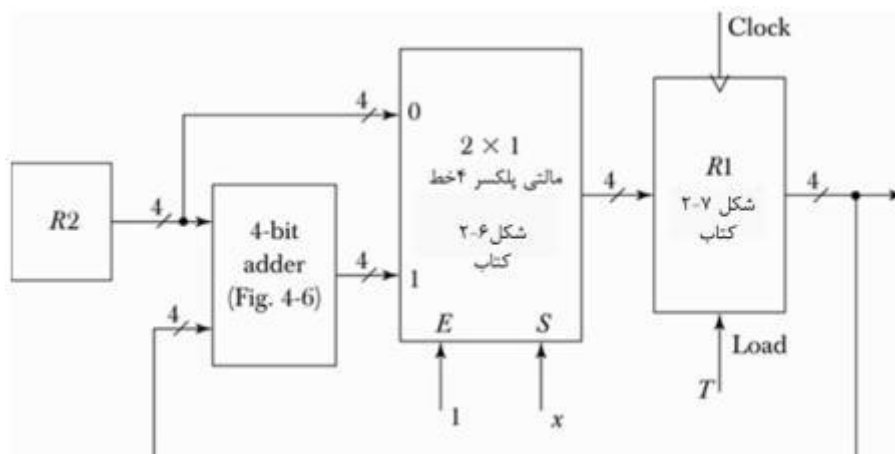
۱۰- عبارت انتقال ثباتی زیر را برای دو ثبات ۴ بیتی R_1 و R_2 در نظر بگیرید.

$$xT : R_1 \leftarrow R_1 + R_2$$

$$x'T : R_1 \leftarrow R_2$$

هر بار که $T=1$ است، یا محتوای R_2 به R_1 اضافه می‌شود اگر $x=1$ باشد یا R_2 به R_1 انتقال می‌یابد اگر $x=0$ باشد. دیاگرامی رسم کنید که پیاده‌سازی سخت‌افزاری دو عبارت را نشان دهد. برای دو ثبات ۴ بیتی، یک جمع‌کننده ۴ بیتی، و یک مولتی پلکسر چهار تایی ۲:۱ که ورودی‌ها به R_1 را انتخاب می‌کند از بلاک دیاگرام استفاده کنید. در نمودار نشان دهید که متغیرهای کنترلی x و T چگونه ورودی‌های مولتی پلکسر و ورودی بار کردن ثبات R_1 را انتخاب می‌کنند.

حل:



در ارتباط با Mux استفاده شده، ۴ Mux 2×1 که برای انتخاب هر کدام از ۴ بیت ورودی‌ها به صورت همزمان استفاده می‌شود و یک Quad Mux را تشکیل می‌دهد.

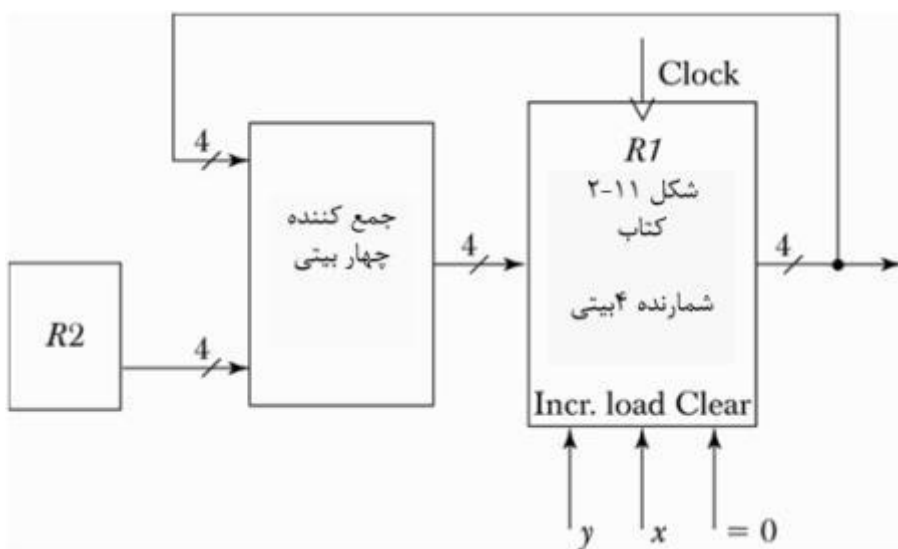
۱۱- با یک شمارنده ۴ بیتی با بار شدن موازی مانند شکل ۱۱-۲ و یک جمع‌کننده ۴ بیتی مانند شکل ۶-۴، یک بلاک دیاگرام رسم کرده و چگونگی پیاده‌سازی عبارت‌های زیر را نشان دهید.

$$x: R_1 \leftarrow R_1 + R_2 \quad R_2 \text{ را به } R_1 \text{ اضافه کن}$$

$$x'y = R_1 \leftarrow R_1 + 1 \quad R_1 \text{ را افزایش بده}$$

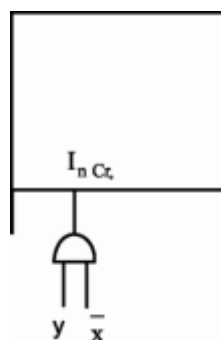
که R_1 یک شمارنده با بار شدن موازی و R_2 یک ثابت ۴ بیت است.

حل:



نکته مهم در این سؤال آن است که فرض شده است y و x طبق طراحی Ctrl Unit، همزمان ۱ نمی شوند. اگر این فرض را در نظر بگیریم، می بایست ورودی Increment شمارنده را به شکل زیر طراحی کنیم.

$$\text{Incr.} \leftarrow y\bar{x}$$



۱۲- مدار جمع - تفریق کننده شکل ۷-۴ دارای مقدارهای زیر برای ورودی مد

M و ورودی‌های داده A و B است، در هر حالت، مقادارهای خروجی $C_4, S_4, S_0, S_1, S_2, S_3$ را تعیین کنید.

	M	A	B
الف	۰	۰۱۱۱	۰۱۱۰
ب	۰	۱۰۰۰	۱۰۰۱
ج	۱	۱۱۰۰	۱۰۰۰
د	۱	۰۱۰۱	۱۰۱۰
هـ	۱	۰۰۰۰	۰۰۰۱

حل: طبق مدار جمع - تفریق‌کننده، در $M=0$ عمل جمع و در $M=1$ عمل تفریق انجام می‌پذیرد. پس:

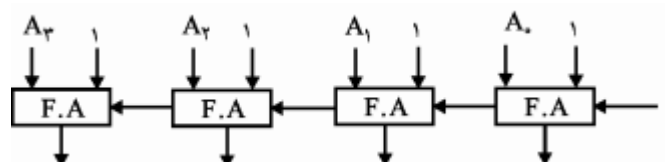
M	A	B	خروجی مجموع	Carry
۰	۰۱۱۱ + ۰۱۱۰		۱۱۰۱	۰
				$7+6=13$
۰	۱۰۰۰ + ۱۰۰۱		۰۰۰۱	۱
				$8+9=17$
۱	۱۱۰۰ - ۱۰۰۰		۰۱۰۰	۱
				$12-8=4$
۱	۰۱۰۱ - ۱۰۱۰		۱۰۱۱	۰
				$5-10=-5$
۱	۰۰۰۰ - ۰۰۰۱		۱۱۱۱	۰
				$0-1=-1$

۱۳- یک مدار ترتیبی کاهنده چهار بیتی را با چهار مدار تمام جمع‌کننده بسازید.

حل: برای اینکه خروجی $A-1$ را از مدار شامل ۴ F.A حاصل کنیم می‌توانیم به روش زیر عمل کنیم:

$$A-1 = A + (1) \text{ مکمل } ۲$$

بنابراین مدار به شکل زیر خواهد بود:



Carry in

خروجی کاهش گر ۴ بیت

۱۴- فرض کنید که مدار چهار بیتی شکل ۹-۴ در یک مدار مجتمع قرار گرفته باشد. اتصالات لازم برای دو IC از این نوع را برای ساختن یک مدار حسابی ۸ بیتی نشان دهید.

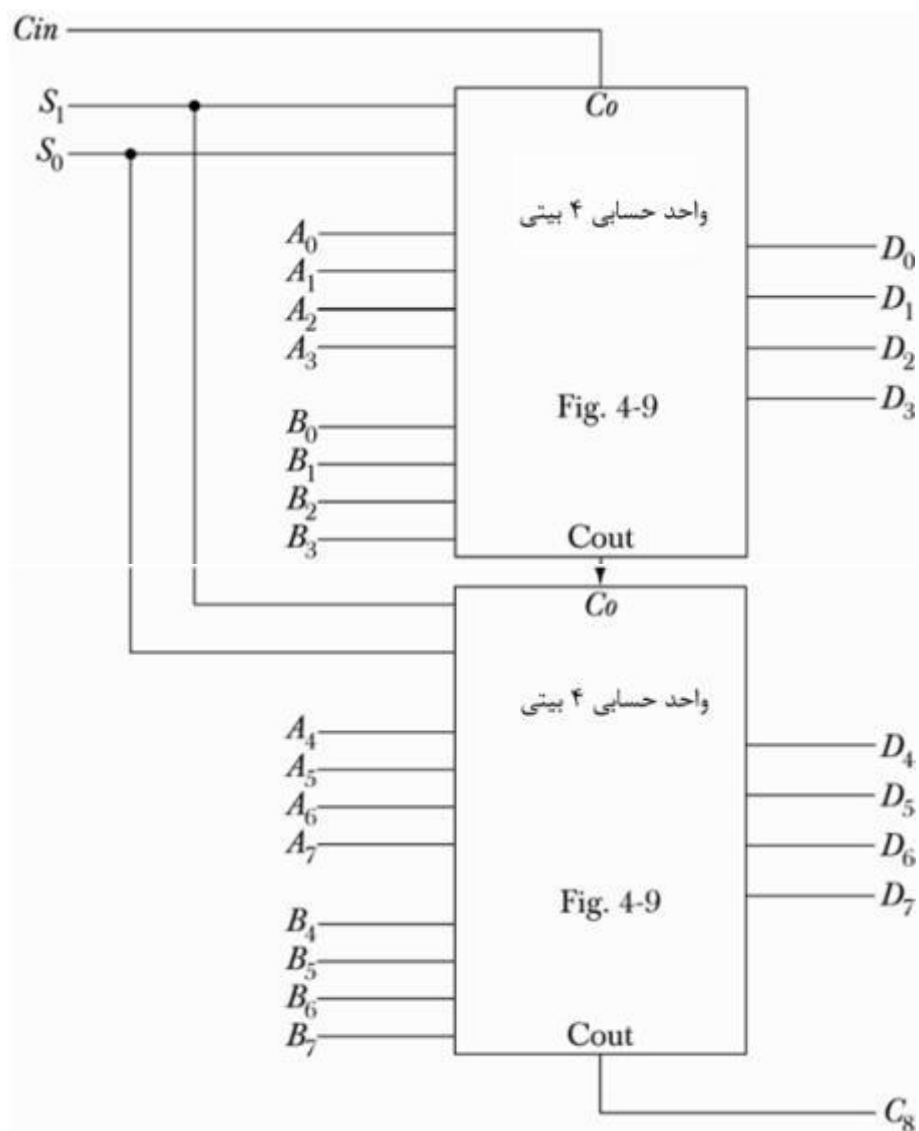
حل: برای گسترش از ۴ بیت به ۸ بیت مدار حسابی، موارد زیر انجام می شود:

۱- اتصال رقم نقلی خروجی IC اول به رقم نقلی ورودی IC دوم

۲- اتصال S_0 و S_1 به انتخاب کننده های هر دو گروه MUX ها در دو IC

۳- اتصال رقم نقلی ورودی به رقم نقلی ورودی IC اول

پس:



۱۵- یک مدار حسابی با یک متغیر انتخاب S و دو خط ورودی n بیتی A و B طراحی کنید. مدار چهار عمل حسابی زیر را با توجه به نقلی ورودی C_{in} تولید می‌کند. بلاک دیاگرام را برای دو مرحله اول مدار رسم کنید.

S	$C_{in} = 0$	$C_{in} = 1$
-----	--------------	--------------

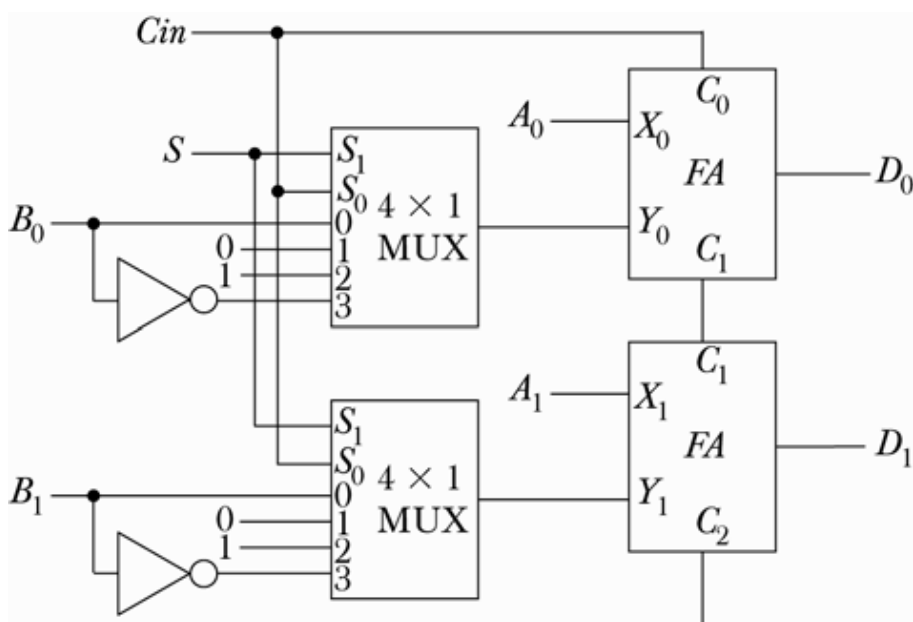
۰	$D = A + B$ (جمع)	$D = A + 1$ (افزایش)
۱	$D = A - 1$ (کاهش)	$D = A + \bar{B} + 1$ (تفریق)

حل: طبق موارد گفته شده، جدول درستی را تشکیل می‌دهیم به قسمی که وضعیت

دو پایه S و C_{in} مشخص کند که چه ورودی‌هایی به F.A وصل شوند.

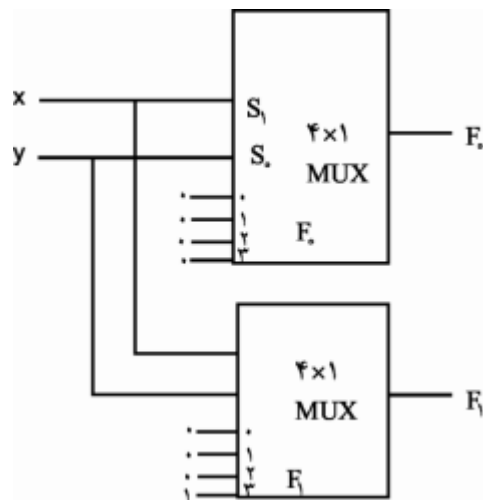
S	C_{in}	Y_1	Y_2	
۰	۰	A	B	(A+B) جمع
۰	۱	A	۰	A+1 افزایش
۱	۰	A	۱	A-1 کاهش
۱	۱	A	\bar{B}	A-B تفریق

پس Y_1 همیشه A و Y_2 طبق نتایج بالا از طریق MUX به F.A داده می‌شود. پس:



۱۶- مدار ترکیبی که هر یک از ۱۶ تابع منطقی جدول ۵-۴ را انتخاب و تولید می کند رسم کنید.

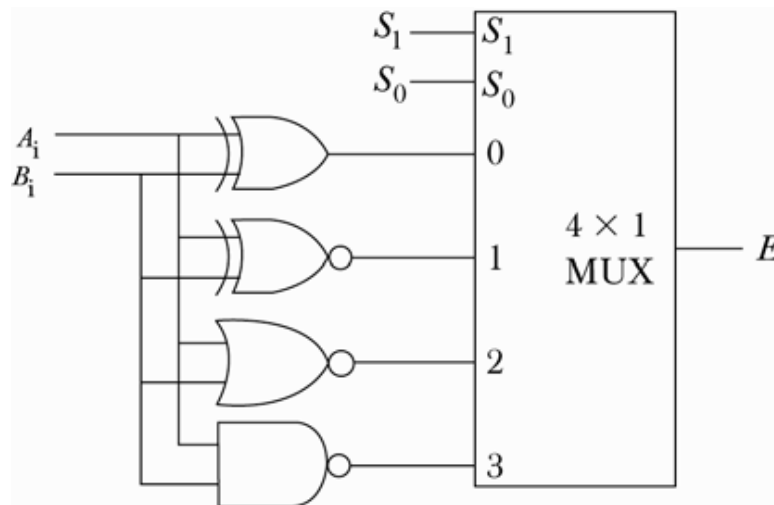
حل: با توجه به جدول، می توان با استفاده از 4×1 MUX، هر کدام از F ها را با توجه به ورودی x و y تعیین کرد. برای نمونه، قسمت تولید F_0 و F_1 مدار به شکل زیر خواهد بود:



بقیه F ها ($F_2 - F_{15}$) نیز به همین ترتیب ساخته خواهند شد؛ یعنی ورودی MUXهای مربوط به آنها همان x و y برای انتخاب و مقادیر ایستای F_i در جدول ۵-۴ کتاب است.

۱۷- یک مدار دیجیتال که چهار عمل منطقی OR انحصاری، NOR انحصاری، NAND و NOR را انجام دهد طراحی کنید. دو متغیر انتخاب بکار ببرید. دیاگرام منطقی یک مرحله نمونه را نشان دهید.

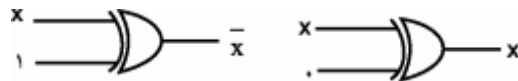
حل:



۱۸- ثبات A، عدد ۸ بیتی ۱۱۰۱۱۰۰۱ را نگهداری می‌کند. عملوند B و ریزعمل منطقی لازم را برای تغییر A به هر یک از مقادیرهای زیر معین کنید.

الف) ۰۱۱۰۱۱۰۰۱ ب) ۱۱۱۱۱۱۰۱

حل: در این گونه مسائل با استفاده از آزمون و خطا می‌توان عملوند دوم و عملگر مورد نظر را یافت. اما با استفاده از ریزعمل منطقی XOR، می‌توان عملوند دوم را به صورت دقیق پیدا کرد. به طوری که



بنابراین:

$\begin{array}{r} A \quad 11011001 \\ B \quad 00100100 \\ \hline A \oplus B \quad 11111101 \end{array}$	(ب) XOR	$\begin{array}{r} A = 11011001 \\ \text{XOR } B = 10110100 \\ \hline A \oplus B \quad 01101101 \end{array}$
---	---------	---

$$\begin{array}{r} A = 11011001 \quad \text{OR} \\ B = 11111101 \\ \hline A \vee B \quad 11111101 \end{array}$$

۱۹- ثبات‌های ۸ بیتی AR, BR, CR, DR ابتدا دارای مقدارهای زیراند.

AR=11110010

BR=11111111

CR=10111001

DR=11101010

محتوای هشت بیتی هر یک از ثبات‌ها را پس از اجرای هر یک از رشته ریزعمل‌ها مشخص کنید.

$AB \leftarrow AR + BR$

$CR \leftarrow CR \wedge DR, BR \leftarrow BR + 1$

$AR \leftarrow AR - CR$

حل:

(الف)

AR = 11110010

BR = 11111111 +

AR \leftarrow 11110001, BR=11111111, CR = 10111001

DR = 11101010

(ب)

CR=10111001

DR=11101010 (AND) BR=11111111

CR \leftarrow 10101000, +1

BR \leftarrow 00000000, AR=11110001

, DR=11101010

(ج)

AR=11110001

CR=10101000 -

AR \leftarrow 01001001, BR=00000000, CR = 10101000

, DR=11101010

۲۰- یک ثبات هشت بیتی دارای عدد دودویی ۱۰۰۱۱۰۰ است. پس از یک

شیفت به راست حسابی مقدار ثبات چقدر است؟ با شروع از مقدار اولیه

۱۰۰۱۱۱۰۰، مقدار ثبات را پس از یک شیفت به چپ حسابی معین کنید، و بگویید

آیا سرریز وجود دارد؟

حل:

الف) شیفت به راست حسابی ۱۱۰۰۱۱۱۰ ArithMetic Shift right

ب) شیفت به چپ حسابی ۰۰۱۱۱۰۰۰ ArithMetic shift left

سرریز اتفاق افتاده است. زیرا عدد منفی به عدد مثبت تبدیل شده است (به دلیل تغییر بیت علامت)

۲۱- با شروع از مقدار اولیه $R = 11011101$ دنباله مقادیرهای دودویی را در R پس از یک شیفت به چپ منطقی و به دنبال آن یک شیفت به راست چرخشی، و سپس با یک شیفت به راست منطقی و در پایان یک شیفت به چپ چرخشی تعیین کنید.

حل: مقدار اولیه: ۱۱۰۱۱۱۰۱

- ۱- شیفت به چپ منطقی ۱۰۱۱۱۰۱۰
- ۲- شیفت به راست چرخشی ۰۱۰۱۱۱۰۱
- ۳- شیفت به راست منطقی ۰۰۱۰۱۱۱۰
- ۴- شیفت به چپ چرخشی ۰۱۰۱۱۱۰۰

۲۲- مقدار H در شکل ۱۲-۴ چیست به شرطی که $I_L = 0, I_R = 1, S = 1, A = 1001$ باشد.

حل: طبق مدار شیفت،

$$S=1 \Rightarrow \begin{matrix} H_0 & H_1 & H_2 & H_3 \\ A_1 & A_2 & A_3 & I_L \end{matrix}$$

بنابراین شیفت به چپ به شکل مقابل خواهد بود.

$$H=10010 \Rightarrow 0010$$

شیفت به چپ

۲۳- چه چیزی در عبارات انتقال ثبات زیر نادرست است؟

الف) $xT : AR \leftarrow \overline{AR}, AR \leftarrow 0$

ب) $yT : R_1 \leftarrow R_2, R_1 \leftarrow R_3$

ج) $zT : PC \leftarrow AR, PC \leftarrow PC+1$

حل: توجه شود که علامت « و » به معنی اجرای همزمان است. پس در رابطه با

عبارت (الف)، نمی توان یک رجیستر را همزمان مکمل و یا صفر کرد.

در رابطه با عبارت (ب)، نمی توان دو مقدار را در یک رجیستر به صورت همزمان

انتقال داد.

در رابطه با عبارت (ج)، نمی توان همزمان رجیستر PC را افزایش داد و مقدار AR را

در آن قرار داد.

فصل پنجم: سازمان و طراحی یک کامپیوتر پایه

۱- یک کامپیوتر از حافظه‌ای با ۲۵۶K کلمه ۳۲ بیتی استفاده می‌کند. یک دستورالعمل دودویی در یک کلمه از حافظه ذخیره شده است. دستورالعمل چهار بخش دارد. بیت غیرمستقیم، یک کد عملیاتی، یک کد ثبات برای تعیین یکی از ۶۴ ثبات و بخش آدرس.

الف) چند بیت در کد عملیاتی، کد ثبات و آدرس وجود دارد:

$$18 \text{ بیت برای آدرس حافظه } 256K = 2^8 \times 2^{10} = 2^{18}$$

$$6 \text{ بیت برای آدرس‌دهی رجیسترها } 64 = 2^6$$

۱ بیت غیرمستقیم : indirect

$$7 = 32 - (18 + 6 + 1) \text{ بیت برای کد عملیاتی}$$

ب) قالب کلمه دستورالعمل را ترسیم و تعداد بیت در هر قسمت را معین کنید.

$$\text{بیت } 32 = 18 + 6 + 1 + 7$$

I	opcode	Register	Address
---	--------	----------	---------

پ) ورودی‌های داده و آدرس حافظه چند بیت وجود دارد؟

۳۲ بیت برای ورودی داده و ۱۸ بیت برای ورودی آدرس حافظه.

۲- تفاوت بین دستور با آدرس مستقیم و غیرمستقیم چیست؟ چند دسترسی به حافظه برای هر نوع دستورالعمل لازم است تا عملوند را به ثبات پردازشگر منتقل کند؟
حل:

آدرس‌دهی غیرمستقیم در صورت بیشتر بودن تعداد بیت داده از تعداد بیت آدرس حافظه در دستورالعمل می‌تواند امکان آدرس‌دهی فضای بزرگ‌تری از حافظه را به ما بدهد. (مثلاً ۱۶ بیت آدرس به جای ۱۲ بیت). همچنین نیاز به دستور برای بدست آوردن عملوند را از بین می‌برد و در یک دستور این کار را انجام می‌دهد. (این کار به

خصوص هنگام استفاده از اشاره گرها در زبان برنامه نویسی مفید است.)

۲- دستور با آدرس مستقیم دو دسترسی به حافظه نیاز دارد: خواندن دستورالعمل و خواندن عملوند

۳- دستور با آدرس غیرمستقیم سه دسترسی به حافظه نیاز دارد: خواندن دستورالعمل، خواندن آدرس مؤثر و خواندن عملوند.

۳- ورودی های کنترل زیر در سیستم گذرگاه شکل ۴-۵ فعالند. برای هر مورد، انتقال ثباتی را کد در پالس ساعت بعدی اجرا می شود تعیین کنید.
حل:

جمع کننده	حافظه	LD ثبات	S_0	S_1	S_2
—	خواندن	IR	۱	۱	۱
—	—	PC	۰	۱	۱
—	نوشتن	DR	۰	۰	۱
جمع	—	AC	۰	۰	۰

الف) خواندن مقدار حافظه بر روی گذرگاه و بار شدن آن در IR

$$IR \leftarrow M[AR]$$

ب) انتقال مقدار TR به PC به واسطه گذرگاه

$$PC \leftarrow TR$$

ج) قرار گرفتن مقدار AC روی گذرگاه، نوشتن مقدار در حافظه و بار شدن آن در

$$DR \leftarrow AC, M[AR] \leftarrow AC, DR \leftarrow AC$$

د) جمع کردن مقدار DR با AC : $AC \leftarrow AC + DR$

۴- انتقال ثبات های زیر قرار است در سیستم شکل ۴-۵ اجرا شوند. برای هر

انتقال: (۱) مقدار دودویی که باید به ورودی های انتخاب گذرگاه S_0 و S_1 , S_2

اعمال شود را تعیین کنید؛ (۲) ثباتی که کنترل LD آن باید فعال شود (اگر وجود

دارد)؛ (۳) عمل نوشتن یا خواندن حافظه (اگر نیاز است)؛ و (۴) عمل در

جمع کننده و مدار منطقی (اگر وجود دارد).

ب) $IR \leftarrow N[AR]$

الف) $AR \leftarrow PC$

د) $AC \leftarrow DR, DR \leftarrow AC$

ج) $M[AR] \leftarrow TR$

حل:

	(1) <u>$S_2 S_1 S_0$</u>	(2) <u>Load(LD)</u>	(3) <u>Memory</u>	(4) <u>Adder</u>
(a) $AR \leftarrow PC$	010 (PC)	AR	—	—
(b) $IR \leftarrow M[AR]$	111 (M)	IR	Read	—
(c) $M[AR] \leftarrow TR$	110 (TR)	—	Write	—
(d) $DR \leftarrow AC$ $AC \leftarrow DR$	100 (AC)	DR and AC	—	Transfer DR to AC

توضیح قسمت (د): توجه کنید که این دو عمل همزمان می توانند انجام پذیرند چون یکی از گذرگاه استفاده کرده و دیگری نیاز به گذرگاه ندارد. همچنین با توجه به حساس بر لبه بودن ثبات ها و در نظر گرفتن زمان انتشار مقدارهای نامعتبر (در اثر تغییر مقدار همزمان) در ثبات ها ثبت نمی شود.

۵ - توضیح دهید که چرا هیچ یک از ریز عمل های زیر نمی تواند در طول یک پالس ساعت در سیستم شکل ۴-۵ اجرا شود. رشته ریز عمل های لازم برای انجام عمل را تعیین کنید.

الف) $IR \leftarrow M[PC]$: PC مستقیماً نمی تواند آدرس را برای حافظه فراهم کند، آدرس باید به AR منتقل شود:

$AR \leftarrow PC$

$IR \leftarrow M[AR]$

ب) $AC \leftarrow AC + TR$ عمل جمع باید با DR انجام شود. TR باید به DR منتقل شود.

$DR \leftarrow TR$

$AC \leftarrow AC + DR$

ج) (AC تغییر نمی‌کند) $DR \leftarrow DR + AC$

حاصل جمع به AC منتقل می‌شود (نه DR). همچنین برای نگه داشتن مقدار AC، باید آن را به صورت موقت در DR (یا TR) ذخیره کنیم: (با توجه به قسمت (د) سؤال (۴))

$AC \leftarrow DR, DR \leftarrow AC$

$AC \leftarrow AC + DR$

$AC \leftarrow DR, DR \leftarrow AC$

۶- قالب های دستور کامپیوتر پایه شکل ۵-۵ و لیست دستورهای جدول ۵-۲

را در نظر بگیرید. برای هر یک از دستورهای ۱۶ بیتی، کد معادل چهار رقمی مبنای شانزده را نوشته و به زبان ساده بگویید این دستور چه کاری انجام می‌دهد.

(الف)

$$\frac{0001}{ADD} \frac{0000}{(024)_{16}} \frac{0010}{(024)_{16}} \frac{0100}{(024)_{16}} = (1024)_{16}$$

جمع مقدار خانه $M[024]$ با AC (ADD 024)

(ب)

$$\frac{1}{I} \frac{011}{ST} \frac{0001}{(124)_{16}} \frac{0010}{(124)_{16}} \frac{0100}{(124)_{16}} = (B124)_{16}$$

ذخیره $AC = M[M[124]]$ (STAI 124)

(ج)

$$\frac{0111}{Register} \frac{0000}{(INC) AC} \frac{0010}{افزایش} \frac{0000}{(7020)_{16}} = (7020)_{16}$$

۷- کدام دو دستور برای ۱ کردن فلیپ فلاپ E در کامپیوتر پایه به کار

می‌روند؟

حل:

برای ۱ کردن فلیپ فلاپ E ابتدا باید آن را پاک کنیم و سپس آن را متمم کنیم:

CLE پاک کردن E

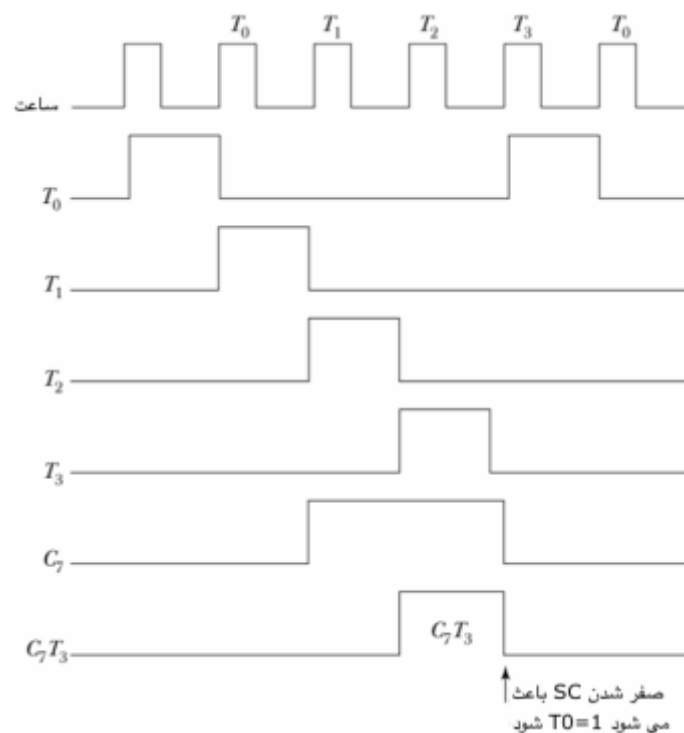
CME متمم کردن E

۸- یک نمودار زمان‌بندی مشابه شکل ۵-۷ رسم کنید و فرض کنید SC در T_3 برابر ۰ شود به شرط اینکه سیگنال کنترل C_7 فعال باشد.

C_7T_3 : $SC \leftarrow 0$

C_7 با لبه پالس مربوط به T_1 فعال می‌شود.

حل:



۹- محتوای AC در کامپیوتر پایه عدد مبنای شانزده A937 است و مقدار اولیه E

برابر ۱ است. محتوای AC، E، PC، AR و IR در مبنای ۱۶ پس از اجرای دستور

CLA چیست؟ عمل قبل را ۱۱ بار با هر یک از دستورات عمل‌های دسترسی به ثبات

تکرار کنید. مقدار اولیه PC را عدد مبنای شانزده ۰۲۱ فرض کنید.

حل:

	E	AC	PC	AR	IR
مقدار اولیه	1	A937	021	—	—
CLA	1	0000	022	800	7800
CLE	0	A937	022	400	7400
CMA	1	56C8	022	200	7200
CME	0	A937	022	100	7100
CIR	1	D49B	022	080	7080
CIL	1	526F	022	040	7040
INC	1	A938	022	020	7020
SPA	1	A937	022	010	7010
SNA	1	A937	023	008	7008
SZA	1	A937	022	004	7004
SZE	1	A937	022	002	7002
HLT	1	A937	022	001	7001

دقت کنید که در دستور SNA چون AC منفی است و شرط برقرار است یک واحد به مقدار PC اضافه شده است. (دستور بعد اجرا نمی‌شود).

۱۰- دستورالعملی در آدرس ۰۲۱ کامپیوتر پایه دارای $I=0$ ، کد عملیاتی AND و آدرس ۰۸۳ است (تمام ارقام در مبنای ۱۶ است). کلمه حافظه واقع در آدرس ۰۸۳ دارای عملوند B8F2 و محتوای AC هم A937 است. در طول سیکل دستور محتوای ثبات‌های زیر را در پایان فاز اجرا تعیین کنید: PC, DR, AR, AC و IR. مسأله را شش بار برای دستورالعمل دسترسی به حافظه دیگری تکرار کنید.

حل:

	PC	AR	DR	AC	IR
مقدار اولیه	021	—	—	A937	—
AND	022	083	B8F2	A832	0083
ADD	022	083	B8F2	6229	1083
LDA	022	083	B8F2	B8F2	2083
STA	022	083	—	A937	3083
BUN	083	083	—	A937	4083
BSA	084	084	—	A937	5083
ISZ	022	083	B8F3	A937	6083

۱۱- محتوای ثبات‌های PC, AR, DR, IR و SC در مبنای شانزده کامپیوتر پایه

وقتی که دستور غیرمستقیم ISZ دریافت و اجرا شود چیست؟ مقدار اولیه PC را 7FF در نظر بگیرید. محتویات حافظه در آدرس 7FF برابر EA9F می باشد. محتویات حافظه در آدرس A9F هم 0C35 می باشد. محتویات حافظه C35 برابر FFFF می باشد. پاسخ را به صورت جدولی با ۵ ستون با هر ستون برای یک ثبات، و هر سطر برای یک سیگنال زمان بندی تهیه کنید. محتوای هر ثبات را پس از لبه مثبت هر پالس ساعت نشان دهید.

حل:

	PC	AR	DR	IR	SC
مقدار اولیه	7FF	—	—	—	0
T ₀	7FF	7FF	—	—	1
T ₁	800	7FF	—	EA9F	2
T ₂	800	A9F	—	EA9F	3
T ₃	800	C35	—	EA9F	4
T ₄	800	C35	FFFF	EA9F	5
T ₅	800	C35	0000	EA9F	6
T ₆	801	C35	0000	EA9F	0

۱۲- محتوای PC در کامپیوتر پایه 3AF است (تمام اعداد در مبنای شانزده). محتویات AC هم 7EC3 است. محتوای حافظه آدرس 3AF برابر 932E می باشد. محتوای حافظه در آدرس 32E برابر 09AC و در آدرس 9AC هم 8B9F است. الف) دستورالعملی که در گام بعد دریافت و اجرا می شود چیست؟ حل: I و ریز عمل از روی بیت ها مشخص می شود.

$$9 = (1001)$$

$$\Rightarrow \text{ADD I } 32E$$

$$I=1 \quad \text{ADD}$$

حافظه	
3AF	932E
32E	09AC
9AC	8B9F

ب) عمل دودویی که در AC پس از اجرای دستورالعمل رخ می‌دهد چیست؟

$$AC = 7EC3 \text{ (جمع)}$$

$$DR = 8B9F$$

$$0A62 \text{ (E=1)}$$

ج) محتوای ثبات‌های IR, AC, DR, AR, PC در مبنای شانزده چیست؟

همچنین مقدار E و I و SC در پایان سیکل دستورالعمل را تعیین کنید.

$$PC = 3AF + 1 = 3B0$$

$$AR = 9AC$$

$$IR = 932E$$

$$DR = 8B9F$$

$$E = 1$$

$$AC = 0A62$$

$$I = 1$$

$$SC = 0000$$

۱۳- فرض کنید که شش دستور دسترسی به حافظه در کامپیوتر پایه در جدول

۵-۴ با جدول زیر تعویض شوند. EA آدرس مؤثر واقع در AR در T_4 است. فرض

کنید که جمع‌کننده و مدار منطقی شکل ۵-۴ می‌تواند عمل XOR را انجام دهد.

$AC \leftarrow AC \oplus DR$. بعلاوه فرض کنید که جمع‌کننده و مدار منطقی نمی‌توانند

مستقیماً تفریق را انجام دهند. تفریق باید به کمک مکمل ۲ انجام شود. دنباله

عبارت انتقال ثباتی لازم برای اجرای هر دستور لیست شده را از T_4 به بعد

مشخص کنید. دقت کنید که هیچ تغییری در AC نباید رخ دهد مگر اینکه

دستورالعمل تغییری را در آن معین کند. شما می‌توانید با استفاده از TR محتویات

AC را موقتاً ذخیره و یا محتویات AC و DR را با هم جابجا کنید.

توضیح	نمایش سمبلیک	کد عمل	سمبل
OR انحصاری با AC	$AC \leftarrow AC \oplus M[EA]$...	XOR

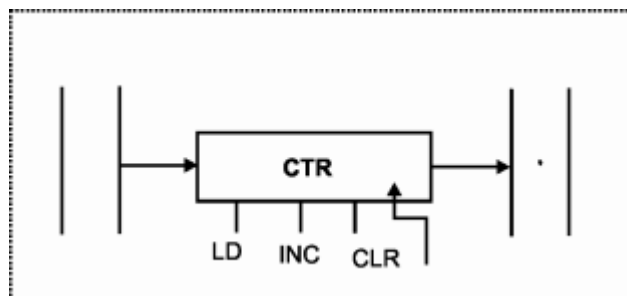
ADM	۰۰۱	$M[EA] \leftarrow M[EA] + AC$	جمع AC با حافظه
SUB	۰۱۰	$AC \leftarrow AC - M[EA]$	تفریق حافظه از AC
XCH	۰۱۱	$AC \leftarrow M[EA], M[EA] \leftarrow AC$	تبادل AC با حافظه
SEQ	۱۰۰	If ($M[EA]=AC$) then $PC \leftarrow PC+1$	گذر در صورت برابری
BPA	۱۰۱	if ($AC > 0$) then ($PC \leftarrow EA$)	انشعاب اگر AC مثبت و غیر صفر باشد.

حل:

<u>XOR</u>	$D_0T_4 :$	$DR \leftarrow M[AR]$
	$D_0T_5 :$	$AC \leftarrow AC \oplus DR, SC \leftarrow 0$
<u>ADM</u>	$D_1T_4 :$	$DR \leftarrow M[AR]$
	$D_1T_5 :$	$DR \leftarrow AC, AC \leftarrow AC + DR$
	$D_1T_6 :$	$M[AR] \leftarrow AC, AC \leftarrow DR, SC \leftarrow 0$
<u>SUB</u>	$D_2T_4 :$	$DR \leftarrow M[AR]$
	$D_2T_5 :$	$DR \leftarrow AC, AC \leftarrow DR$
	$D_2T_6 :$	$AC \leftarrow \overline{AC}$
	$D_2T_7 :$	$AC \leftarrow AC + 1$
	$D_2T_8 :$	$AC \leftarrow AC + DR, SC \leftarrow 0$
<u>XCH</u>	$D_3T_4 :$	$DR \leftarrow M[AR]$
	$D_3T_5 :$	$M[AR] \leftarrow AC, AC \leftarrow DR, SC \leftarrow 0$
<u>SEQ</u>	$D_4T_4 :$	$DR \leftarrow M[AR]$
	$D_4T_5 :$	$TR \leftarrow AC, AC \leftarrow AC \oplus DR$
	$D_4T_6 :$	If ($AC = 0$) then ($PC \leftarrow PC + 1$), $AC \leftarrow TR, SC \leftarrow 0$
<u>BPA</u>	$D_5T_4 :$	If ($AC = 0 \wedge AC(15) = 0$) then ($PC \leftarrow AR$), $SC \leftarrow 0$

۱۴- تغییرهای زیر را در کامپیوتر پایه انجام دهید.

۱- یک ثبات CTR (ثبات شمارنده) را به سیستم گذرگاه اضافه کنید و آن را با

 $S_2 S_1 S_0 = 000$ انتخاب نمایید.

۲- ISZ را با دستوری که یک عدد را در CTR بار می کند جایگزین کنید.

LDC Address : $CTR \leftarrow M[Address]$

$D_6 T_4$: $CTR \leftarrow M[AR]$, $SC \leftarrow 0$

۳- یک دستور دسترسی به ثبات ICSZ اضافه کنید: CTR را یک واحد افزایش داده و از اجرای دستور بعدی اگر حاصل افزایش صفر باشد صرف نظر نماید. در مورد فایده این تغییر بحث کنید.

ICSZ $D_7 I' T_3 B_{12}$: $CTR \leftarrow CTR + 1$

$D_7 I' T_4 B_{12}$: if $(CTR = 0)$ then $(PC \leftarrow PC + 1)$ $SC \leftarrow 0$

این کار دستور ISZ را که دسترسی به حافظه است به دستور ICSZ که دسترسی به ثبات است تبدیل می کند. دستور جدید ICSZ می تواند در زمان T_4 به جای T_6 اجرا شود که باعث هدر رفتن ۲ سیکل می شود و روش دیگر پیاده سازی دستور ICSZ به این صورت است:

ICSZ: $D_7 I' T_3 B_{12}$: $CTR \leftarrow CTR + 1$, if $(CTR = FFFF)$ then $PC \leftarrow PC + 1$, $SC \leftarrow 0$

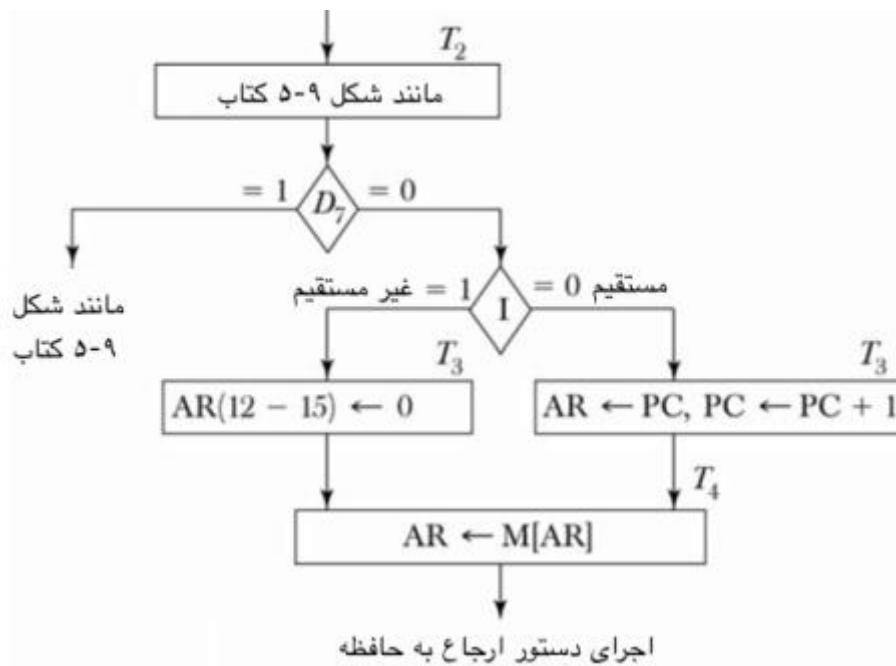
مزیت این پیاده سازی این است که در زمان T_3 اجرا می شود و از هدر رفتن ۳ سیکل ساعت جلوگیری می کند. همچنین همانند دستورهای دسترسی به ثبات دیگر، در T_3 اجرای آن تمام می شود.

توجه داشته باشید که برای اضافه کردن دستور ICSZ باید یکی از دستورات دسترسی به ثبات را حذف کنیم یا ۱ بیت به کد دستورالعمل اضافه کنیم یا نحوه نمایش

دستورالعمل‌های دسترسی به ثبات را تغییر دهیم که نیاز به دیکد کردن خواهد داشت.

۱۵- واحد حافظه کامپیوتر پایه در شکل ۱۳-۵ را با یک حافظه 16×65536 جایگزین کنید. این حافظه آدرس ۱۶ بیتی نیاز دارد. قالب دستورالعمل دسترسی به حافظه در شکل (۵-۵ الف) برای $I=1$ ثابت می ماند و بخش آدرس در مکان‌های ۰ تا ۱۱ قرار دارد. اما وقتی $I=0$ است (آدرس مستقیم) آدرس دستورالعمل با ۱۶ بیت در کلمه دیگری که به دنبال دستور آمده داده می شود. ریز عمل‌های زمان‌های T_2, T_3 (و T_4 اگر نیاز باشد) تغییر دهید تا با این پیکربندی هماهنگ باشد.

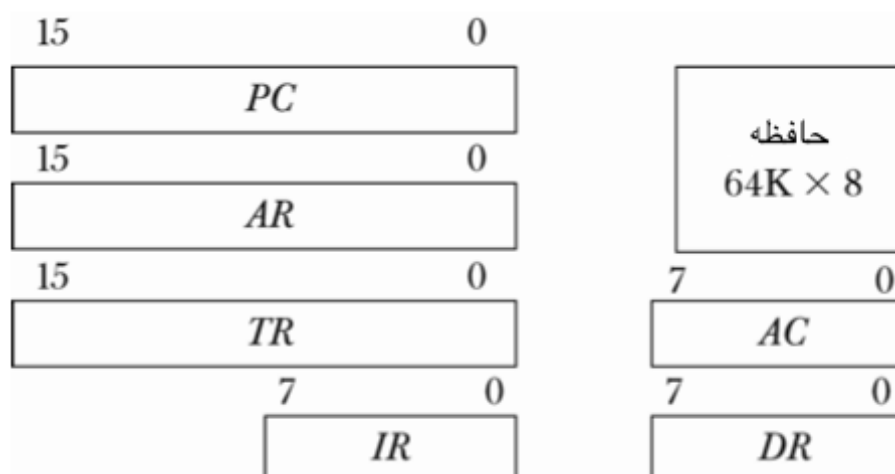
حل:



این طراحی از هدر رفتن T_3 در حالت آدرس‌دهی مستقیم جلوگیری کرده و از آن برای آدرس‌دهی ۱۶ بیتی استفاده کرده که حافظه‌ی بزرگ‌تری را پشتیبانی می‌کند.

۱۶- کامپیوتری از یک حافظه ۸ بیتی ۶۵۵۳۶ کلمه‌ای استفاده می‌کند. این کامپیوتر دارای ثابت‌های PC , AR , TR (هر یک ۱۶ بیت) و AC , DR و IR (هر یک ۸ بیت) است. یک دستور دسترسی به حافظه شامل سه کلمه است: یک کد عملیات ۸ بیتی (یک کلمه) و یک آدرس ۱۶ بیتی (در دو کلمه بعدی). تمام عملوندها هشت بیتی هستند. بیت غیرمستقیم هم وجود ندارد.

الف) بلاک دیاگرامی از کامپیوتر رسم کنید و ثابت‌ها و حافظه را همانند شکل ۵-۳ نشان دهید. (از یک گذرگاه مشترک استفاده نکنید.)



ب) جایگیری یک نمونه دستور سه کلمه‌ای را به همراه عملوند ۸ بیتی در حافظه نشان دهید.



پ) دنبایه ریز عمل ها برای دریافت یک دستور دسترسی به حافظه را لیست کنید و سپس عملوند را در DR قرار دهید. از سیگنال زمانی T_0 شروع کنید.

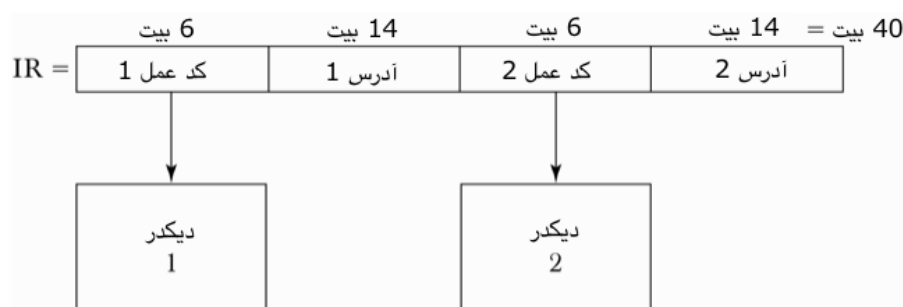
$T_0:$ $IR \leftarrow M(PC), PC \leftarrow PC + 1$

$T_1:$ $AR(0-7) \leftarrow M[PC], PC \leftarrow PC + 1$

$T_2:$ $AR(8-15) \leftarrow M[PC], PC \leftarrow PC + 1$

$T_3:$ $DR \leftarrow M[AR]$

۱۷- یک کامپیوتر دیجیتال دارای ۱۹۳۸۹ حافظه ۴۰ بیتی در هر کلمه است قالب کد دستور از شش بیت برای عملگر و ۱۴ بیت برای آدرس تشکیل شده است. (بیت غیرمستقیم ندارد) دو دستورالعمل در یک کلمه جای داده شده‌اند و یک ثابت دستورالعمل ۴۰ بیتی IR هم در واحد کنترل وجود دارد. برنامه‌ای را برای فازهای برداشت و اجرا در این کامپیوتر بنویسید.



برنامه برداشت و اجرا به این صورت است:

- ۱- دستور دوگانه ۴۰ بیتی را از حافظه خوانده و در IR قرار بده، همچنین مقدار PC را یک واحد افزایش بده.
- ۲- کد دستور ۱ را دیکد کن.
- ۳- دستور ۱ را با استفاده از آدرس ۱ اجرا کن.
- ۴- کد دستور ۲ را دیکد کن.
- ۵- دستور ۲ را با استفاده از آدرس ۲ اجرا کن.
- ۶- به گام اول باز گرد.

۱۸- یک برنامه خروجی از آدرس ۲۳۰۰ نوشته شده است. این برنامه وقتی کامپیوتر یک وقفه را در $FGO=1$ تشخیص دهد اجرا می‌شود. (در حالی که $IEN=1$ است).

الف) چه دستوری باید در آدرس ۱ قرار گیرد؟

انشعاب بدون شرط به خانه ۲۳۰۰

BUN 2300

ب) دو دستور آخر برنامه خروجی چیست؟ فعال کردن وقفه‌ها

انشعاب بدون شرط با آدرس غیرمستقیم 0

ION 0

BUN I ϕ

۱۹- عبارت انتقال ثباتی برای ثبات R و حافظه در یک کامپیوتر مطابق زیر

است (Xها توابع کنترل هستند و به طور تصادفی رخ می‌دهند).

$X'_3 X_1 : R \leftarrow M[AR]$ کلمه حافظه را در R بنویس

$X'_1 X_2 : R \leftarrow AC$ انتقال AC بر R

$X'_1 X_3 : M[AR] \leftarrow R$ R را در حافظه بنویس

حافظه دارای ورودی‌های داده، خروجی‌های داده، ورودی‌های آدرس و ورودی‌های

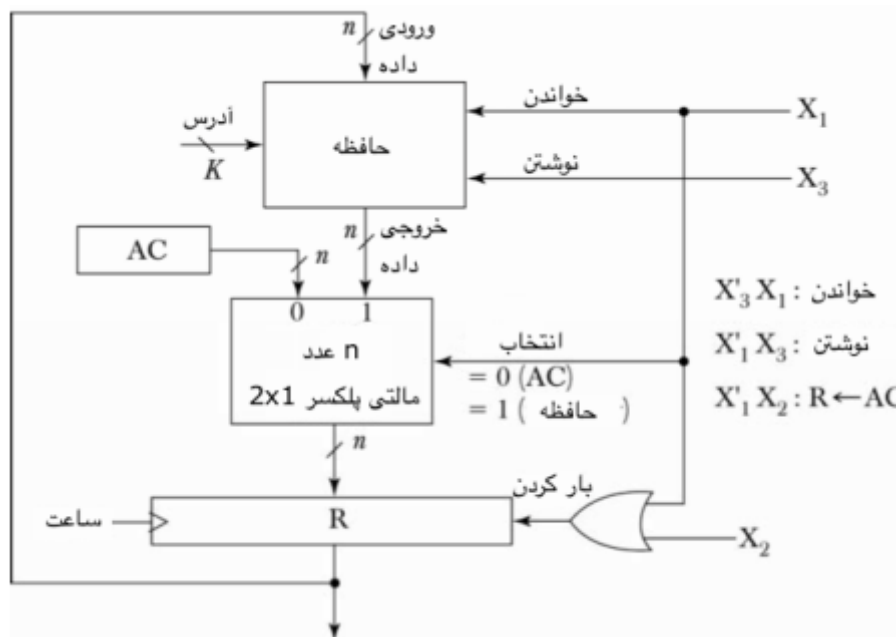
کنترل برای خواندن و نوشتن همانند شکل ۱۲-۲ است. پیاده سازی سخت‌افزاری R و

حافظه را به شکل بلاک دیاگرام بکشید. نشان دهید که چگونه توابع کنترل X_1 تا X_3

ورودی R، ورودی مولتی پلکس‌هایی که شما در دیاگرام وارد کرده‌اید، و ورودی‌های

خواندن و نوشتن حافظه را انتخاب می‌کنند.

حل:



۲۰- دنباله عمل‌هایی که باید روی فلیپ فلاپ F انجام شوند (در کامپیوتر پایه به کار نرفته) با عبارت انتقال ثباتی تعیین شده‌اند.

XT_3 : $F \leftarrow 1$ در F نشانده شود.

yT_1 : $F \leftarrow 0$ با 0 پاک شود.

ZT_2 : $F \leftarrow F$ مکمل F

WT_5 : $F \leftarrow G$ مقدار G را به F انتقال بده

در غیر این صورت محتوای F نباید تغییر یابد. دیاگرام منطقی مربوط به اتصال گیت‌هایی که توابع کنترل و ورودی‌های فلیپ فلاپ F را تشکیل می‌دهند رسم کنید. از یک فلیپ فلاپ JK استفاده کرده و تعداد گیت‌ها را حداقل کنید.

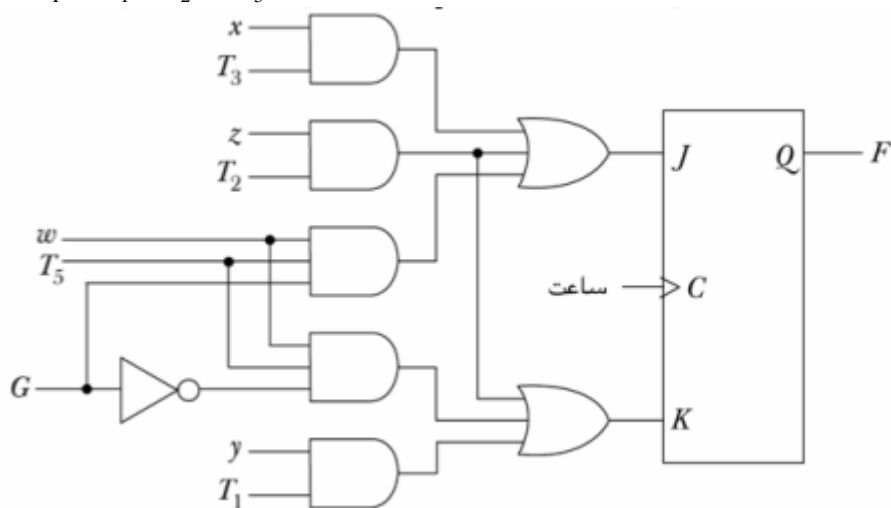
حل:

با توجه به جدول تحریک JK , J باید در حالت‌های set یا مکمل یا انتقال از G فعال باشد پس داریم:

$$J_F = XT_3 + ZT_2 + WT_5G$$

به همین ترتیب برای K_F داریم:

$$K_F = YT_1 + ZT_2 + WT_5G'$$



۲۱- مدار کنترل گیتی شمارنده برنامه PC را در کامپیوتر پایه بدست آورید.

از جدول ۵ - ۶ داریم:

$$(Z_{AC}=1 \text{ if } AC=0, \quad Z_{DR}=1 \text{ if } DR=0)$$

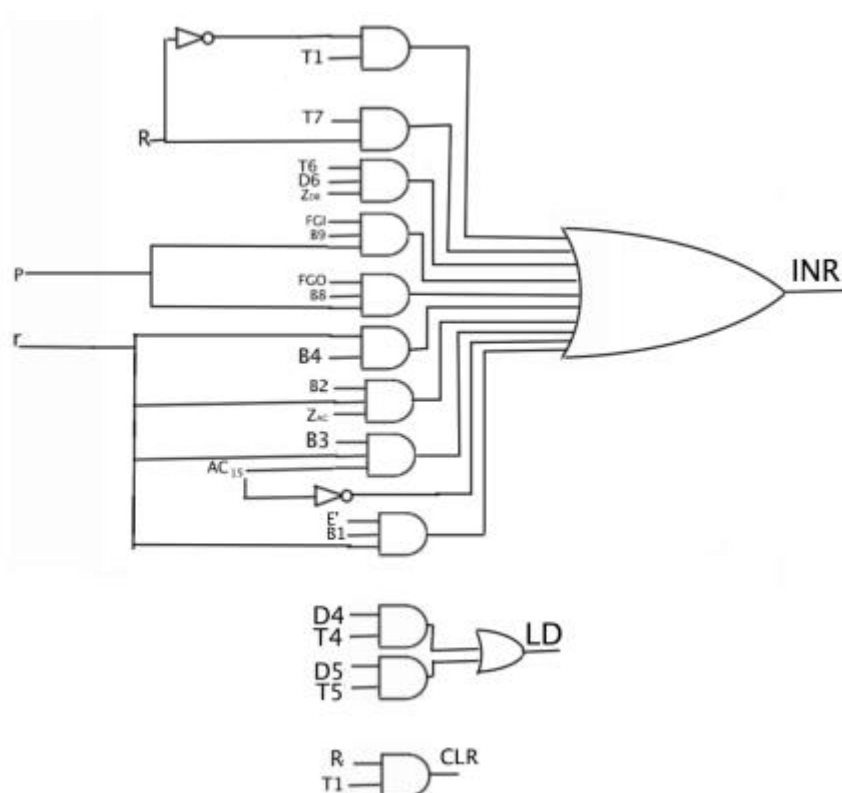
$$INC(PC) = R'T_1 + RT_2 + D_6T_6Z_{DR} + PB_9(FGI)$$

$$+ PB_8(FGO) + rB_4 + (AC_{15})' + rB_3(AC_{15})$$

$$+ rB_2Z_{AC} + rB_1E'$$

$$LD(PC) = D_4T_4 + D_5T_5$$

$$CLR(PC) = RT_1$$



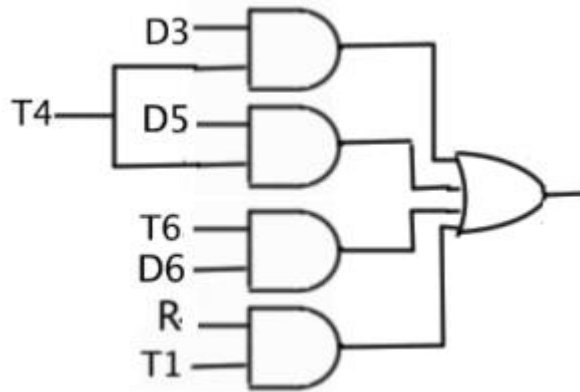
۲۲- مدار کنترل گیتی ورودی نوشتن در حافظه را در کامپیوتر پایه بدست آورید.

حل:

از جدول ۶ - ۵ داریم:

$$(M[AR] \leftarrow XX)$$

$$\text{Write} = D_3 T_4 + D_5 T_4 + D_6 T_6 + R T_1$$



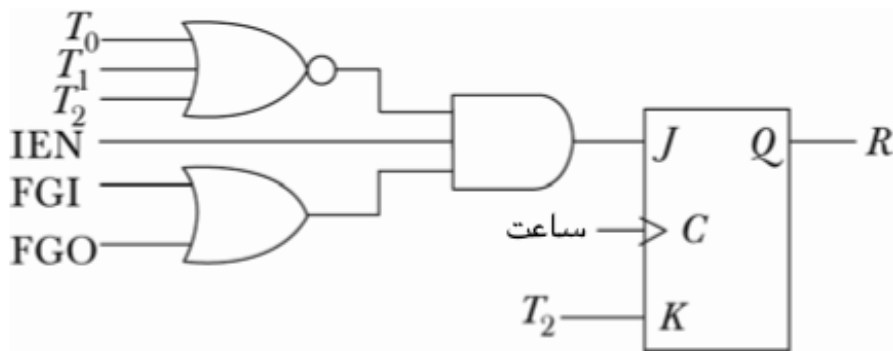
۲۳- مدار منطقی کامل وقفه را در کامپیوتر پایه نشان دهید. از فلیپ فلاپ JK استفاده کنید و گیت‌ها را به حداقل برسانید.

حل:

از جدول ۶ - ۵ داریم:

$$(T_0 + T_1 + T_2)'(IEN)(FGI + FGO) : R \leftarrow 1$$

$$RT_2 ; R \leftarrow 0$$



۲۴- عبارت بولی را برای x_2 (جدول ۷ - ۵ را ببینید) بدست آورید. نشان

دهید که x_2 می‌تواند با یک گیت AND و یک گیت OR تولید شود.

حل:

x_2 خروجی PC را روی گذرگاه قرار می‌دهد.

از جدول ۶ - ۵ داریم:

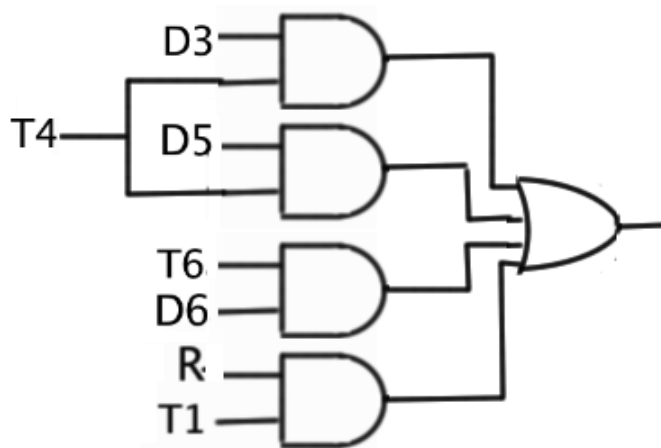
$$R'T_0 : AR \leftarrow PC$$

$$RT_0 : TR \leftarrow PC$$

$$D_5T_4 : M[AR] \leftarrow PC$$

$$x_2 = R'T_0 + RT_0 + D_5T_4 = (R' + R)T_0 + D_5T_4$$

که $R + R' = 1$.

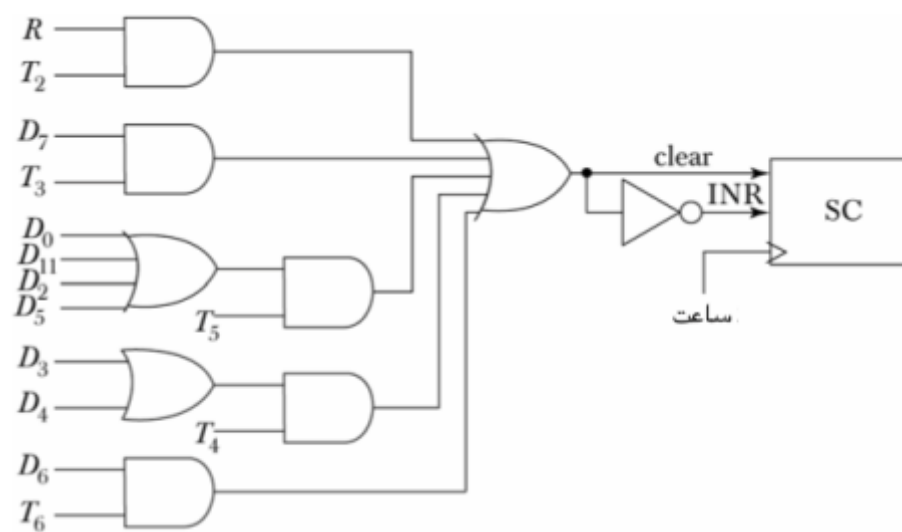


۲۵- عبارت بولی برای یک مدار گیتی که شمارنده SC را پاک کند بدست آورید. دیاگرام منطقی آن را رسم نموده و نشان دهید که چگونه خروجی به ورودی‌های INC و CLR از شمارنده‌ی SC وصل می‌شود. (شکل ۶-۵). تعداد گیت‌ها را حداقل نمایید.

حل:

از جدول ۶-۵ داریم:

$$CLR(SC) = RT_2 + D_7T_3(I' + I) + (D_0 + D_1 + D_2 + D_5)T_5 + (D_3 + D_4)T_4 + D_6T_6$$



فصل ششم: برنامه‌نویسی کامپیوتر پایه

۱- برنامه زیر در واحد حافظه کامپیوتر پایه ذخیره شده است. محتوای AC ، PC ، IR را به شانزده شانزدهی در پایان اجرای هر دستورالعمل نشان دهید. تمام اعداد در جدول به شانزده شانزدهی هستند.

مکان	دستورالعمل
۰۱۰	CLA
۰۱۱	ADD 016
۰۱۲	BUN 014
۰۱۳	HLT
۰۱۴	AND 017
۰۱۵	BUN 013
۰۱۶	CIA5
۰۱۷	93C6

حل:

ابتدا ترتیب اجرای دستورالعمل‌ها را مطابق روبرو خواهیم داشت:

- 1- CLA
- 2- ADD 016
- 3- AND 017
- 4- HLT

حال رجیسترهای AC و PC و IR در این ترتیب اجرا به صورت زیر خواهد بود.

AC	PC	IR	اجرای دستورالعمل
0000	011	7800	CLA
C1A5	012	1016	ADD 016
C1A5	014	4014	BUN 014
8184	014	7001	HLT
8184	015	0017	AND 017

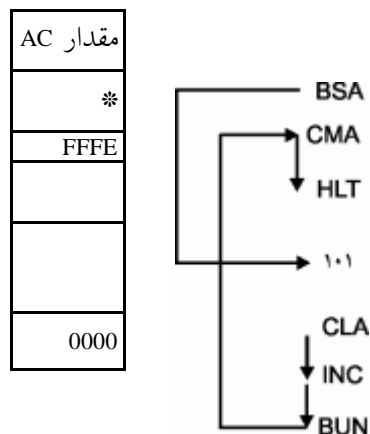
کد شانزده شانزدهی دستورالعمل‌های کامپیوتر در جدول ۶-۱ کتاب موجود است.

۲- برنامه زیر لیستی از دستورالعمل‌ها در مبنای شانزده است. کامپیوتر با شروع از آدرس ۱۰۰ دستورالعمل‌ها را اجرا می‌کند. محتوای AC و کلمه حافظه درون آدرس ۱۰۳ هنگامی که کامپیوتر متوقف می‌شود چیست؟

حل: با استفاده از جدول «۶-۱» دستورالعمل‌ها را بدست می‌آوریم.

مکان	دستورالعمل "Hex"	دستورالعمل
100	5103	BSA 103
101	7200	CMA
102	7001	HLT
103	0000	101
104	7800	CLA
105	7020	INC
106	C103	BUN 103 I

در ارتباط با اجرای دستورالعمل‌ها، دستورالعمل ۱۰۳ BSA، مکان ۱۰۱ در آدرس ۱۰۳ به عنوان آدرس برگشت ذخیره شده و PC از آدرس ۱۰۴، اجرای برنامه را ادامه می‌دهد. پس ترتیب اجرای دستورالعمل‌ها به صورت زیر خواهد بود:



0001

همچنین مقدار AC در اجرای دستورالعمل‌ها نشان داده شده است.

۳- برنامه زبان اسمبلی تولید شده با یک کامپایلر را از برنامه فرترن زیر

بنویسید؛ متغیرها را عدد صحیح فرض کنید.

- 1) SUM=0
- 2) SUM=SUM+ A + B
- 3) DIF=DIF-C
- 4) SUM=SUM+DIF

حل: تک تک خطوط را به برنامه اسمبلی مورد نظر تبدیل می‌کنیم.

- 1) SUM = 0

CLA
STA SUM

- 2) SUM=SUM+ A + B

LDA SUM
ADD A
ADD B
STA SUM

- 3) DIF=DIF-C

LDA C
CMA
INC
ADD DIF
STA DIF

- 4) SUM=SUM+DIF

LDA SUM
ADD DIF
STA SUM

۴- آیا حرف I می‌تواند به عنوان یک آدرس سمبلیک در برنامه زبان اسمبلی

تعریف شده برای کامپیوتر پایه استفاده شود؟ پاسخ خود توضیح دهید.

حل: بله؛ استفاده از I به عنوان آدرس سمبلیک دو صورت می‌تواند داشته باشد.

I [دستورالعمل] ۱)

I I [دستورالعمل] ۲)

برای نمونه در دستورالعمل بارگذاری LDA داریم:

۱) LDA I

۲) LDA I I

در عبارت ۱، I می‌تواند یک آدرس سمبولیک فرض شود. در عبارت ۲، I اول

آدرس سمبلیک، و I دوم بیت غیرمستقیم (Indirect Reference) را نشان می‌دهد.

۵- اگر سطری از کد برنامه دارای شبه دستورالعمل ORG یا END باشد، عنوان

نیز داشته باشد، در مرور اول اسمبلر (شکل ۶-۱) چه رخ خواهد داد. فلوچارت را

تغییر دهید تا در صورت رخداد بالا پیغام خطا در بر داشته باشد.

حل: عباراتی مانند ORG یا END در کد اسمبلی کامپیوتر پایه، نباید در مرورها به

عنوان "Label" فرض شوند. برای اینکه، تمایز بین عبارات مثل END و ORG، و

عنوان‌ها "Label" برقرار باشد تا ORG و END تشخیص داده شوند، تغییرات زیر را

برای کشف خطا در مرور اول انجام می‌دهیم.

DEC -35

ب) نشان دهید که یک کلمه حافظه، می تواند ترجمه شده دودویی کد را ذخیره کند. همچنین محتوای دودویی آن را ارایه دهید.

الف) هر کلمه ۱۶ بیت و دو بایت ظرفیت دارد. از طرفی حروف سطر، در حافظه دستورالعمل به صورت که ASCII ۸ بیتی نگهداری می‌شود. پس:

Memory Word High Low		کد مبنای ۱۶		کد دودویی
D	E	۴۴	۴۵	۰۱۰۰ ۰۱۰۰ ۰۱۰۰ ۰۱۰۱
C	“space”	۴۳	۲۰	۰۱۰۰ ۰۰۱۱ ۰۰۱۰ ۰۰۰۰

—	۳	۲D	۳۳	۰۰۱۰	۱۱۰۱	۰۰۱۱	۰۰۱۱
۵	CR (به معنی پایان جمله در خط جاری)	۳۵	۰D	۰۰۱۱	۰۱۰۱	۰۰۰۰	۱۱۰۱

(ب) بدیهی است، مثلاً کد دودویی عدد ۳۵ در مبنای ۲، با کد نوشته شده در بالا متفاوت است.

$$(35)_{10} = (0000\ 0000\ 0010\ 0011)_2$$

$$\Rightarrow (-35) = (FFDD)_{16}$$

۷- الف) جدول سمبل آدرس تولید شده برای برنامه جدول ۱۳-۶ در حین مرور اول اسمبلر را بدست آورید.

(ب) برنامه ترجمه شده را به شانزده شانزدهی بنویسید.

حل: الف) برنامه جدول (۱۳-۶)، برنامه سمبلیک برای جمع ۱۰۰ عدد است. طبق فلوچارت ارائه شده برای مرور اول اسمبلر داریم:

LOP	۱۰۵	$(-100)_{10} = (FF9C)_{16}$
ADS	۱۰B	$(75)_{10} = (0048)_{16}$
PTR	۱۰C	$(23)_{10} = (0017)_{16}$
NBR	۱۰D	
CTR	۱۰E	
SUM	۱۰F	

در عبارات بالا، مکان ۱۰۵، یعنی ۱۵ امین دستورالعمل بعد از دستورالعمل ۱۰۰ ORG، بقیه نیز به همین ترتیب.

(ب)

Loc	Hex			Loc	Hex		
100	210B	LDA	ADS	10B	0150	ADS,	HEX 150
101	310C	STA	PTR	10C	0000	PTR,	HEX 0
102	210D	LDA	NBR	10D	FF9C	NBR,	DEC-100
103	310E	STA	CTR	10E	0000	CTR,	HEX 0
104	7800	CLA		10F	0000	SJH,	HEX 0
105	910C	LOP, ADD	PTR I				ORG 150
106	610C	ISZ	PTR	150	004B		DEC 150
107	610E	ISZ	CTR	:	:		:
108	4105	BUN	LOP	:	:		:
109	310F	STA	SUM	1B3	0017		DEC 23
10A	7001	HLT					END

۸ - شبه دستورالعمل BSSN (بلاک آغاز شده به وسیله سمبل) گاهی برای رزرو

N کلمه حافظه برای گروهی از عملوندها به کار می‌رود. مثلاً سطر

A, BSS10

به اسمبلر اطلاع می‌دهد که بلاکی از ۱۰ (دهدهی) مکان را با شروع از مکان

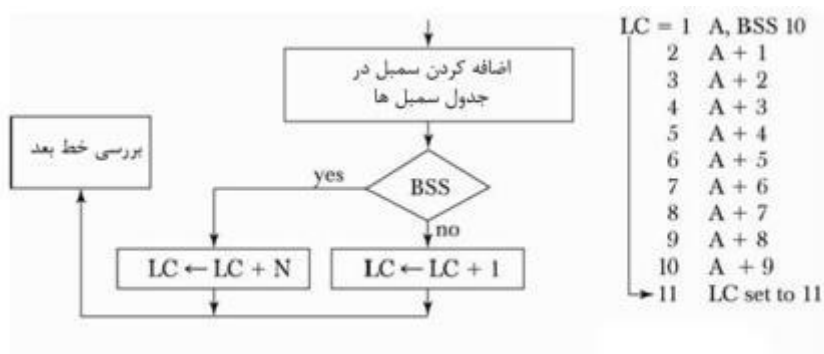
متناظر با سمبل A خالی بگذارد. این مشابه جمله (10) Dimension A در فرترن است.

فلوچارت شکل ۱-۶ را برای پردازش این شبه دستور تغییر دهید.

حل: برای اینکه رزرو ۱۰ کلمه حافظه در الگوریتم مرور اول اسمبلر کامپیوتر پایه

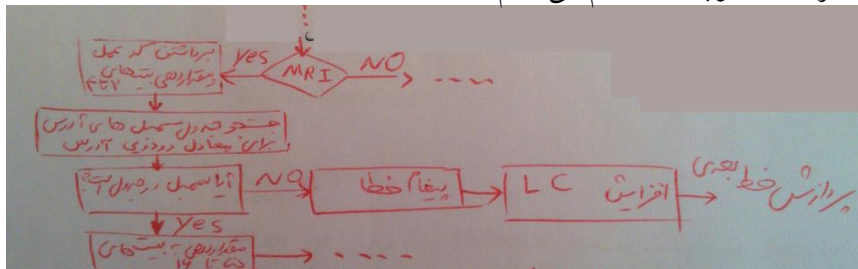
صورت پذیرد، می‌بایست فاصله مقادیر "Location Counter" قبل و بعد، ۱۰ عدد باشد.

بنابراین تغییرات زیر را در فلوچارت مرور اول انجام می‌دهیم:



۹- فلوچارت شکل ۶-۲ را تغییر دهید که اگر یک دستور سمبولیک با برچسب تعریف نشده باشد، یک پیام خطا در بر گیرد.

حل: فلوچارت ۶-۲، مربوط به مرور دوم اسمبلر کامپیوتر پایه است. طبق سؤال تغییر زیر را در فلوچارت انجام می دهیم:



۱۰- نشان دهید که چگونه جدول های MRI و non - MRI می توانند در حافظه ذخیره شوند.

حل: شیوه تنظیم دستورات دسترسی به حافظه و سایر دستورات در جداول MRI و non MRI مانند زیر خواهد بود:

الف) ذخیره جدول دستورات دسترسی به حافظه: (MRI)

برای مثال برای دستور ADD که یک دستور دسترسی به حافظه است داریم:

ADD (مقدار)

Memory word	symbol		Hex	
۱	A	D	۴۱	۴۴
۲	D	space	۴۴	۲۰
	مقدار (Value)		مقدار به شانزده شانزدهی	

ب) ذخیره جدول دستورات (non MRI)

برای مثال برای دستور CMA داریم:

CMA

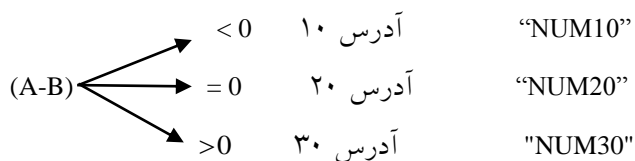
Memory word	symbol		Hex	
۱	C	M	۴۳	۵۳
۲	A	space	۴۱	۲۰
	مقدار (Value)		مقدار به شانزده شانه‌دهی	

۱۱- برنامه اسمبلی (معادل دودویی دستورالعمل‌ها) تولید شده به وسیله کامپایلر را برای عبارت شرطی زیر بدست آورید:

IF(A-B)10,20,30

حل:

توضیح: برنامه شروط زیر را چک کرده و انشعاب پیدا می‌کند:



ابتدا عمل تفریق را انجام داده سپس با استفاده از دستورالعمل‌های مقایسه و گذر مانند SNA و SPA و SZA، انشعاب‌های لازم را انجام می‌دهیم: دو نمونه از برنامه‌هایی که به عنوان راه حل به ذهن می‌رسند: برنامه ۱ سمت چپ، و ۲ سمت راست.

LDA B	
CMA	
INC	
ADD A /	انجام عمل تفریق
SPA /	گزار در صورت مثبت بودن AC
BUN NUM10 /	(A-B) < 0
SZA	
BUN NUM30 /	(A-B) > 0
BUN NUM20 /	(A-B) = 0

LDA B	
CMA	
INC	
ADD A	
SNA	
BUN NUM30	
SZA	
BUN NUM10	
BUN NUM20	

توجه کنید، برنامه‌های فوق اشکال اساسی در برقراری شرط‌های مورد نظر دارند. در هر دو برنامه فرض کنید $(A-B)=0$ ، چون $(A-B)$ عدد مثبت نیست، (در برنامه (۱)) برنامه به NUM10 پرش خواهد کرد. همچنین در برنامه (۲) چون $(A-B)$ عدد منفی نیست، برنامه به لیبل NUM30 پرش خواهد کرد. در هر دو حالت $(A-B)=0$ شناسایی نخواهد شد. راه حل که این مشکل را نداشته باشد مانند زیر است:

```
LDA B
CMA
INC
ADD A
SPA
BUN ZeroOR Equal /  $(A-B) \leq 0$ 
BUN NUM 30 /  $(A-B) > 0$ 
ZeroOREual, SZA
BUN NUM 10 /  $(A-B) < 0$ 
BUN NUM 20 /  $(A-B) = 0$ 
```

۱۲- الف) توضیح دهید که اجرای برنامه زیر چه کاری انجام می‌دهد. مقدار

مکان CTR وقتی که کامپیوتر متوقف می‌شود چیست؟

ب) جدول سمبل‌های آدرس بدست آمده در مرور اول اسمبلر را بنویسید.

ج) کد شانزده شانزدهی (Hex) برنامه ترجمه شده را بنویسید.

کد برنامه

```
ORG 100
CLE
CLA
STA CTR
LDA WRD
SZA
BUN ROT
BUN STP
ROT, CIL
SZE
BUN AGN
BUN ROT
AGN, CLE
ISZ CTR
SZA
```



```
BUN ROT
STP , HLT
CTR , Hex 0
WRD , Hex 62C1
END
```

حل:

الف) این برنامه، تعداد رقم‌های ۱ موجود در عدد ذخیره شده در مکان WRD (در اینجا 62C1 شانزده شانزده‌ی) را می‌شمارد. پس داریم:

$$62C1 = (0110\ 0010\ 1100\ 0001)_2$$

تعداد ۱ های موجود در WRD = ۶

بنابراین CTR خواهد بود: $(0006)_{16}$

ب) و ج)

		ORG	100	
100	7400	CLE		
101	7800	CLA		
102	3110	STA	CTR	/Initialize counter to zero
103	2111	LDA	WRD	
104	7004	SZA		
105	4107	BUN	ROT	
106	410F	BUN	STP	/ Word is zero; stop with CTR =0
107	7040	ROT, CIL		/Bring bit to E
108	7002	SZE		
109	410B	BUN	AGN	/bit = 1, go to count it
10A	4107	BUN	ROT	/bit = 0, repeat
10B	7400	AGN, CLE		
10C	6110	ISZ	CTR	/Increment counter
10D	7004	SZA		/check if remaining bits = 0
10E	4107	BUN	ROT	/No; rotate again
10F	7001	STP, HLT		/yes, stop
110	0000	CTR, HEX	0	
111	62C1	WRD, HEX	62C1	
		END		

۱۳- یک حلقه برنامه با استفاده از اشاره‌گر و شمارنده بنویسید که محتوای مکان

شانزده شانزده‌ی ۵۰۰ تا 5FF را ۰ کند.

حل: در ارتباط با مقدار شمارنده، می‌بایست فاصله دهدهی ۵۰۰ تا 5FF را در مبنای ۱۶ محاسبه کنیم.

$$\begin{aligned} 0 - F \quad 16 \\ \Rightarrow 16 \times 16 = (256)_{10} \\ 0 - F \quad 16 \end{aligned}$$

بنابراین $CTR = -256$ انتخاب می‌شود به عنوان شمارنده بالارونده.
در ضمن، در ارتباط با افزایش اشاره گر برای دسترسی به مکان های بعدی دو راه داریم:

(۱) محتوای اشاره گر در رجیستر AC ریخته شده و از دستورالعمل INC استفاده کنیم.

(۲) از دستورالعمل ISZ استفاده کنیم. زیرا اطمینان داریم مقدار اشاره گر، صفر نخواهد شد.

پس داریم:

```
ORG 100
LDA ADS
STA PTR
LDA NBR
STA CTR
CLA
LOP, STA PTR I
ISZ PTR
ISZ CTR
BUN LOP
HLT
ADS, HEX 500
PTR, HEX 0
NBR, DEC -256
CTR, HEX 0
END
```

۱۴- برنامه‌ای بنویسید که دو عدد را با روش جمع تکراری، درهم ضرب کند.

مثلاً برای ضرب 5×4 ، برنامه ۵ را چهار بار جمع کند، یعنی $5 + 5 + 5 + 5$

حل: در این روش ضرب، طبق صورت سؤال، عدد دوم در شمارنده قراردادده می‌شود و عدد اول با خودش جمع می‌شود تا بعد از رسیدن شمارنده بالارونده به عدد ۰، حاصلضرب آماده باشد.

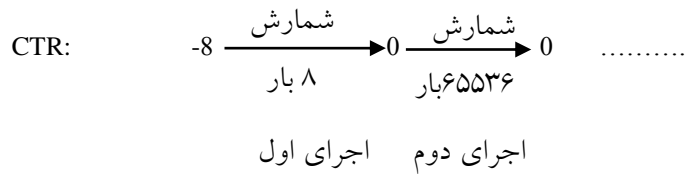
دلیل استفاده از شمارنده بالارونده در برنامه‌های اسمبلی، این است که دستورالعمل قابل استفاده برای چک کردن پایان شمارنده، ISZ است. پس داریم: (برای مثال $\frac{5}{B} \times \frac{4}{A}$)

```
LDA    A
SZL
BUN    NZR
HLT
NZR , CMA
INC
STA    CTR
CLA
LOP , ADD B
ISZ    CTR
BUN    LOP
HLT
A, DEC 4
B, DEC 5
CTR , HEX 0
END
```

۱۵- برنامه ضرب جدول ۱۴-۶ مقداردهی اولیه نشده است. پس از یکبار اجرای برنامه، مکان CTR مقدار صفر می‌گیرد. نشان دهید اگر برنامه دوباره از مکان ۱۰۰ شروع شود، حلقه 65536 بار تکرار خواهد شد. دستورهای لازم برای مقداردهی اولیه برنامه را اضافه کنید.

حل: با توجه به برنامه جدول ۱۴-۶ که برنامه ضرب دو عدد مثبت است، اولین باری که برنامه اجرا می‌شود، شمارنده از ۸- تا ۰، ۸ بار می‌شمارد تا ۸ بار حلقه اجرا شده و ۸ بار شیفت اجرا شود. اگر برنامه دقیقاً دوباره از آدرس ۱۰۰ تکرار اجرا شود، یعنی مقادیر متغیرها حفظ شود، با توجه به اینکه از دستور ISZ استفاده می‌شود و همچنین با توجه به اینکه CTR، یک متغیر ۱۶ بیتی است، برای اینکه دوباره ۰ شود

باید 65536 بار بشمارد. یعنی:



همچنین متغیر P نیز حاصلضرب در اجرای قبلی الگوریتم باقی خواهد ماند. برای رفع این اشکال، CTR و P را مقداردهی اولیه می‌کنیم:

LDA NBR } مقداردهی اولیه CTR
STA CTR }
CLA }
STA P } مقداردهی اولیه P یا 0

.....

.....

NBR , DEC -8
CTR , Hex 0
P , Hex 0

۱۶- برنامه‌ای برای ضرب دو عدد مثبت بدون علامت ۱۶ بیتی بنویسید و حاصلضرب آنها را به صورت یک عدد بدون علامت با دقت مضاعف بدست آورید.

حل: در کتاب، جمع دو عدد با دقت مضاعف “double precision” توضیح داده شده است، در رابطه با برنامه ضرب دو عدد مثبت بدون علامت ۱۶ بیتی، چون نتیجه عمل ضرب حداکثر ۳۲ بیتی خواهد بود، بنابراین برای MultipliCand به دلیل نیاز به شیفت به چپ و همینطور برای حاصلضرب جزئی “Partial Product” متغیرهای ۳۲ بیتی اخذ خواهد شد. یعنی برای ۵ هر کدام از این دو مقدار، ما به دو کلمه از حافظه نیاز خواهیم

داشت.

بنابراین در جمع حاصلضرب جزیی با MultipliCand از جمع با دقت مضاعف و برای شیفت به چپ MutipliCand، از شیفت به چپ در دو مکان از حافظه استفاده خواهیم کرد.

همچنین، شمارنده ما به جای ۸-، ۱۶- مقداردهی خواهد شد.

LOP , CLE	
LDA	
CIR	
STA	
SZE	
BUN One	
BUN Zero	
One , LDA XL	<p>جمع با دقت مضاعف X و P</p> <p>مانند جدول ۱۵-۶ در کتاب</p>
ADD PL	
STA PL	
CLA	
CIL	
ADD XH	
ADD PH	
STA PH	
CLE	
Zero , LDA XL	
CIL	<p>شیفت به چپ با دقت مضاعف X</p> <p>«بیت E، با ارزش‌ترین بیت XL را به</p> <p>کم ارزش‌ترین بیت XH انتقال می‌دهد.»</p>
STA XL	
LDA XH	
CIL	
STA XH	
ISZ CTR	
BUN LOP	
HLT	

۱۷- برنامه‌ای برای ضرب دو عدد علامت‌دار بنویسید. اعداد منفی ابتدا به صورت متمم ۲ علامت‌دار نشان داده می‌شوند. حاصلضرب باید با دقت ساده باشد و در صورت منفی بودن هم، به صورت متمم ۲ علامت‌دار نشان داده شود.

حل:

X Multipli Cand

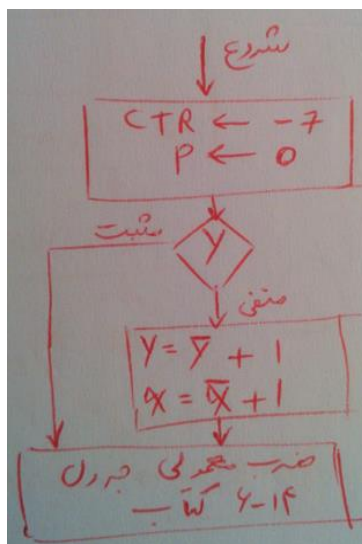
×

Y Multiplier
Result

در عبارت بالا، بحث بسیار مهم، اصالت بیت‌های Multiplier است زیرا ۰ و ۱ بودن این بیت‌ها، جمع موجود در الگوریتم ضرب را درگیر کرده و نتیجه را تغییر خواهد داد. بنابراین سعی ما در مثبت بودن Y یعنی Multiplier است. بنابراین:

اگر Y مثبت باشد، همان ضرب معمولی انجام خواهد شد.
اگر Y منفی باشد، هر دو Y و X را مکمل ۲ کرده تا Y مثبت شود. سپس ضرب معمولی انجام می‌شود.

پس:



نکته در ارتباط با مقدار اولیه شمارنده، $CTR = -7$ لحاظ شده است. دلیل این است که چون نهایتاً Y مثبت است، بیت Y_7 یعنی با ارزش‌ترین بیت (بیت علامت) ۰ است. پس با ۷ بار شیفت ضرب حاصل خواهد شد.

۱۸- برنامه‌ای بنویسید که دو عدد با دقت مضاعف را از هم تفریق کند.

حل: طبق تعریف تفریق داریم:

$$A - B = A + (-B)$$

بنابراین بعد از مکمل ۲ کردن BL و BH در عین حفظ رقم نقلی، جمع با دقت مضاعف را انجام می‌دهیم. دلیل ذخیره رقم نقلی در رجیستر ذخیره موقت داده "Temp" این است که ما به AC برای ادامه مکمل ۲ کردن B نیاز داریم.

پس: ذخیره شود در

$C \longleftarrow A - B$

```
CLE
LDA  BL
CMA
INC
ADD  AL
STA  AL
CLA
CIA
CIL
STA  TMP
LDA  BH
CMA
ADD  AH
ADD  TMP
STA  CH
HLT
TMP , Hex 0
```

۱۹- برنامه‌ای بنویسید که OR انحصاری منطقی دو عملوند منطقی را حساب کند.

حل: می‌دانیم:

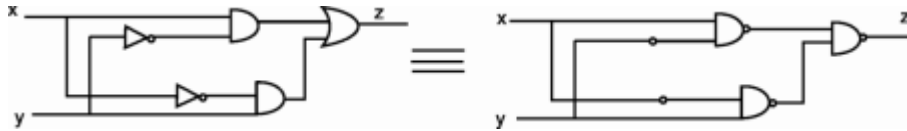
$$z = x \oplus y = xy' + x'y$$

حال دو روش برای بیان عبارت براساس AND و NOT داریم: (به این دلیل که دستورالعمل‌های AND و CMA برای ساخت ترکیبات منطقی استفاده می‌شوند).

۱- از راه جبربول:

$$z = xy' + x'y = [(xy)'. (x'y)']'$$

۲- رسم تابع منطقی با گیت‌های AND و NOT



برنامه بدین شرح خواهد بود:

```

LDA  y
CMA
AND  x
CMA
STA TMP/(xy)'  

LDA  x
CMA
AND  y
CMA
AND  TMP
CMA
STA  z
HLT
x, .....
y, .....
z, .....
TMP, .....

```

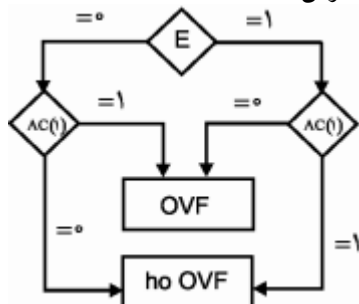
۲۰- برنامه‌ای برای شیفت حسابی به چپ بنویسید که در صورت رخ دادن

سرریز به OVF انشعاب کند.

حل: با توجه به اینکه تنها شیفت چرخشی به چپ در دستورالعمل‌ها داریم «CIL» برای تشخیص سرریز «مثبت شدن عدد منفی یا منفی شدن عدد مثبت» می‌بایست، بیت موجود در رقم نقلی را با بیت علامت رجیستر، بعد از اجرای دستور CIL «در حالی که E را پاک کرده‌ایم»، مقایسه کنیم.

از روندنمای زیر می‌توان متوجه شد که OR انحصاری AC(1) و E، وجود رخداد

OVF را تعیین می‌نمایند. (منظور از AC(1)، با ارزش ترین بیت AC است)



با استفاده از مقایسه و پرش‌ها برنامه را به شکل زیر طراحی می‌کنیم:

```
LDA    X
CLE                    /      مقدمات شیفت به چپ حسابی
CIL
SZE
BUN    ONE
SPA
BUN    OVF              /      شامل عدد منفی AC, E=0
BUN    EXIT
ONE , SNA
BUN    OVF              /      شامل عدد مثبت AC, E=1
EXIT , HLT
```

۲۱- زیر روالی برای تفریق دو عدد بنویسید. در برنامه فراخواننده، مفروق و

مفروق منه به دنبال دستورالعمل BSA آمده‌اند. حاصل تفریق به سومین مکان پس از BSA در برنامه اصلی باز می‌گردد.

حل: در حقیقت سؤال به دنبال یک زیرروال همراه با پارامتردهی به صورت گفته شده هست.

برای برنامه صدا زننده زیر روال داریم:

```
BSA    SUB      }
Hex    12 34    } 4321 - 1234
Hex    4321     }
Hex    0        / نتیجه
```

زیر روال:

```
SUB    , Hex
LDA    SUB I      / دریافت اولین پارامتر
```

CMA

INC

ISZ SUB / افزایش اشاره گر برای دسترسی به پارامتر بعدی

ADD SUB I / پارامتر دوم جمع یا مکمل ۲ پارامتر اول

ISZ SUB

STA SUB

ISZ SUB

BUN SUB I

۲۲- زیرروالی بنویسید که هر داده را در یک بلاک متمم کند. در برنامه

فراخواننده، دو پارامتر پس از دستورالعمل BSA آمده است: آدرس شروع بلاک و تعداد کلمات بلاک.

حل: با فرض آنکه آدرس شروع $(100)_{16}$ و تعداد کلمات حافظه ۳۲ است، داریم:

برنامه صد زننده زیر روال:

BSA CMP

Hex 100

DEC 32

CMP , Hex 0

LDA CMP I

STA PTR

ISZ CMP

LDA CMP I

CMA

INC

CMA

INC

STA CTR / شمارنده

LOP, LDA PTR I

CMA

STA PTR I

ISZ PTR

ISZ CTR

BUN LOP

ISZ CMP

BUN CMP I

PTR,

CTR,

در دستورالعمل‌های بالا، CMP ISZ و PTR ISZ صرفاً برای افزایش یک واحد استفاده شده‌اند.

۲۳- زیر روال بنویسید که E و AC را چهار بار به راست بچرخانند. اگر ابتدا محتوای AC برابر $(079C)_{16}$ و $E=1$ باشد، محتوای آنها پس از اجرای زیر روال چیست؟

حل: طبق صورت سؤال داریم:

CR4 , Hex 0
CIR
CIR
CIR
CIR
BUN CR4 I

در رابطه با مقادیر داریم:

$$AC:(079C)_{16} = (0000 \ 0111 \ 1001 \ 1100)_2, E=1$$

$$AC = (1000 \ 0011 \ 1100 \ 1110)_2, E=0 \quad \text{بعد از چرخش اول}$$

$$AC = (0100 \ 0001 \ 1110 \ 0111)_2, E=0 \quad \text{بعد از چرخش دوم}$$

$$AC = (0010 \ 0000 \ 1111 \ 0011)_2, E=1 \quad \text{بعد از چرخش سوم}$$

$$AC = (1001 \ 0000 \ 0111 \ 0111 \ 1001)_2, E=1 \quad \text{بعد از چرخش چهارم}$$

بنابراین، بعد از اجرای زیر روال:

$$AC = (9079)_{16}, E=1$$

۲۴- برنامه‌ای بنویسید که کاراکترهای ورودی را بپذیرد، هر دو کاراکتر را در

یک کلمه فشرده کند و پشت سر هم در بافر حافظه ذخیره نماید. اولین آدرس بافر

$(400)_{16}$ و اندازه آن $(512)_{10}$ است. اگر بافر سرریز کند کامپیوتر بایستی متوقف

می شود.

حل: فرض شود که زیر روال به اسم IN2 «جدول ۲۰-۶ کتاب» دریافت و فشرده

کردن دو کاراکتر در یک کلمه را بر عهده دارد.

برنامه به صورت زیر است:

```

LDA  ADS
STA  PTR      /      تنظیم اشاره گر
LDA  NBR
STA  CTR      /      تنظیم شمارنده
LOP , BSA  IN2  /      بعد از اجرای این زیر روال، دو کاراکتر در AC هستند
STA  PTR  I
ISZ  PTR      /      صرفاً افزایش اشاره گر
ISZ  CTR      /      افزایش و مقایسه شمارنده
BUN  LOP
HLT
ADS , Hex  400
PTR , Hex  0
NBR , DEC  - 512
CTR , Hex  0

```

۲۵- برنامه ای بنویسید که دو کاراکتر در مکان WRD را باز کند. و آنها را در بیت های ۰ تا ۷ از مکان های CH1 و CH2 ذخیره کند. بیت ۹ تا ۱۵ باید صفر باشد.

حل: ابتدا زیر روال به نام SR8 را تعریف می کنیم که ۸ بار، محتویات رجیستر AC را به راست شیفت می دهد. می توانیم قبل از هر بار شیفت، CLE انجام دهیم و یا بعد از ۸ بار شیفت چرخشی، محتوا را با 00FF، AND کنیم، تا نیمه بالایی رجیستر ۰ باشد.

```

SR8 , Hex  0
CIR
CIR
CIR
CIR
CIR
CIR
CIR
CIR
BUN SR8  I

```

برای برنامه اصلی داریم:

```

LDA  WRD
AND  MS1  /  AND  With  00FF
STA  CH1

```

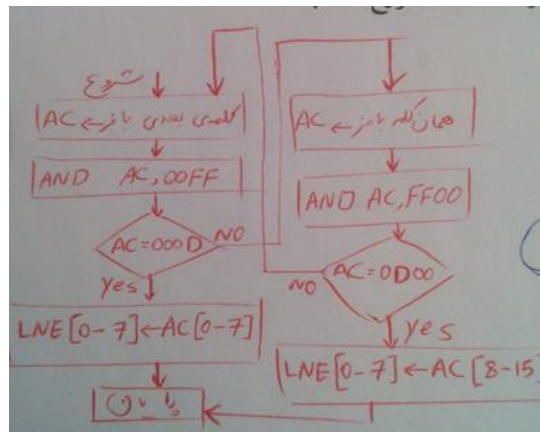
```
LDA WRD
AND MS2 / AND With FF00
CLE
BSA SR8
```

```
STA CH2
HLT
WRD, Hex, ...
CH1, Hex, ...
CH2, Hex, ...
MS1, Hex 00FF
MS2, Hex FF00
```

۲۶- فلوچارتی را برای برنامه‌ای بدست آورید که وجود CR (0D در مبنای ۱۶)

را در بافر حافظه بررسی کند. بافر دارای ۲ کاراکتر در هر کلمه است. وقتی به کد CR دیده شود، برنامه آن را به بیت‌های ۰ تا ۷ مکان LNE منتقل می‌کند بدون آنکه بیت‌های ۸ تا ۱۵ تغییر کنند.

حل: کد CR در بافر، یا در کد بالا و یا در کد پایین هر کلمه از بافر قرار دارد. برای کد بالا، از AND با FF00 و برای کد پایین از AND با 00FF استفاده می‌کنیم.



۲۷- روال سرویس‌دهی SRV را از جدول ۲۳-۶ به کد شانزده شانه‌ای

معادلش ترجمه کنید. فرض کنید که روال از مکان ۲۰۰ شروع شده باشد.

حل:

Location	Hex code	
200	3213	SRV, STA SAC
201	7080	CIR
202	3214	STA SE
203	F200	SKI
204	4209	BUN NXT
205	F800	INP
206	F400	OUT
207	B215	STA PT1 I
		ISZ PT1
208	6215	NXT, SKO
209	F100	
20A	420E	BUN EXT
20B	A216	LDA PT2 I
20C	F400	OUT
20D	6216	ISZ PT2
20E	2214	EXT, LDA SE
20F	7040	CIL
210	2213	LDA SAC
211	F080	ION
212	C000	BUN ZR0 I
213	0000	SAC, ---
214	0000	SE, ---
215	0000	PT1, ---
216	0000	PT2, ---

۲۸- یک روال سرویس‌دهی وقفه که تمام عملیات لازم را انجام می‌دهد بنویسید ولی وسیله ورودی فقط اگر مکان MOD تماماً ۱ باشد، سرویس‌دهی شود. وسیله خروجی هم فقط اگر مکان MOD تماماً ۰ باشد سرویس‌دهی شود.

حل: همانند برنامه سرویس‌دهی جدول ۲۳-۶، با این تفاوت که، هنگام سرویس‌دهی به وسیله ورودی، با دستورالعمل ISZ، $(FFFF)_{16}$ بودن MOD بررسی می‌شود. و هنگام سرویس‌دهی به وسیله خروجی، با دستورالعمل SZA، $(\phi\phi\phi\phi)_{16}$ بودن MOD بررسی می‌شود.

SRV, STA SAC
CIR

STA SE
LDA MOD
ISZ
BUN NXT / MOD \neq FFFF
SKI
BUN NXT
INP
OUT
STA PT1 I
ISZ PT1
NXT , LDA MOD
SZA
BUN EXT / MOD \neq $\phi\phi\phi\phi$
SKO
BUN EXT
LDA PT2 I
OUT
ISZ PT2
EXT ,

ادامه برنامه مانند جدول ۲۳-۶

فصل هفتم: کنترل ریز برنامه‌نویسی شده

۱- تفاوت بین ریز پردازنده و ریز برنامه چیست؟ آیا می‌توان ریز پردازنده‌ای بدون یک ریز برنامه طراحی کرد؟ آیا همه‌ی کامپیوترهای ریز برنامه‌ریزی شده ریز پردازنده هستند؟

حل: یک ریز پردازنده یک پردازنده کوچک است (کامپوتری بر روی یک تراشه). ریز برنامه، برنامه‌ای است شامل دنباله‌ای از ریز عمل‌ها، واحد کنترل یک ریز پردازنده می‌تواند به صورت سخت‌افزاری (سیم‌بندی) یا ریز برنامه‌ریزی شده باشد، براساس طراحی مورد نظر، لزومی ندارد یک کامپیوتر ریز برنامه‌ریزی شده ریز پردازنده باشد.

۲- تفاوت بین کنترل سخت‌افزاری و کنترل ریز برنامه‌نویسی شده را بیان کنید. آیا می‌توان یک کنترل سخت‌افزاری مرتبط شده با یک حافظه کنترلی داشت؟

حل: واحد کنترل سخت‌افزاری کاملاً به صورت سخت‌افزاری پیاده‌سازی شده که دارای سرعت بالایی است اما در برابر تغییر مجموعه دستورها، انعطاف‌پذیر نیست و باید دوباره طراحی و پیاده‌سازی شود. واحد کنترل ریز برنامه‌نویسی شده ترکیبی از سخت‌افزار برای اجرای ریز عمل‌ها و حافظه‌ای شامل ریز عمل‌هاست که انعطاف‌پذیری بیشتری در برابر تغییر مجموعه دستورها دارد و نیاز به طراحی و پیاده‌سازی مجدد ندارد (کافی است ریز برنامه آن را تغییر دهیم) اما سرعت کمتری نسبت به واحد کنترل سخت‌افزاری دارد. کنترل سخت‌افزاری، بنا به تعریف شامل حافظه کنترل نیست.

۳- این اصطلاحات را تعریف کنید: الف) ریز عمل، ب) ریز دستورالعمل، ج) ریز برنامه، د) ریز کد.

حل:

ریز عمل: یک عمل پایه کامپیوتر دیجیتال

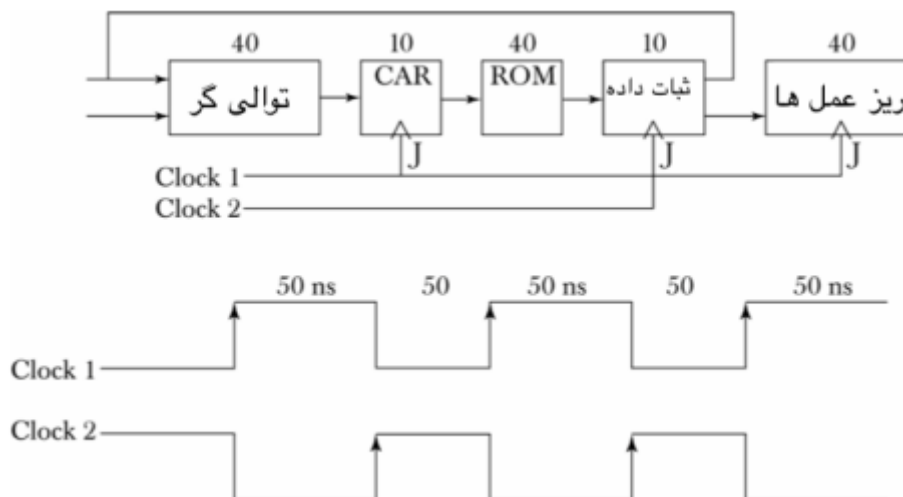
ریز دستورالعمل: یک دستور ذخیره شده در حافظه کنترل

ریز برنامه: دنباله‌ای از ریز دستورالعمل‌ها

ریز کد: همان ریز برنامه است.

۴- سازمان یک کنترل زیر برنامه‌نویسی شده که در شکل ۷-۱ نشان داده شده است دارای تأخیرهای انتشاری زیر است. 40ns برای تولید آدرس بعدی، 10 ns برای انتقال آدرس به داخل ثبات کنترل آدرس و 40ns برای دستیابی به حافظه کنترلی ROM و 10ns برای انتقال ریز دستورالعمل به ثبات کنترل داده و 40ns برای اجرای ریز عمل مورد نظر که به وسیله کلمه کنترلی مشخص شده است. حداکثر فرکانس پالس ساعتی که کنترل می‌تواند به کار برود چیست؟ اگر ثبات کنترل داده بکار نرود، فرکانس ساعت چه خواهد بود؟

حل:



با توجه به شکل داریم:

زمان مورد نیاز برای انجام یک مرحله:

$$\text{حداقل زمان} = 10 + 40 + 10 + 40 = 100 \text{ ns}$$

CAR ROM Data Reg Max (Micro Operations , Sequencer)

پس:

$$\text{فرکانس هر پالس ساعت} = \frac{1}{100 \times 10^{-9}} = \frac{1000}{100} \times 10^6 = 10 \text{ MHz}$$

اگر ثبات داده کنترل استفاده نشود:

$$\text{حداقل زمان} = 10 + 40 + 40 = 90 \text{ ns}$$

CAR ← ROM ← Max (Micro operations , sequencer)

که در این صورت یک فاز داریم و نیاز به 2 Clock نیز نداریم. فرکانس برابر است

با:

$$\text{فرکانس هر پالس ساعت} = \frac{1}{90 \times 10^{-9}} = 1101 \text{ MHz}$$

۵ - سیستم شکل ۷-۲ از یک حافظه کنترلی ۱۰۲۴ کلمه‌ای و هر کلمه ۳۲ بیتی

استفاده می‌کند. ریز دستورالعمل دارای سه میدان است. میدان ریز عمل‌ها ۱۶ بیتی است.

الف) میدان آدرس انشعاب و میدان انتخاب چند بیتی هستند؟

پس به ۱۰ بیت آدرس نیاز داریم.

$$\text{حافظه کنترل} = 2 \times 32$$

۱۶ ۱۰ ۶

ریز عمل	آدرس	انتخاب
---------	------	--------

ب) اگر ۱۶ بیت وضعیت وجود داشته باشد، چند بیت منطق انشعاب برای

انتخاب یک بیت وضعیت لازم است؟

$$\text{۴ بیت : } 16 = 2^4 \text{ حالت}$$

ج) چند بیت برای انتخاب ورودی مالتی پلکسر باقی مانده است؟

$$\text{بیت } 6 - 4 = 2$$

۶ - حافظه کنترل شکل ۷-۲ دارای ۴۰۹۶ کلمه ۲۴ بیتی است.

الف) در ثبات آدرس چند بیت وجود دارد؟ $2^{12} \times 24 = \text{حافظه کنترل}$

پس ثبات آدرس کنترل ۱۲ بیتی است.

ب) هر یک از چهار ورودی که به داخل مالتی پلکسر می‌روند چند بیتی‌اند؟
۱۲ بیتی، چون وارد ثبات آدرس می‌شوند که شامل ۱۲ بیت آدرس است.

ج) تعداد ورودی‌های هر مالتی پلکسر چند است و چند مالتی پلکسر لازم است؟
برای هر بیت یک مالتی پلکسر نیاز داریم که از بین چهار ورودی مختلف انتخاب کند پس ۱۲ مالتی پلکسر ۴ به ۱ نیاز است.

۷- با استفاده از روال نگاشت که در شکل ۳-۷ توصیف شد، برای هر یک از کدهای عملیاتی زیر آدرس اولین ریز دستورالعمل را بنویسید.

الف) ۰۰۱۰ ؛ ب) ۱۰۱۱ ؛ ج) ۱۱۱۱

حل:

الف) $0010 \rightarrow 0001000=8$

ب) $1011 \rightarrow 0101100=44$

ج) $1111 \rightarrow 0111100=60$

۸- یک روال نگاشت تنظیم کنید که هشت ریز دستورالعمل متوالی را برای هر روال در اختیار بگذارد. کد عملیاتی شش بیتی است و حافظه کنترلی ۲۰۴۸ کلمه دارد.

حل:

بیت ۶ = کد عمل

بیت ۱۱ = آدرس حافظه کنترل

رویه نگاشت به این صورت خواهد بود.

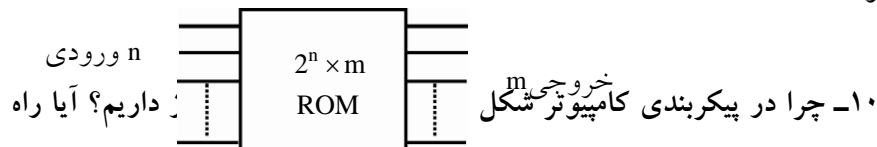
شیفت مورد نیاز $2^3 = 8$

$$\Rightarrow 00 \left| \begin{array}{c} \text{xxxxxx} \\ \text{xxxxxx} \end{array} \right| 000$$

۹- توضیح دهید که چگونه با یک حافظه فقط خواندنی می‌توان عمل نگاشت از کد دستورات عمل به آدرس ریز دستورات عمل را انجام داد. مزیت این روش نسبت به روش شکل ۳-۷ چیست؟

حل: با استفاده از ROM مانند شکل زیر، که در آن n تعداد بیت کد عمل و m تعداد بیت آدرس حافظه کنترل است می‌توان این کار را انجام داد به این صورت که در هر خانه از ROM، آدرس شروع روال مربوط به آن در حافظه کنترل قرار داده شده باشد و بیت‌های کد عمل به ورودی ROM متصل باشد.

مزیت این روش استفاده بهتر از حافظه کنترل و عدم نیاز به پرش برای روال‌های طولانی است (آدرس شروع هر روال لازم نیست مضربی از توان ۲ باشد و عددی دلخواه است).



دیگری برای انتقال اطلاعات از چند منبع به یک مقصد مشترک وجود دارد؟

حل: زیرا در هر مرحله ورودی‌های DR و AR باید از بین ورودی‌های موجود انتخاب شود و برای این کار به مالتی پلکسر نیاز داریم. پس دو مالتی پلکسر یکی برای DR و دیگری برای AR نیاز است. راه‌های دیگر برای این کار استفاده از بافر سه حالت یا مدار گیتی (معادل یک مالتی پلکسر) می‌باشد.

۱۱- با استفاده از جدول ۱-۷، برای ریز اعمال زیر، محتوای ۹ بیت میدان ریز عمل را مشخص کنید.

(الف)

$$AC \rightarrow AC+1, DR \rightarrow DR+1$$

(ب)

$$PC \rightarrow PC+1, DR \leftarrow M[AR]$$

(ج)

$DR \leftarrow AC, AC \leftarrow DR$

حل:

F1	F2	F3				
011	110	000	INCAC	INCDR	NOP	(الف)
000	100	101	NOP	READ	INCPC	(ب)
100	101	000	DRTAC	ACTDR	NOP	(ج)

۱۲- با استفاده از جدول ۷-۱، ریز اعمال سمبلیک زیر را به عبارات انتقال ثباتی و دودویی تبدیل کنید.

الف) READ , INCPC

ب) ACTDR , DRTAC

ج) ARTPC , DRTAC , WRITE

حل:

الف) READ $DR \leftarrow M[AR]$ $F_2 = 100$ Binary
 INCPC $PC \leftarrow PC + 1$ $F_3 = 101$ 000100101

ب) ACTDR $DR \leftarrow AC$ $F_2 = 101$ 100101000
 DRTAC $AC \leftarrow DR$ $F_1 = 100$

ج) ARTPC $PC \leftarrow AR$ $F_3 = 110$
 DRTAC $AC \leftarrow DR$ $F_1 = 100$ غیرممکن هر دو از F_1 استفاده می کنند.
 WRITE $M[AR] \leftarrow DR$ $F_1 = 111$

۱۳- فرض کنید که می خواهیم روال ADD بیان شده در جدول ۷-۲ را به دو ریز دستورالعمل زیر تبدیل کنیم:

ADD:	READ	I	CALL	INDR2
	ADD	U	IMP	FETCH

زیر روال INDR2 چه باید باشد؟

حل:

اگر $I=0$ باشد، عملوند در ریز دستورالعمل اول خوانده شده و در ریز دستورالعمل دوم به AC اضافه می‌شود.

اگر $I=1$ باشد، آدرس مؤثر در DR قرار می‌گیرد و کنترل برنامه به روال INDR2 می‌رود. زیر روال INDR2 باید عملوند را در DR قرار دهد:

```
INDR2: DRTAR  U  IMP  NEXT
        READ   U  RET   --
```

۱۴- در زیر ریز برنامه سمبلیک یک دستورالعمل در کامپیوتر بخش ۳-۷ آمده

است

```
ORG     40
NOP     S  JMP  FETCH
NOP     Z  JMP  FETCH
NOP     I  CALL INDRCT
ARTPC   U  JMP  FETCH
```

الف) عملی که هنگام اجرای دستورالعمل انجام می‌شود را مشخص کنید.

همان‌طور که می‌بینیم ریز برنامه در صورتی که $AC=0$ یا $AC<0$ باشد کاری انجام نمی‌دهد و به روال FETCH پرش می‌کند. در غیر این صورت ($AC>0$) اگر $I=1$ باشد، ابتدا روال INDRCT را فراخوانی می‌کند تا آدرس مؤثر بدست آید و سپس محتوای AR را در PC قرار می‌دهد (انشعاب اگر AC مثبت و غیر صفر باشد).

$$\text{if } (AC > 0) \text{ then } (PC \leftarrow EA)$$
 (EA آدرس مؤثر است.)

ب) چهار ریز دستورالعمل را به فرم معادل دودویی آنها تبدیل کنید.

40: 000 000 000 1000 1000000
 41: 000 000 000 1100 1000000
 (ب) 42: 000 000 000 0101 1000011
 43: 000 000 110 0000 1000000

۱۵- کامپیوتر بخش ۳-۷ دارای ریز برنامه دودویی زیر است.

ریز برنامه دودویی	آدرس
۰۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۰۱۱	۶۰
۱۱۱۱۰۰۰۰۰۱۰۱۱۰۰۰۰۰	۶۱
۰۰۱۰۰۱۰۰۰۱۰۱۰۰۱۱۱۱۱	۶۲
۱۰۱۱۱۰۰۰۱۱۱۱۰۱۱۱۱۰۰	۶۳

الف) آن را به ریز برنامه سمبلیک، مطابق جدول ۲-۷، ترجمه کنید. (FETCH در آدرس ۶۴ و INDRCT در آدرس ۶۷ است.)

(الف)

60:	CLRAC, COM	U	JMP	INDRCT
61:	WRITE, READ	I	CALL	FETCH
62:	ADD, SUB	S	RET	63(NEXT)
63:	DRTAC, INCDR	Z	MAP	60

ب) تمام اشکالاتی را که با اجرای این ریز برنامه در کامپیوتر پیش می‌آید بنویسید.
 ۶۰: نمی‌توان همزمان AC را پاک کرد و متمم کرد. همچنین با پرش به INDRCT کنترل برنامه به خط ۶۱ بر نمی‌گردد.
 ۶۱: نمی‌توان همزمان در حافظه نوشت و از آن خواند. همچنین CALL همانند JMP عمل می‌کند زیرا در پایان FETCH، RET وجود ندارد.
 ۶۲: نمی‌توان همزمان جمع و تفریق انجام داد. همچنین RET بدون توجه به S، اجرا می‌شود.
 ۶۳: MAP بدون توجه به Z و ۶۰ اجرا می‌شود.

۱۶- دستورات زیر را به کامپیوتر بخش ۳-۷ اضافه کنید (EA آدرس مؤثر است): برنامه سمبلیک را برای هر روال مطابق جدول ۲-۷ بنویسید. (توجه کنید که مقدار AC نباید تغییر کند مگر این که دستورالعمل ، یک تغییر در AC را مشخص نماید.)

شرح	شکل سمبلیک	کد عمل	سمبل
AND	$AC \leftarrow AC \wedge M[EA]$	۰۱۰۰	AND
تفریق	$AC \leftarrow AC - M[EA]$	۰۱۰۱	SUB
جمع با حافظه	$M[EA] = M[EA] + AC$	۰۱۱۰	ADM
ماسک کردن بیت‌ها با مقدار حافظه	$AC \leftarrow AC \wedge \overline{M[EA]}$	۰۱۱۱	BTCL
انشعاب در صورت صفر بودن AC	if (AC=0) then (PC ← EA)	۱۰۰۰	BZ
گذر در صورت برابری	if (AC = M[EA]) then (PC ← PC+1)	۱۰۰۱	SEQ
انشعاب اگر AC مثبت و غیر صفر باشد.	if (AC > 0) then (PC ← EA)	۱۰۱۰	BPNZ

حل: برنامه سمبلیک روال‌ها به صورت زیر است:

AND :	ORG 16			
	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
ANDOP :	AND	U	JMP	FETCH
<hr/>				
SUB :	ORG 20			
	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
	SUB	U	JMP	FETCH
<hr/>				
ADM :	ORG 24			
	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
	DRTAC, ACTDR	U	JMP	NEXT
	ADD	U	JMP	EXCHANGE +2

(Table 7.2)

BICL :	ORG 28			
	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
	DRTAC, ACTDR	U	JMP	NEXT
	COM	U	JMP	ANDOP

BZ :	ORG 32			
	NOP	Z	JMP	ZERO
ZERO :	NOP	U	JMP	FETCH
	NOP	I	CALL	INDRCT
	ARTPC	U	JMP	FETCH

SEQ :	ORG 36			
	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
	DRTAC, ACTDR	U	JMP	NEXT
	XOR (or SUB)	U	JMP	BEQ1
BEQ 1 :	ORG 69			
	DRTAC, ACTDR	Z	JMP	EQUAL
	NOP	U	JMP	FETCH
EQUAL :	INC PC	U	JPM	FETCH
BPNZ :	ORG 40			
	NOP	S	JMP	FETCH
	NOP	Z	JMP	FETCH
	NOP	I	CALL	INDRCT
	ARTPC	U	JMP	FETCH

۱۷- یک روال ریز برنامه برای دستورالعمل ISZ (افزایش و عبور در صورت صفر بودن) که در فصل ۵ (جدول ۴-۵) تعریف شده بنویسید. از قالب ریز دستورالعمل بخش ۳-۷ استفاده کنید. توجه کنید که شرط وضعیت $DR = 0$ در میدان CD کامپیوتر بخش ۳-۷ در دسترس نیست، با این وجود می‌توانید AC و DR را با هم تعویض کرده و با بیت Z چک کنید که آیا $AC = 0$ است یا نه.

حل:

ISZ :	NOP	I	CALL	INDRCT
	READ	U	JMP	NEXT
	INCDR	U	JMP	NEXT
	DRTAC, ACTDR	U	JMP	NEXT (or past, INDRCT)
	DRTAC, ACTDR	Z	JMP	ZERO
	WRITE	U	JMP	FETCH
ZERO :	WRITE, INCPC	U	JMP	FETCH

۱۸- روال ریز برنامه سمبلیک را برای دستورالعمل BSA (انشعاب و ذخیره آدرس) که در فصل ۵ تعریف شد بنویسید (جدول ۴-۵). از قالب ریز دستورالعمل بخش ۳-۷ استفاده کنید. تعداد ریز دستورالعمل‌ها را به کمینه ترین حالت برسانید.

حل:

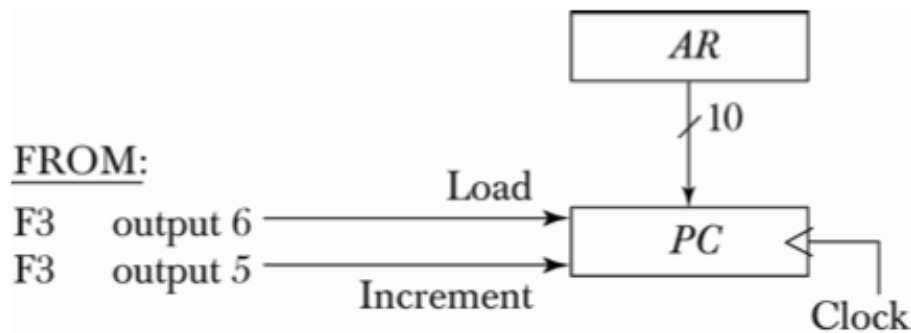
BSA :	NOP	I	CALL	INDRCT
	PCTDR, ARTPC	U	JMP	NEXT
	WRITE, INCPC	U	JMP	FETCH

۱۹- نشان دهید که چگونه خروجی‌های ۵ و ۶ دیکدر F3 در شکل ۷-۷ به شمارنده برنامه PC وصل می‌شوند.

حل: از جدول ۷-۱ داریم:

$$F3=101(5) \quad PC \leftarrow PC+1$$

$$F3=110(6) \quad PC \leftarrow AR$$



۲۰- نشان دهید که چگونه یک میدان ۹ بیتی ریز عمل در ریز دستورالعمل را می‌توان به زیر میدان‌هایی تقسیم کرد تا ۴۶ ریز عمل را مشخص نمایند؟ چند ریز عمل می‌توان در یک ریز دستورالعمل مشخص کرد؟

حل:

یک زیر میدان ۵ بیتی می‌تواند $2^5 - 1(NOP) = 31$ ریز عمل را مشخص کند.

یک زیر میدان ۴ بیتی می‌تواند $2^4 - 1 = 15$ ریز عمل را مشخص کند.

$$\text{بیت } 9 = 5 + 4$$

$$\text{ریز عمل } 46 = 31 + 15$$

۲۱- کامپیوتری دارای ۱۶ ثبات، یک ALU (واحد محاسبه و منطق) با ۳۲ عمل و یک شیفت دهنده با هشت عمل است که همگی به یک سیستم گذرگاه مشترک متصل اند.

الف) برای یک ریز عمل کلمه کنترلی بنویسید.

SRC1	SRC2	DEST	ALU	SHIFT
4	4	4	5	3

ب) تعداد بیت‌ها را در هر میدان کلمه کنترلی مشخص کنید و یک طرح رمزگذاری کلی را ارائه نمایید.

۱۶ ثبات به چهار بیت برای انتخاب نیاز دارد. ۳۲ عمل ALU به ۵ بیت و ۸ عمل شیفت به ۳ بیت برای انکد کردن عمل نیاز دارند. برای هر عمل به سه ثبات نیاز داریم.

ج) بیت‌های کلمه کنترل ریز عمل $R_4 \leftarrow R_5 + R_6$ را نشان دهید.

R5	R6	R4	ADD	SHIFT
۰۱۰۱	۰۱۱۰	۰۱۰۰	۰۰۱۰۰	۰۰۰

۲۲- فرض کنید که منطق ورودی توالی‌گر ریز برنامه شکل ۸-۷ چهار ورودی I_0, I_1, I_2 و T (تست) و سه خروجی S_0, S_1 و L را داراست. عملیات اجرا شده در واحد در جدول زیر لیست شده است. یک مدار منطقی ورودی با استفاده از حداقل تعداد گیت‌ها طراحی کنید.

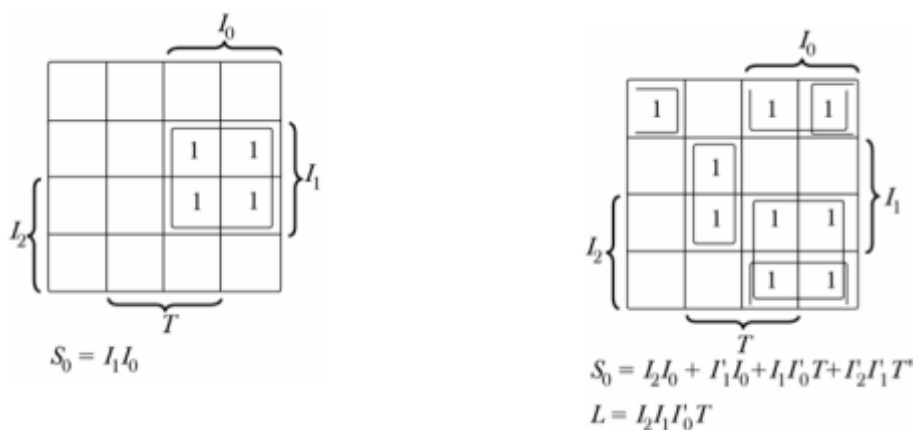
عمل	I_2	I_1	I_0
افزایش CAR اگر $T=1$ ، پرش به AD اگر $T=0$	۰	۰	۰
پرش به AD بدون شرط	x	۰	۱
افزایش بدون شرط CAR	۱	۰	۰
پرش به AD اگر $T=1$ ، افزایش CAR اگر $T=0$	۰	۱	۰

فراخوانی زیر روال اگر $T=1$ ، افزایش CAR اگر

$T=0$
 بازگشت بدون شرط از زیر روال
 نگاشت بدون شرط آدرس خارجی

حل:

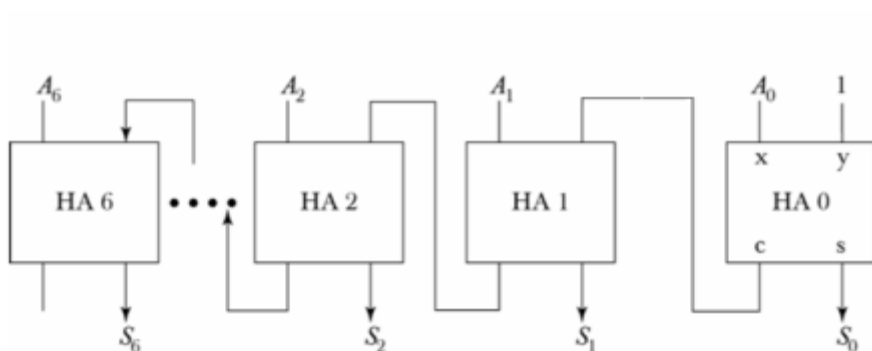
I_2	I_1	I_0	T	S_1	S_0	L	
0	0	0	0	0	1	0	AD1
0	0	0	1	0	0	0	INC(0)
0	0	1	0	0	1	0	AD(1)
0	0	1	1	0	1	0	AD(1)
0	1	0	0	0	0	0	INC(0)
0	1	0	1	0	1	0	AD(1)
0	1	1	0	1	0	0	RET(a)
0	1	1	1	1	0	0	RET(a)
1	0	0	0	0	0	0	INC(0)
1	0	0	1	0	0	0	INC(0)
1	0	1	0	0	1	0	AD(1)
1	0	1	1	0	1	0	AD(1)
1	1	0	0	0	0	0	INC(0)
1	1	0	1	0	1	1	CALL(1)
1	1	1	0	1	1	0	MAP(3)
1	1	1	1	1	1	0	MAP(3)



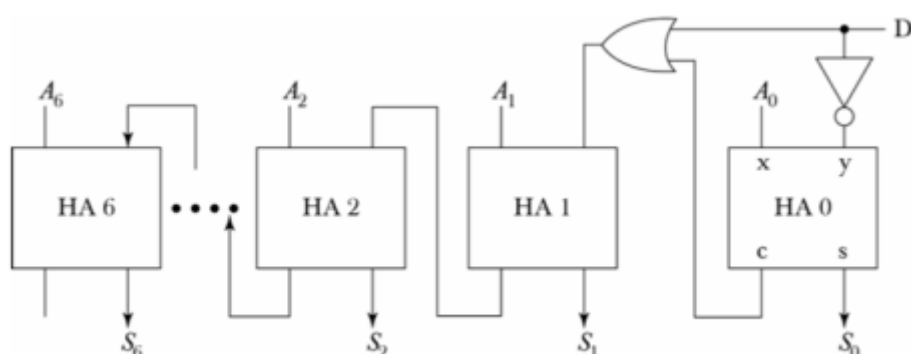
۲۳- یک افزایش گر ترکیبی ۷ بیتی برای توالی گر ریز برنامه شکل ۸ - ۷ طراحی کنید (شکل ۸-۴) با اضافه کردن یک ورودی کنترلی D افزایش گر را تغییر دهید. وقتی $D=0$ است، مدار یک واحد افزایش می‌یابد، ولی وقتی $D=1$ است مدار دو واحد افزایش می‌یابد.

حل:

الف) افزایش گر ترکیبی ۷ بیتی:

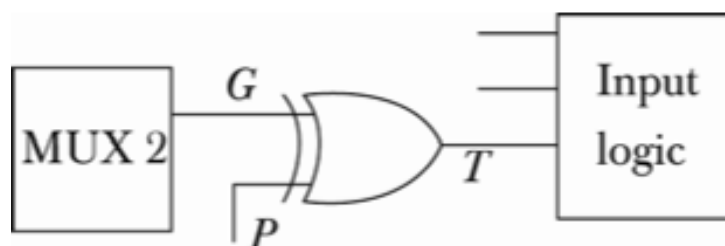


ب) مدار تغییر یافته:



۲۴- بین MUX2 و مدار منطقی ورودی در شکل ۸-۷ یک گیت OR انحصاری قرار دهید. یکی از ورودی‌های گیت از خروجی آزمایشی مالتی پلکسر وارد می‌شود. ورودی دیگر گیت از بیتی با نام P (به معنی قطبیت) در ریز دستورات عمل حاصل از حافظه کنترلی می‌آید. خروجی گیت به ورودی T در مدار منطقی ورودی متصل می‌شود. کنترل قطبیت P چه کاری انجام می‌دهد؟
حل:

P برای تعیین قطبیت بیت وضعیت انتخاب شده به کار می‌رود.



زمانی که $P=0$, $T=G$ زیرا $G \oplus 0 = G$

زمانی که $P=1$, $T=G'$ زیرا $G \oplus 1 = G'$

که در آن G مقدار بیت وضعیت انتخاب شده در MUX2 است.

فصل هشتم: واحد مرکزی پردازش

۱- یک CPU با ساماندهی گذرگاه مشابه با شکل ۸-۲ دارای ۱۶ ثبات ۳۲ بیتی، یک ALU و یک رمزگشای مقصد است.
الف) چند مولتی پلکسر در گذرگاه A وجود دارد و اندازه هر مولتی پلکسر چقدر است.

ب) چند ورودی انتخاب کننده برای MUX A و MUX B لازم است.
ج) در دیکدر چند ورودی و خروجی وجود دارد؟
د) چند ورودی و خروجی داده، از جمله ورودی و خروجی رقم نقلی، در ALU وجود دارد.

ه) با فرض وجود ۳۵ عمل در ALU، کنترل را برای سیستم فرمولبندی کنید.
حل:

الف) رجیسترها ۳۲ بیتی و ۱۶ رجیستر \Rightarrow ۳۲ مالتی پلکسر هرکدام 16×1
ب) ۱۶ رجیستر \Rightarrow هر MUX، ۴ ورودی برای انتخاب یکی از ۱۶ رجیستر
ج) ۱۶ رجیستر \Rightarrow دیکدر ۴ به ۱۶
د) ورودی: $32 + 32 + 1 = 65$
خروجی: $32 + 1 = 33$
ه) ۱۶ رجیستر، بنابراین تعداد بیت ها $\Rightarrow SELA = 4$ و $SELB = 4$ ، $SELD = 4$
۳۵ دستورالعمل \Rightarrow بیت $OPR = 6$

پس،

$$4 + 4 + 4 + 6 = 18 \text{ بیت}$$

SELA	SELB	SELD	OPR
------	------	------	-----

۲- سیستم شکل ۸-۲ دارای تأخیرهای زمانی زیر است: ۳۰ns برای انتشار در MUX، ۸۰ns برای عمل جمع در ALU، ۲۰ns تأخیر در دیکدر مقصدياب و ۱۰ns

برای ورود داده به ثبات مقصد، حداقل سیکل زمانی برای استفاده به عنوان پالس ساعت چقدر است؟

حل: طبق مدار، مدار ALU به نتیجه خروجی MUXها بستگی دارد. همچنین نوشتن در رجیستر مقصد به خروجی ALU نیاز دارد. طبق صورت سؤال:

خروجی ALU در (ns) ۱۰۰ آماده است.

خروجی دیگر در (ns) ۲۰ آماده است.

$$\text{Max}(100, 20) = 100 \Rightarrow$$

$$30 + 80 + 10 = 120 \text{ n(sec) پس}$$

در حقیقت خروجی دیگر، حین تشکیل خروجی ALU، آماده شده است.

۳- کلمه کنترلی را که باید به پردازشگر شکل ۲-۸ اعمال شود تا ریز اعمال زیر را پیاده سازی کند، مشخص کنید.

$$\text{الف) } R_1 \leftarrow R_2 + R_3$$

$$\text{ب) } R_4 \leftarrow R_4$$

$$\text{ج) } R_5 \leftarrow R_5 - 1$$

$$\text{د) } R_6 \leftarrow \text{shL} R_1$$

$$\text{ه) } R_7 \leftarrow \text{input}$$

حل: جدول زیر را تشکیل می دهیم:

دستورالعمل	SELA	SELB	SELD	OPR	کلمه کنترل دودویی
$R_1 \leftarrow R_2 + R_3$	R_2	R_3	R_1	ADD	۰۱۰ ۰۱۱ ۰۰۱۰۰۰۱۰
$R_4 \leftarrow \bar{R}_4$	R_4	—	R_4	COMA	۱۰۰ ۰۰۰ ۱۰۰۰۱۱۱۰
$R_5 \leftarrow R_5 - 1$	R_5	—	R_5	DECA	۱۰۱ ۰۰۰ ۱۰۱۰۰۱۱۰
$R_6 \leftarrow \text{SH1} R_1$	R_1	—	R_6	SHLA	۰۰۱ ۰۰۰ ۱۱۰۱۱۰۰۰
$R_7 \leftarrow \text{Input}$	Iuput	—	R_7	TSFA	۰۰۰ ۰۰۰ ۱۱۱۰۰۰۰۰

۴- ریز عملیات اجراشونده در پردازشگر شکل ۸-۲ را وقتی که کلمات کنترل ۱۴ بیتی زیر اعمال می‌شود معین کنید.

الف) ۰۰۱۰۱۰۰۱۱۰۰۱۰۱

ب) ۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰

ج) ۰۱۰۰۱۰۰۱۰۰۱۱۰۰

د) ۰۰۰۰۰۱۰۰۰۰۰۰۱۰

هـ) ۱۱۱۱۰۰۰۱۱۱۰۰۰۰

حل: روند جواب این سؤال، عکس روند جواب سؤال قبل است. یعنی با داشتن کلمه کنترل دودویی، می‌بایست دستورالعمل مناسب را بنویسیم. پس جدول زیر را تشکیل می‌دهیم.

ریز دستورالعمل	OPR	SELD	SELB	SELA	کلمه کنترلی دودویی
$R_3 \leftarrow R_1 - R_2$	SUB	R_3	R_2	R_1	الف) ۰۰۱ ۰۱۰ ۰۱۱ ۰۰۱۰۱
$\text{Output} \leftarrow \text{Input}$	TSFA	None	Input	Input	ب) ۰۰۰ ۰۰۰ ۰۰۰ ۰۰۰۰
$R_2 \leftarrow R_2 \oplus R_2$	XOR	R_2	R_2	R_2	ج) ۰۱۰ ۰۱۰ ۰۱۰ ۰۱۱۰۰
$\text{Output} \leftarrow \text{Input} + R_1$	ADD	None	R_1	Input	د) ۰۰۰ ۰۰۱ ۰۰۰ ۰۰۰۱۰
$R_3 \leftarrow \text{Shr } R_7$	SHRA	R_3	R_4	R_7	هـ) ۱۱۱ ۱۰۰ ۰۱۱ ۱۰۰۰۰

۵- فرض شود در پشته شکل ۸-۳، $sp=000000$ باشد. چند تعداد داده در

پشته وجود دارد اگر:

الف) $\text{FULL}=1, \text{EMPTY}=0$

ب) $\text{FULL}=0, \text{EMPTY}=1$

حل:

الف) پشته با ۶۴ داده پر شده است.

ب) پشته خالی است (داده‌ای در پشته وجود ندارد)

۶- پشته طوری سازمان یافته است که همیشه SP به مکان بعدی خالی در پشته اشاره می‌نماید این به آن معنی است که مقدار اولیه SP در شکل ۴-۸ می‌تواند ۴۰۰۰ بوده و اولین داده در پشته در مکان ۴۰۰۰ از پشته ذخیره شود. برای اعمال push و pop عملیات جزئی لازم را بنویسید.

حل: در هر زمان stack pointer به خانه خالی بالای داده‌ها اشاره می‌کند. پس:

push: $1-M[SP] \leftarrow DR$

pop: $1-SP \leftarrow SP+1$

$2-SP \leftarrow SP-1$

$2-DR \leftarrow M[SP]$

۷- عبارات محاسباتی را از میانوندی به RPN تبدیل کنید.

الف) $A*B*C*D+E*F$

ب) $A*B+A*(B*D+C*E)$

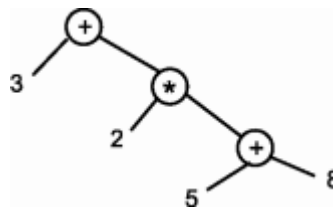
ج) $A+B*[C*D+E*(F+G)]$

د) $A*[B+C*(D+E)]$

$F*(G+H)$

حل: در تبدیل عبارات میانوندی به عبارات RPN، ابتدا درخت عبارت، براساس اولویت پایین به بالا را رسم کرده و سپس پیمایش Postorder انجام می‌دهیم. (پیمایش post order ابتدا فرزند سمت چپ، سپس فرزند سمت راست و سپس پدر را پیمایش می‌کند.)

مثال:



$3+2*(5+8)$

\Rightarrow پیمایش

$3258+*+$

به همین ترتیب برای این سؤال داریم:

الف) $AB*CD*EF*++$

ب) $AB*ABD*CE*++$

ج) $FG + E * CD * + B * A +$

د) $ABCDE + * + * FGH + * /$

۸- عبارت محاسباتی زیر را از RPN به میانوندی تبدیل کنید.

الف) $ABCDE + * - /$

ب) $ABCDE * / - +$

ج) $ABC * / D - EF / +$

د) $ABCDEFG + * + * + *$

حل: روند پاسخ این سؤال، عکس سؤال قبل است. یعنی تبدیل فرم RPN به فرم میانوندی عبارت در این سؤال نیز برای عبارت یک پشته ترتیب داده و از چپ به راست عمل می‌کنیم:

در صورت برخورد با عملوند، آن را در پشته push می‌کنیم.

در صورت برخورد با عملگر، دو عنصر بالای پشته را چپ و راست عملگر قرار

می‌دهیم و پس عبارت را محاسبه کرده، نتیجه را در پشته push می‌کنیم.

پس داریم:

الف) $ABCDE + * - / \Rightarrow \frac{A}{B - (D + E) * C}$

ب) $ABCDE * / - + \Rightarrow A + B - \frac{C}{D * E}$

ج) $ABC * / D - EF / + \Rightarrow \frac{A}{B * C} - D + \frac{E}{F}$

د) $ABCDEFG + * + * + * \Rightarrow (((F + G) * E + D) * C + B) * A$

۹- عبارت محاسباتی عددی زیر را به RPN تبدیل و عملیات پشته را برای

محاسبه نتیجه نشان دهید.

$[(3+4)(10(2+6)+8)]$

حل: ابتدا عبارت را به فرم RPN تبدیل کرده و سپس روند محاسبه عبارت با پشته

را اجرا می‌کنیم.

$$(3+4)[10(2+6)+8]=616$$

$$\xrightarrow{\text{RPN}} 34+26+10*8+*$$

حال داریم: از چپ به راست در صورت برخورد با هر عملوند، آن را در پشته push کرده و در صورت برخورد با عملگر، عمل عملگر را با دو عنصر بالای پشته انجام می‌دهیم. در هر ستون، بالاترین داده، عنصر بالای پشته است. ترتیب برخورد با عملوندها و عملگرها در جدول از چپ به راست است.

					۶		۱۰		۸		
پشته		۴		۲	۲	۸	۸	۸۰	۸۰	۸۸	
	۳	۳	۷	۷	۷	۷	۷	۷	۷	۷	۶۱۶ نتیجه
	۳	۴	+	۲	۶	+	۱۰	*	۸	+	*

در اینجا دو نکته مهم وجود دارد.

- ۱- ما به تعداد عملوندها، در پشته عمل push انجام می‌دهیم.
- ۲- اینکه ظرفیت پشته را چقدر در نظر بگیریم «در اینجا ما ظرفیت را ۳ در نظر گرفته ایم» بستگی به آزمون و خطا در بدست آوردن مقدار عبارت با پشته دارد. در اینجا ظرفیت ۳ برای پشته کافی بوده است.

۱۰- سازمان یک حافظه به شکل FIFO (اولین ورودی اولین خروجی) است به این مفهوم که اولین داده ذخیره شده، اولین داده بازیابی شده است. نشان دهید چگونه یک حافظه FIFO با سه شمارنده کار می‌کند. یک شمارنده نوشتن، WC، آدرس نوشتن در حافظه را نگه می‌دارد. یک شمارنده خواندن RC آدرس خواندن در حافظه را حفظ می‌نماید. یک شمارنده فضای موجود در حافظه، ASC، تعداد کلمات ذخیره شده در FIFO را معین می‌نماید. ASC برای هر کلمه ذخیره شده یک واحد اضافه می‌شود و برعکس برای کلمات دریافت شده یک واحد کم می‌گردد.

حل: ساختار FIFO بیانگر ساختار «صف» در ساختمان داده‌ها می‌باشد.

با توجه به سؤال، هر گاه عمل write صورت می‌پذیرد. یک واحد به شمارنده ASC اضافه می‌گردد و هر گاه عمل Read صورت می‌پذیرد، یک واحد از شمارنده ASC کم می‌گردد.

پس:

Write : IF(full=0)

$M[WC] \leftarrow DR$

$WC \leftarrow WC + 1$

$ASC \leftarrow ASC + 1$

Read : IF(empty=0)

$DR \leftarrow M[RC]$

$RC \leftarrow RC + 1$

$ASC \leftarrow ASC - 1$

از دو پرچم full و empty برای چک کردن پر یا خالی بودن صف استفاده می‌شود.

۱۱- یک کامپیوتر دارای دستورالعمل‌های ۳۲ بیتی و آدرس ۱۲ بیتی است. اگر تعداد ۲۵۰ دستورالعمل دو آدرس وجود داشته باشد، چه تعداد دستورالعمل تک آدرسی را می‌توان تشکیل داد؟

حل: دستورالعمل‌ها ۳۲ بیتی

آدرس‌ها ۱۲ بیتی

$$256 - 250 = 6 \leftarrow \text{opcode } 8 \text{ بیتی} \leftarrow 2^8 = 256$$

پس:

$$8 + 12 + 12 = 32 \text{ بیت}$$

Op-code	Address 1	Address 2
---------	-----------	-----------

دستورالعمل‌های دو آدرسی

پس:

Op-code	Address
---------	---------

→ $24576 = 6 \times 2^{12}$ = ماکزیمم تعداد دستورالعمل‌های تک آدرسی

لازم به توضیح است این سؤال در ترجمه کتاب موریس مانو، لحاظ نشده است اما در متن انگلیسی آن مطرح شده است.

۱۲- برنامه‌ای بنویسید که عبارت ریاضی زیر را محاسبه کند.

$$X = \underline{A - B + C * (D * E - F)}$$

$$G + H * K$$

الف) با استفاده از کامپیوتر با ثبات‌های عمومی و دستورات سه آدرسه

ب) با استفاده از کامپیوتر با ثبات‌های عمومی و دستورات دو آدرسه

ج) با استفاده از کامپیوتر نوع انباره‌ای با دستورات یک آدرسه

د) با استفاده از کامپیوتر با ساماندهی پشته با دستورات صفر آدرسه

حل: ابتدا فرم RPN را با توجه به مباحث مطرح شده به دست می‌آوریم. سپس طبق موارد گفته شده عمل می‌کنیم.

فرم RPN: $AB - CDE * F - * + GHK * + /$

الف) با دستورات سه آدرسی	ب) با دستورات دو آدرسی	ج) با دستورات تک آدرسی	د) با دستورات صفر آدرسی
SUB R ₁ ,A,B	MOV R ₁ ,A	Load A	با استفاده از فرم RPN
MUL R ₂ ,D,E	SUB R ₁ ,B	SUB B	Push A
SUB R ₂ ,R ₂ ,F	MOV R ₂ ,D	Store X	Push B

SUB	Load	D	MUL R ₂ ,E	MUL R ₂ , R ₂ , C
Push C	MUL	E	SUB R ₂ ,F	ADD R ₁ , R ₁ , R ₂
Push D	SUB	F	MUL R ₂ ,C	MUL R ₂ ,H,K
Push E	MUL	C	ADD R ₁ ,R ₂	ADD R ₂ ,R ₂ ,G
MUL	ADD	X	MOV R ₂ ,H	DIV X,R ₁ ,R ₂
Push F	Store	X	MUL R ₂ ,K	
SUB	Load	H	ADD R ₂ ,G	
MUL	MUL	K		
ADD	ADD	G	DIV R ₁ ,R ₂	
Push G	Store	T	MOV X,R ₁	
Push H	Load	X		
Push K	DIV	T		
MUL	Store	X		
ADD				
DIV				
POP X				

۱۳- واحد حافظه کامپیوتر ۲۵۶K کلمه ۳۲ بیتی دارد. کامپیوتر دارای قالب دستورات چهار میدانی است: میدان کد عملیاتی ، میدان روش آدرس دهی برای هفت روش موجود، میدان آدرس ثبات برای انتخاب یکی از ۶۰ ثبات پردازشگر، و آدرس حافظه، قالب دستور و تعداد بیت ها در هر میدان را اگر دستور در یک کلمه حافظه باشد معین کنید.

حل:

ظرفیت حافظه: $2^{18} = 2^{10} \times 2^8 = 256k \leftarrow 18$ خط آدرس
 تعداد رجیسترها: 60 , $6 = 2^6 = 64$ بیت مشخص کردن رجیستر
 تعداد مدهای آدرس دهی: 7 , $8 = 2^3 \leftarrow 3$ بیت مشخص کردن مد آدرس دهی
 پس:

Address	Resister	Mode	op-code
۱۸	۶	۳	۵
= ۳۲			

۱۴- یک دستور دو کلمه‌ای در حافظه و در آدرسی که با سمبل W مشخص شده ذخیره شده است. میدان آدرس دستور، (ذخیره شده در W+1) با سمبل Y معین شده است. عملوند بکار رفته در هنگام اجرای دستور در آدرس Z قرار دارد. یک ثبات اندیس دارای مقدار X است. نشان دهید که چگونه Z از سایر آدرس‌ها محاسبه می‌شود اگر که روش آدرس دهی دستور به شکل زیر باشد.

الف) مستقیم ب) غیرمستقیم ج) نسبی د) اندیس دار

حل: آدرس مؤثر در Z

الف) مستقیم: $Z = Y$

ب) غیرمستقیم: $Z = M[Y]$

ج) نسبی: $Z = Y + W + Z$

↓
آدرس بعدی PC

و) اندیس دار: $Z = Y + X$

۱۵- یک دستورالعمل انشعاب از نوع نسبی در آدرس معادل دهدهی ۷۵۰ از یک حافظه ذخیره شده است انشعاب به آدرس ۵۰۰ صورت گرفته است.
 الف) مقدار میدان آدرس نسبی دستور (معادل دهدهی) چقدر است؟
 ب) مقدار میدان آدرس نسبی را به دودویی در ۱۲ بیت نشان دهید. (چرا باید عدد به فرم مکمل ۲ باشد؟)
 ج) مقدار دودویی در PC را پس از فاز برداشت (واکشی) بدست آورید و

مقدار ۵۰۰ را محاسبه کنید، سپس نشان دهید که مقدار دودویی در PC بعلاوه آدرس نسبی محاسبه شده در قسمت (ب) برابر عدد دودویی ۵۰۰ است.

حل: در خانه ۷۰۰ هستیم ← $PC=751$ حال با چه آدرس نسبی باید به آدرس ۵۰۰ برویم؟
(الف)

$$500 - 751 = -251 \text{ : آدرس نسبی}$$

(ب)

$$-251 = 2's \text{ complement } (251) = 111100000101$$

(ج)

$$PC=751=00111101111$$

$$RA : -251=111100000101$$

$$\Rightarrow 500=PC+RA=000111110100$$

۱۶- واحد کنترل وقتی که یک دستور با آدرس غیرمستقیم را برداشت و اجرا می‌کند، چند بار به حافظه مراجعه می‌نماید اگر الف) دستور از نوع محاسباتی بوده و یک عملوند را از حافظه لازم داشته باشد ب) دستور از نوع انشعاب باشد.
حل:

دستورالعمل حسابی: ۳ = دریافت عملوند از حافظه + دریافت آدرس مؤثر +

دریافت دستورالعمل از حافظه

دستورالعمل انشعاب: ۲ = دریافت آدرس مؤثر و دادن آن به PC + دریافت

دستورالعمل از حافظه

۱۷- میدان آدرس یک دستور با آدرس اندیس دار چه باشد تا مشابه یک دستور

غیرمستقیم ثباتی گردد؟

حل: طبق تعریف داریم:

مد آدرس دهی ثباتی غیرمستقیم: محتویات $R_1 = \text{آدرس مؤثر}$ ، $R_1 = \text{ثبات}$

مد آدرس دهی شاخص دار: آدرس موجود در دستورالعمل + محتویات R_1 = آدرس مؤثر
 R_1 = ثبات

بنابراین برای برآورده کردن خواسته سؤال: ϕ = آدرس موجود در دستورالعمل

۱۸- یک دستور در مکان ۳۰۰ با میدان آدرسش در مکان ۳۰۱ ذخیره شده است.

میدان آدرس دارای مقدار ۴۰۰ است. یک ثبات پردازشگر R_1 نیز حاوی عدد ۲۰۰ است. آدرس مؤثر را اگر روش آدرس دهی دستور الف) مستقیم ب) بلافصل ج) نسبی د) غیرمستقیم ثباتی ه) شاخص دار با R_1 به عنوان ثبات شاخص باشد به دست آورید.

حل:

الف) مد آدرس دهی مستقیم: آدرس مؤثر = ۴۰۰

ب) مد آدرس دهی بلافصل: آدرس مؤثر = ۳۰۱

ج) مد آدرس دهی نسبی: آدرس مؤثر = ۷۰۲ = ۴۰۰ + ۳۰۲

د) مد آدرس دهی ثباتی غیرمستقیم: آدرس مؤثر = ۲۰۰

ه) مد آدرس دهی شاخص دار: آدرس مؤثر = ۶۰۰ = ۴۰۰ + ۲۰۰

۱۹- با فرض داشتن یک کامپیوتر ۸ بیتی، یک جمع با دقت مضاعف برای اعداد

بدون علامت ۳۲ بیتی زیر با استفاده از دستور جمع با رقم نقلی را انجام دهید. هر بایت به صورت عدد دو رقمی مبنای ۱۶ می باشد.

$$(6E \ C3 \ 56 \ 78) + (13 \ 55 \ 6B \ 8F)$$

حل: برای این عبارت، دو رقم دو رقم شانزده شانزدهی را جمع کرده و در هر

جمع، رقم نقلی را محاسبه کرده به بعدی می دهیم:

$$\begin{array}{cccc} 6E & C3 & 56 & 7A \\ \hline 13 & 55 & 6B & 8F \\ \hline 82 & 18 & C2 & \phi 9 \end{array}$$

$$C=1 \ C=0 \ C=1 \ C=0$$

رقم نقلی ورودی \rightarrow رقم نقلی خروجی = *

۲۰- برای دو رشته اعداد دودویی ۱۰۰۱۱۱۰۰ و ۱۰۱۰۱۰۱۰ ، اعمال منطقی AND ، OR ، XOR را انجام دهید.

حل:

$$\begin{array}{r} \text{AND : } 10011100 \\ \quad 10101010 \\ \hline 10001000 \end{array}$$

$$\begin{array}{r} \text{OR : } 10011100 \\ \quad 10101010 \\ \hline 10111110 \end{array}$$

$$\begin{array}{r} \text{XOR : } 10011100 \\ \quad 10101010 \\ \hline 00110110 \end{array}$$

۲۱- عدد ۱۶ بیتی ۱۰۰۱۱۰۱۰۱۱۰۰۱۱۰۱۰۱ داد تا:

الف) ۸ بیت اول ۰ شود.

ب) هشت بیت آخر ۱ شود.

ج) هشت بیت وسط مکمل شود.

حل: فرض شود $A = 1001101011001101$

الف) برای ۰ کردن تعدادی از بیت‌ها از عملگر AND استفاده می‌کنیم. (AND Mask) $A = (1001101011001101)$ مکان‌های بدون تغییر ۱ و مکان‌های مورد نظر برای ۰ شدن، ۰ در نظر می‌گیریم. پس:

$$\text{AND (A, 0000000011111111)}$$

ب) برای ۱ کردن تعدادی از بیت‌ها از عملگر OR استفاده می‌کنیم.

مکان‌های بدون تغییر ۰ و مکان‌های مورد نظر برای ۱ شدن، ۱ در نظر می‌گیریم.

پس:

$OR(A, 0000\ 00001111\ 1111)$

ج) برای مکمل کردن تعدادی از بیت‌ها از عملگر XOR استفاده می‌کنیم.

مکان‌های بدون تغییر، ۰ و مکان‌های مورد نظر برای مکمل کردن، ۱ در نظر

می‌گیریم پس:

$XOR(A, 0000\ 1111\ 1111\ 0000)$

۲۲- یک ثبات ۸ بیتی دارای مقدار ۰۱۱۱۱۰۱۱ است و بیت نقلی نیز ۱ است.

هشت عمل شیفت موجود در جدول ۸-۹ را روی آنها پیاده کنید. هر بار مقدار اولیه

عدد داده شده بالا باشد.

حل: داده اولیه: ۰۱۱۱۱۰۱ و $Carry = 1$ داریم:

SHR : 00111101 (شیفت به راست منطقی)

SHL : 11110110 (شیفت به چپ منطقی)

SHRA : 00111101 (شیفت به راست حسابی)

تغییر بیت علامت ← رخداد سرریز < SHLA : 11110110 (شیفت به چپ

حسابی)

ROR : 10111101 (چرخش به راست)

ROL : 11110110 (چرخش به چپ)

RORC : 10111101 (چرخش به راست با نقلی)

ROLC : 11110111 (چرخش به چپ با نقلی)

۲۳- اعداد علامت‌دار روبرو را به شکل اعداد دودویی را به صورت ۸ بیت

نمایش دهید، ۸۳- و ۸۳+ و ۶۳- و ۶۳+

الف) جمع $(+63) + (-83)$ را به دودویی انجام و نتیجه را تفسیر کنید.

ب) تفریق $(+83) - (-68)$ را به دودویی انجام و نشان دهید که سرریز رخ

می دهد.

ج) عدد دودویی ۶۸- را یکبار به راست شیفت دهید و مقدار شیفت داده شده را به دهدهی بدست آورید.

د) عدد دودویی ۸۳- را یکبار به چپ شیفت دهید و ببینید آیا سرریز وجود دارد یا نه؟

حل: اعداد منفی را مکمل ۲ خواهیم کرد.

+83: 01010011

+68: 01000100

-83: 10101101

-68: 10111100

(الف)

-83 10101101

+68 01000100

-15 11110001

ب) تشخیص سرریز با XOR کردن دو رقم نقلی آخر صورت می پذیرد.

$$-68 - (+83) = -68 + (-83)$$

$$\Rightarrow -68 \quad 10111100$$

$$\begin{array}{r} -83 \quad 10101101 \\ \hline 01101001 \end{array}$$

با توجه به دو رقم نقلی آخر، سرریز رخ داده است.

ج)

$$-68: 1011100 \Rightarrow 11011110 : -34$$

د)

$$-83 : 10101101 \Rightarrow 01011010 : -166$$

بیت علامت تغییر کرده است. پس سرریز رخ داده است.

۲۴- نشان دهید مداری که خرجی ۰ را در شکل ۸-۸ چک می کند، یک

گیت NOR است.

حل: برای تشخیص ۰ بودن جواب ALU، می‌بایست بیان کنید که تک تک بیت‌های جواب ۰ هستند.

$Z = F'_0 + F'_1 + F'_2 + F'_3 + F'_4 + F'_5 + F'_6 + F'_7 = (F_0 + F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7)'$:NOR
۲۵- یک کامپیوتر ۸ بیت دارای یک ثبات R است. مقادیر بیت‌های وضعیت C و S و Z و V (شکل ۸-۸) را پس از اجرای هر یک از دستورات زیر معین کنید. مقدار اولیه R در هر مورد ۷۲ در مبنای شانزده است. اعداد زیر نیز در مبنای شانزده هستند.

الف) عملوند بلافصل C6 را با R جمع کنید.

ب) عملوند بلافصل 1E را با R جمع کنید.

ج) عملوند بلافصل 9A را از R کم کنید.

د) R را با R XOR نمایید.

حل: متذکر می‌شویم مواردی که رخداد سرریز را موجب می‌شوند، فعلاً جمع و منها و شیفت حسابی به راست هستند. برای یافتن مقدارها، دو رقم نقلی آخر محاسبه و بیت علامت محاسبه اهمیت دارند.

$$\begin{array}{r} 01110010 \\ +00011110 \\ \hline 10010000 \end{array} \quad \text{ب)}$$

$$C=0, S=1, Z=0, V=1$$

$$\begin{array}{r} 01110010 \\ + \\ 11000110 \\ \hline 00111000 \end{array} \quad \text{الف)}$$

$$C=1, S=0, Z=0, V=0$$

$$\text{ج) } R - 9A = R + (-9A)$$

$$\begin{array}{r} 01110010 \\ \text{AND } 10001100 \\ \hline 00000000 \end{array} \quad \text{د)}$$

$$\begin{array}{r} -9A = 01100110 \\ +01110010 \\ \hline 11011000 \end{array}$$

$$C=0, S=0, Z=1, V=0 \qquad C=0, S=1, Z=0, V=1$$

(هـ)

$$\begin{array}{r} 01110010 \\ \text{XOR} + 01110010 \\ \hline 00000000 \end{array}$$

$$C=0, S=0, Z=1, V=0$$

۲۶- دو عدد بدون علامت توسط رابطه $A-B$ با هم مقایسه شده‌اند. بیت نقلی به عنوان بیت قرض پس از مقایسه در بیشتر کامپیوترهای تجاری در نظر گرفته می‌شود، به طوری که اگر $A < B$ باشد $C=1$ است. نشان دهید که نسبت اندازه A نسبت به B از وضعیت C و Z طبق جدول مربوط به این مسئله و (جدول ۸-۱۱) قابل استنتاج است.

حل: طبق سؤال، وضعیت‌های شرطی جدول ۸-۱۱، برای A و B بدون علامت، از طریق بیت‌های C و Z تعیین می‌گردد. (مقایسه با $(A-B)$) پس داریم:

$$C=1 \Leftrightarrow A < B \Rightarrow C=0 \Leftrightarrow A \geq B$$

$$Z=1 \Leftrightarrow A = B \Rightarrow Z=0 \Leftrightarrow A \neq B$$

بنابراین:

$$A > B \Leftrightarrow (A \geq B) \text{ AND } (A \neq B) : C=0 \text{ AND } Z=0 : (C'Z')=1$$

$$A \leq B \Leftrightarrow (A < B) \text{ OR } (A = B) : C=1 \text{ OR } Z=1 : (C+Z)=1$$

پس:

رابطه	وضعیت بیت‌ها
$A > B$	$(C'Z')=1$
$A \geq B$	$C=0$

$$\begin{array}{ll}
 A < B & C = 1 \\
 A \leq B & (C + Z) = 1 \\
 A = B & Z = 1 \\
 A \neq B & Z = 0
 \end{array}$$

۲۷- دو عدد علامت‌دار به شکل مکمل ۲ با هم طبق رابطه A-B مقایسه شده‌اند. با توجه به نتیجه مقایسه بیت‌های وضعیت S و Z و V و ۰ و ۱ خواهند شد. (دقت کنید که اگر سرریز رخ دهد تغییر علامتی وجود خواهد داشت). نشان دهید که اندازه نسبی A نسبت به B با توجه به بیت‌های وضعیت در جدول مربوط به این مسئله (و جدول ۸-۱۱) قابل استنتاج است.

حل: طبق سؤال، وضعیت‌های شرطی جدول ۸-۱۱، برای A و B علامت‌دار از طریق بیت‌های S، V و Z تعیین می‌گردد.

داریم:

$$\begin{aligned}
 A \geq B &\Rightarrow (A - B) \geq 0 \Rightarrow \begin{cases} S = 1, V = 1 \\ S = 0, V = 0 \end{cases} \Rightarrow (S \oplus V) = 0 \\
 A < B &\Rightarrow \begin{cases} S = 1, V = 0 \\ S = 0, V = 1 \end{cases} \Rightarrow (S \oplus V) = 1 \\
 A > B &\Rightarrow (A \geq B) \text{ AND } (A \neq B) \Rightarrow (S \oplus V) = 0 \text{ and } Z = 0 \\
 A \leq B &\Rightarrow (A < B) \text{ OR } (A = B) \Rightarrow (S \oplus V) = 1 \text{ OR } Z = 1
 \end{aligned}$$

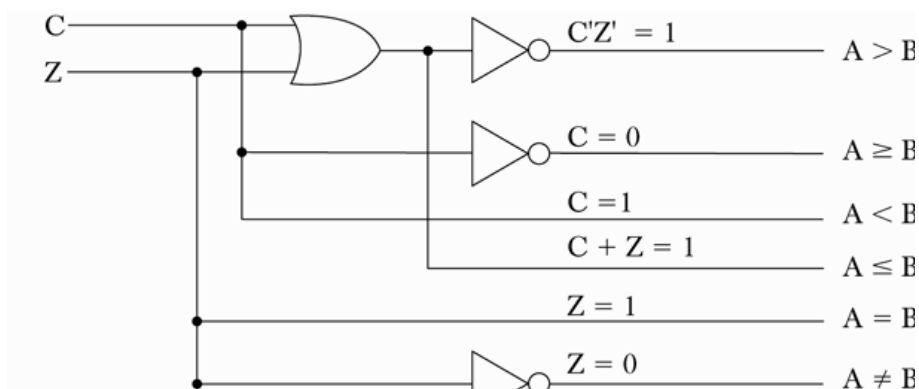
پس:

رابطه	وضعیت بیت‌ها
$A > B$	$(S \oplus V) = 0 \text{ and } Z = 0$
$A \geq B$	$(S \oplus V) = 0$

$A < B$	$(S \oplus V) = 1$
$A \leq B$	$(S \oplus V) = 1 \quad \text{or} \quad Z = 1$
$A = B$	$Z = 1$
$A \neq B$	$Z = 0$

۲۸- لازم است یک مدار دیجیتال با چهار ورودی C و S و Z و V با ۱۰ خروجی طراحی شود، که هر یک مربوط به یک شرط انشعاب در مسئله ۲۶-۸ و ۲۷-۸ است. (حالات مساوی و نامساوی در هر دو جدول یکی است.) نمودار منطقی مدار با استفاده از دو گیت OR، یک XOR و پنج معکوس کننده را رسم کنید.

حل: برای طراحی مدار منطقی رابطه‌های گفته شده، از نتایج بدست آمده در پاسخ سؤالات ۲۶ و ۲۷ استفاده کرده و با استفاده از بیت‌های S ، V ، Z ، C به عنوان ورودی، مدار را طراحی می‌نماییم.



(ج) با توجه به قسمت الف $C=0, Z=0, S=1, V=0$

(د) رابطه‌های انشعابی برقرار:

$$C=0 \Rightarrow \text{BNC}$$

$$Z=0 \Rightarrow \text{BNZ}$$

$$S=1 \Rightarrow \text{BM}$$

$$V=0 \Rightarrow \text{BNV}$$

۳۰- برنامه‌ای در یک کامپیوتر دو عدد بدون علامت A و B را با انجام عمل $A-B$ و تصحیح بیت‌های وضعیت مقایسه می‌کند. اگر $A=01000001$ و $B=10000100$ باشد،

(الف) اختلاف را محاسبه و نتیجه دودویی را تفسیر کنید.

(ب) مقادیر بیت‌های وضعیت C (قرض) و Z را معین کنید.

(ج) دستورات انشعاب شرطی را از جدول ۸-۱۱ که اتفاق می‌افتند را لیست کنید.

حل:

(الف)

$$A-B = A + (-B)$$

$$A = 01000001 \quad 65$$

$$B = 10000100 \quad 132$$

$$A-B \quad \underline{10111101} \quad -67$$

(ب)

رقم قرض در تفریق $Z=0, C=1$

(ج)

$$A=65, B=132 \Rightarrow A < B \Rightarrow \text{BL, BLE, BNE}$$

↓
کوچکتر
کوچکتر مساوی
مخالف

۳۱- برنامه‌ای در یک کامپیوتر دو عدد علامت‌دار A و B را با پیش قالب بندی عمل $A-B$ تصحیح بیت‌های وضعیت مقایسه می‌کند. اگر $A=01000001$ و

B = 10000100 باشند.

الف) اختلاف را محاسبه و نتیجه دودویی را تفسیر کنید.

ب) مقادیر بیت‌های S و Z و V را معین کنید.

ج) دستورات انشعاب شرطی را از جدول ۸-۱۱ که اتفاق می‌افتد لیست کنید.

حل: A و B علامت‌دار هستند. $A - B = A + (-B)$

$\begin{array}{r} A = 01000001 : 65 \\ - B = 10000100 : 124 \\ \hline A - B \quad 1011110 \end{array}$	$\begin{array}{r} 01000001 \\ + 01111100 \text{ (الف)} \\ \hline 10111101 \end{array}$
--	--

ب) با توجه به قسمت الف: $S = 1, Z = 0, V = 1$

ج)

$$A = 65, B = -124 \Rightarrow A > B \Rightarrow BGT, BGE, BNE$$

مخالف کوچکتر مساوی بزرگتر

۳۲- محتویات بالاترین مکان حافظه یک پشته ۵۳۲۰ است. محتوای اشاره‌گر

پشته (SP) نیز ۳۵۶۰ می‌باشد. یک دستور فراخوانی دو کلمه‌ای در آدرس ۱۱۲۰

حافظه قرار دارد و در آدرس ۱۱۲۱ هم میدان آدرس آن یعنی ۶۷۲۰ ذخیره شده

است. محتوای PC و SP و بالاترین مکان پشته چیست؟

الف) قبل از برداشتن دستور فراخوانی از حافظه

ب) پس از اجرای دستور فراخوانی

ج) پس از بازگشت از زیر روال

حل:

	PC	SP	مقدار بالای TOS → stack
مقدار ابتدایی	۱۱۲۰	۳۵۶۰	۵۳۲۰
بعد از فراخوانی "Call"	۶۷۲۰	۳۵۵۹	۱۱۲۲

۳۵۶۰ ۵۳۲۰ ۱۱۲۲ بعد از بازگشت “Return”

۳۳- اختلاف عمده بین دستور انشعاب، فراخوانی و وقفه برنامه چیست؟

حل: دستورالعمل انشعاب: انشعاب به قسمتی از حافظه بدون بازگشت به قسمت

قبلی

زیر روال و فراخوانی آن: انشعاب به زیر روال و سپس بازگشت به برنامه اصلی
وقفه نرم‌افزاری: انشعاب به یک روتین اجرای وقفه «آدرس وقفه به صورت
سخت‌افزاری مشخص می‌گردد» و امکان بازگشت به برنامه اصلی

۳۴- پنج مثال از وقفه خارجی و پنج مثال از وقفه داخلی را ارائه نمایید. فرق

بین وقفه نرم‌افزاری و فراخوانی زیر روال چیست؟

حل: منابع «وقفه» خارجی، دستگاه‌های ورودی و خروجی، زمان‌سنج‌ها، مدارهای
مربوط به کنترل منبع تغذیه (power) و... را شامل می‌شوند. پس می‌توان موارد زیر را به
عنوان وقفه خارجی لیست کرد:

- وقفه دستگاه ورودی و خروجی برای درخواست ارسال داده
- وقفه دستگاه ورودی و خروجی برای اعلام پایان ارسال داده
- وقفه قطع برق از منبع تغذیه
- وقفه تایمر برای افتادن در حلقه‌های نامتناهی
- وقفه انقضای زمان یک رخداد

منابع وقفه داخلی، کاربرد غیر مجاز و اشتباه دستورات و داده‌ها هستند؛

مثال‌هایی از وقفه داخلی عبارت‌اند از:

- سرریز شدن رجیسترها مانند پشته “stack overflow”
- تقسیم بر صفر شدن در یک عبارت حسابی
- اجرای کد عملیاتی نامعتبر

- اجرای دستورالعمل با آدرس توقیف شده «آدرس مربوط به یک پردازنده دیگر در حافظه»

- سر ریز شدن پشته

تفاوت وقفه‌های نرم‌افزاری و فراخوانی زیر روال‌ها به شرح زیر است:

- ۱- آدرس وقفه‌ها به صورت سخت‌افزاری مشخص می‌گردد در صورتی که آدرس فراخوانی زیر روال‌ها به صورت دستورالعمل در حافظه مشخص می‌گردد.
- ۲- حین اجرای وقفه‌های نرم‌افزاری، مواردی مانند PC، رجیسترهای پردازنده و بیت‌های وضعیت "PSW" ذخیره می‌گردند و سپس بازیابی می‌شوند. در حالی که حین فراخوانی زیر روال، تنها PC ذخیره و سپس بازیابی می‌شود.

۳۵- کامپیوتری با انتقال PC و PSW (کلمه وضعیت برنامه) به یک وقفه پاسخ می‌دهد و سپس PSW جدیدی را از مکان IAD که توسط آدرس وقفه مشخص می‌شود می‌خواند. اولین آدرس برنامه سرویس وقفه از IAD+1 خوانده می‌شود.

الف) رشته عملیات جزئی را برای سیکل وقفه لیست کنید.

ب) رشته عملیات را برای بازگشت از زیر روال وقفه لیست کنید.

حل: کلمه وضعیت بیت‌های برنامه "PSW" و شمارنده دستورالعمل "PC" ذخیره و بازیابی می‌شوند. همچنین طبق ماهیت پشته که LIFO است، در هنگام ذخیره ابتدا PSW و سپس PC ذخیره می‌شود و هنگام بازیابی، ابتدا PC و سپس PSW بازیابی می‌شوند.

الف)

$SP \leftarrow SP - 1$

$M[SP] \leftarrow PSW$

$SP \leftarrow SP - 1$

$M[SP] \leftarrow PC$

$TR \leftarrow "Temp"$

$PSW \leftarrow M[TR]$

$TR \leftarrow TR + 1$

$$PC \leftarrow M[TR]$$

↓ ابتدای فاز برداشت دستورالعمل

(ب)

$$PC \leftarrow M[SP]$$

$$SP \leftarrow SP + 1$$

$$PSW \leftarrow M[SP]$$

$$SP \leftarrow SP + 1$$

۳۶— مثال‌هایی از کامپیوترهای دارای قالب دستورات متغیر عبارت‌اند از

IBM370 و VAX11 و Intel 386 قالب دستورات یکی از آنها را با دستورات با طول

ثابت RISCی مقایسه کنید.

حل: برای نمونه قالب دستورالعمل‌های IBM, 370 در اینجا اشاره می‌شود:

IBM 370 ، ۶ نوع قالب دستورالعمل دارد به شرح زیر:

RR	Op-code	R ₁	R ₂
----	---------	----------------	----------------

RX	Op-code	R ₁	X ₂	B ₂	D ₂
----	---------	----------------	----------------	----------------	----------------

RS	Op-code	R ₁	R ₃	B ₂	D ₂
----	---------	----------------	----------------	----------------	----------------

SI	Op-code	I ₂	B ₁	D ₁
----	---------	----------------	----------------	----------------

SS	Op-code	L	B ₁	D ₁	B ₂	D ₂
	S	Op-code	B ₂	D ₂		

بقیه موارد جستجو شده و مقایسه‌ها انجام شود.


۳۷- سه کامپیوتر از دریچه‌های ثابت زیر استفاده می‌کنند، سائز دریچه‌ها و تعداد کل ثابت در هر ثابت چقدر است؟

	کامپیوتر ۱	کامپیوتر ۲	کامپیوتر ۳
ثبات‌های عام	۱۰	۸	۱۶
ثبات‌های محلی	۱۰	۸	۱۶
ثبات‌های مشترک	۶	۸	۱۶
تعداد دریچه‌ها	۸	۴	۱۶

حل: طبق تعاریف داریم:

$$\begin{array}{l}
 \text{Computer 1 : } 10+12+10=32 \\
 \text{Computer 2 : } 8+16+8=32 \\
 \text{Computer 3 : } 16+32+16=64
 \end{array}
 \left\{ \begin{array}{l}
 \Leftarrow \downarrow \downarrow \downarrow \\
 \text{عمومی مشترک شخصی}
 \end{array} \right.
 \begin{array}{l}
 L+2C+G = \text{سائز پنجره}
 \end{array}$$

$$\Leftarrow (L+C)W+G = \text{سائز رجیستر فایل}$$


 تعداد پنجره

$$\text{Computer 1 : } (10+6)8+10=138$$

$$\text{Computer 2 : } (8+8)4+18=72$$

$$\text{Computer 3 : } (16+16)16+16=528$$

۳۸- مثالی از یک دستور کامپیوتر RISC I که اعمال زیر را انجام دهد ارائه کنید.

(الف) یک ثابت را کاهش می‌دهد.

(ب) یک ثابت را مکمل می‌کند.

(ج) علامت محتوای ثابت را معکوس می‌کند.

(د) یک ثابت را ۰ می‌کند.

هـ) یک عدد علامت‌دار را به ۴ تقسیم کند.

و) هیچ کاری انجام ندهد.

حل: با توجه به جدول ۸-۱۲ کتاب:

الف) کاهش رجیستر: SUB R22, #1, R22

ب) مکمل یک رجیستر: XOR R22, #-1, R22

ج) منهای قدرمطلق یک رجیستر: SUB #0, R22, R22

د) صفر کردن یک رجیستر: ADD #0, #0, R22

هـ) تقسیم یک عدد علامت‌دار بر ۴: SRA R22, #2, R22

و) هیچ کار: ADD R₁, #0, R₁ یا OR R₁, R₁, R₁

۳۹- دستوری از RISC I را به زبان اسمبلی بنویسید که سبب پرش به آدرس

۳۲۰۰ گردد به شرطی که بیت وضعیت $Z=1$ باشد.

الف) با استفاده از روش بلافصل

ب) با استفاده از روش نسبی ($PC = 3400$)

حل:

الف) JMP Z, #3200, #0

ب) JMPR Z, -200

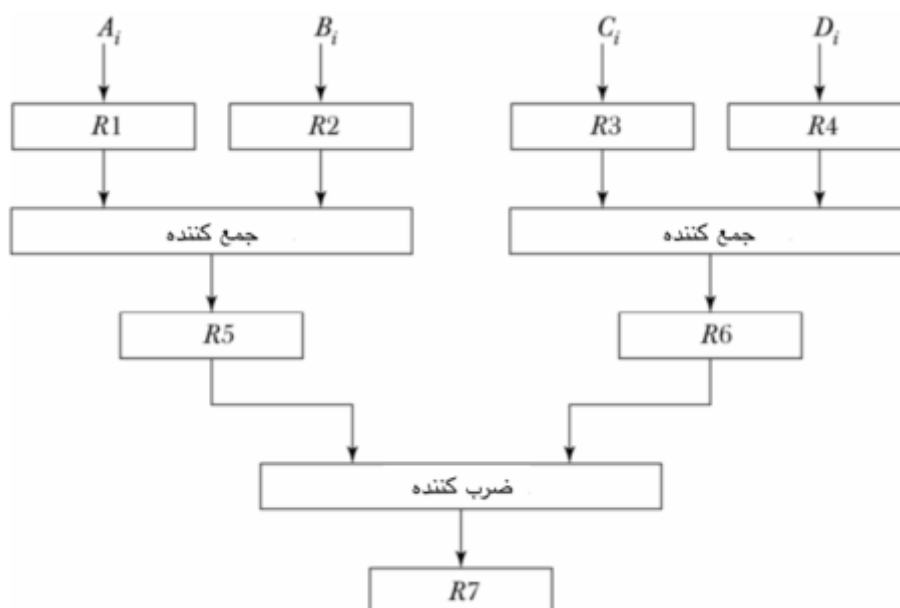
فصل نهم: پردازش خط لوله‌ای و برداری

۱- در برخی از محاسبات علمی لازم است تا رابطه حسابی $(A_i + B_i)(C_i + D_i)$

با رشته‌ای از اعداد انجام شود. یک پیکربندی خط لوله را برای انجام این کار

مشخص کنید. محتوای تمام ثبات‌های خط لوله را برای (۶ تا $i=1$) لیست نمایید.

حل:



۲- یکنمودار فاصله - زمان برای یک خط لوله شش قطعه‌ای رسم کنید که زمان پردازش هشت وظیفه را نشان دهد.

قطعه	1	2	3	4	5	6	7	8	9	10	11	12	13
1	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈					
2		T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈				
3			T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈			
4				T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈		
5					T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈	
6						T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈

پالس ساعت $(k + n - 1)t_p = 6 + 8 - 1 = 13$

۳- تعداد سیکل‌های ساعتی که برای ۲۰۰ وظیفه در یک خط لوله شش قطعه‌ای لازم است را معین کنید.
حل:

اگر k تعداد قطعه و n تعداد وظیفه باشد، پس از k سیکل ساعت اجرای دستور اول تمام می‌شود و $n-1$ دستور باقی‌مانده هر کدام در یک سیکل ساعت به پایان می‌رسند. پس داریم:

سیکل ساعت $k+n-1=62+200-1=205$ = تعداد سیکل ساعت مورد نیاز

۴- یک سیستم غیر خط لوله‌ای برای پردازش یک وظیفه 50ns زمان نیاز دارد. همان وظیفه در یک خط لوله شش قطعه‌ای به یک سیکل ساعت 10 نانوثانیه‌ای نیازمند است. نسبت افزایش سرعت خط لوله برای 100 وظیفه را معین کنید. حداکثر تسریع قابل دسترسی چقدر است؟
حل:

$$S = \frac{\text{زمان اجرای غیر خط لوله‌ای}}{\text{زمان اجرای خط لوله‌ای}} = \frac{ntn}{(k+n-1)t_p} = \frac{100 \times 50}{(6+99) \times 10} = \frac{500}{105} = 4,76$$

$$t_n = 50\text{ ns}, k = 6, t_p = 10\text{ns}, n = 100$$

$$S_{\max} = \lim_{n \rightarrow \infty} \frac{nt_n}{(k+n-1)t_p} = \frac{t_n}{t_p} = \frac{50}{10} = 5$$

۵- خط لوله شکل ۲-۹ دارای زمانهای انتشار زیر است: 40ns برای خواندن عملوند از حافظه به ثبات R_1, R_2, R_3 برای انتشار سیگنال در ضرب‌کننده، 5ns برای انتقال به R_3 و 15ns برای جمع دو عدد و قرار دادن آن در R_5 (الف) حداقل سیکل ساعتی که می‌توان به کار برد چقدر است؟

$$t_p = 45 + 5 = 50$$

حداقل t_p برابر است با حداکثر زمان انتشار قطعه‌ها

ب) یک سیستم غیر خط لوله‌ای می‌تواند همین عمل را با حذف R_3 و R_4 انجام دهد. ضرب و جمع عملوندها بدون استفاده از خط لوله چقدر طول می‌کشد؟

$$t_n = 40 + 45 + 15 = 100 \text{ ns}$$

ج) تسریع عملکرد خط لوله را برای ۱۰ وظیفه و نیز برای ۱۰۰ وظیفه محاسبه کنید.

$$K=3$$

$$S = \frac{nt_n}{(k+n-1)t_p} = \frac{10 \times 100}{(3+9)50} = 1.67 \quad \text{برای } n=10$$

$$S = \frac{100 \times 100}{(3+99)50} = 1.96 \quad \text{برای } n=100$$

د) بیشینه افزایش سرعت قابل دستیابی چقدر است؟

$$S_{\max} = \frac{t_n}{t_p} = \frac{100}{50} = 2$$

۶- می‌خواهیم یک خط لوله ضرب ممیز شناور که دو عدد صحیح دودویی هشت بیتی را ضرب می‌کند طراحی کنیم. هر قطعه از تعدادی گیت AND و یک جمع‌کننده دودویی شبیه به ضرب‌کننده آرایه در شکل ۱۰-۱۰ تشکیل شده است. الف) در هر قطعه چند گیت AND وجود دارد و جمع‌کننده با چه اندازه‌ای نیاز است؟

با مراجعه به بحث ضرب‌کننده آرایه‌ای در بخش ۳-۱۰ می‌بینیم که برای هر قطعه ۸ گیت AND و یک جمع‌کننده ۸ بیتی لازم است.

ب) چند قطعه در خط لوله است: ۷ قطعه در خط لوله وجود دارد (با توجه به ضرب‌کننده آرایه‌ای)

ج) اگر تأخیر انتشار در هر قطعه 30ns باشد، زمان متوسط لازم برای ضرب دو عدد ممیز ثابت در خط لوله چقدر است؟

$$\text{زمان متوسط} = \frac{\text{زمان اجرای خط لوله‌ای}}{\text{تعداد تکلیف‌ها}} = \frac{k+n-1}{n} t_p = \frac{(n+6)30}{n}$$

$$n = 10 : t_{avg} = 48 \text{ ns}$$

$$n = 100 : t_{avg} = 31.8 \text{ ns}$$

$$n = \infty : t_{avg} = 30 \text{ ns}$$

۷- زمان تأخیر چهار قطعه در خط لوله شکل ۹-۶ به صورت زیر است:

$$t_1 = 50 \text{ ns}, t_2 = 30 \text{ ns}, t_3 = 95 \text{ ns}, t_4 = 45 \text{ ns}. \text{ تأخیر ثبات‌های واسط } t_r = 5 \text{ ns}$$

الف) جمع ۱۰۰ جفت عدد در خط لوله چقدر طول می‌کشد؟

$$(\text{زمان لازم برای قطعه ۳}) = 95 + 5 = 100 \text{ ns} = \text{سیکل ساعت}$$

$$\Rightarrow t_p = 100 \text{ ns}$$

$$k = 4$$

زمان لازم برای جمع ۱۰۰ جفت عدد

$$= (k + n - 1)t_p = (4 + 99)100 = 10300 \text{ ns}$$

ب) چگونه می‌توان زمان کل را به نصف زمان محاسبه شده در قسمت الف) کاهش داد؟

قطعه ۳ را به دو قطعه با تأخیرهای ۵۰ ns و ۴۵ ns تقسیم می‌کنیم. در این حالت

داریم:

$$t_p = 50 + 5 = 55 \text{ ns}$$

$$k = 5 \text{ و}$$

$$\text{زمان لازم} = (k + n - 1)t_p = (5 + 99)55 = 5720 \text{ ns}$$

که تقریباً نصف زمان محاسبه شده در قسمت الف است.

۸- چگونه می‌توانید جمع‌کننده ممیز شناور خط لوله‌ای شکل ۹-۶ را برای

جمع ۱۰۰ عدد ممیز شناور $X_1 + X_2 + \dots + X_{300}$ به کار بگیرید؟

حل: خروجی جمع‌کننده را در یک مسیر بازخوردی به ورودی $B \times 2^b$ وصل

می‌کنیم و از ورودی $A \times 2^a$ برای داده‌های X_1 تا X_{100} استفاده می‌کنیم. (از شمایی

مانند خط لوله جمع‌کننده در شکل ۹-۱۲ استفاده می‌کنیم.)

۹- یک خط لوله دستورالعمل شش قطعه‌ای را برای یک کامپیوتر فرمول بندی کنید. عملیات که در هر قطعه باید انجام شود را مشخص کنید.

حل: یک طراحی می‌تواند شامل قطعه‌های زیر باشد (خط لوله، به شش قطعه تقسیم شده است):

- ۱- برداشت دستور از حافظه (FI) ۲- دیکدر دستور (DE)
 - ۳) محاسبه آدرس مؤثر (EA) ۴- برداشت عملوند از حافظه (FO)
 - ۵) اجرای دستور (EX) ۶- ذخیره نتیجه در مکان مناسب (WB)
- ۱۰- چهار روش سخت‌افزاری ممکن را که می‌تواند در خط لوله دستورالعمل برای کمینه کردن افت عملکرد ناشی از انشعاب دستور به کار رود، شرح دهید.
- حل:

۱- دریافت زودهنگام دستور مقصد: در این روش علاوه بر دستور بعد، دستور مقصد نیز دریافت و ذخیره می‌شود؛ اگر شرط انشعاب برقرار باشد خط لوله از دستور مقصد انشعاب ادامه کار می‌دهد. روش کامل‌تر این است که از هر دو جهت عمل برداشت دستور ادامه یابد تا تصمیم انشعاب گرفته شود. نه آن زمان واحد کنترل درباره رشته صحیح دستورالعمل‌ها برای ادامه برنامه تصمیم می‌گیرد.

۲- بکارگیری بافر مقصد انشعاب (BTB): BTB یک حافظه اشتراکی است که در قطعه برداشت خط لوله کار گذاشته می‌شود. هر ورودی به BTB متشکل از آدرس انشعاب اجرا شده قبلی و دستور مقصد برای آن انشعاب است. این بافر چند دستور بعد از انشعاب را نیز ذخیره می‌کند. وقتی که خط لوله دستور انشعاب را دیکد می‌کند BTB را برای آدرس دستورالعمل جست‌وجو می‌نماید. اگر آدرس در BTB باشد، دستور مستقیماً در دسترس است و برداشت پیش‌هنگام دستور برای مسیر جدید ادامه می‌یابد. اگر دستور در BTB نباشد خط لوله به رشته دستور جدیدی شیفت نموده و دستور مقصد را در BTB ذخیره می‌کند. مزیت این روش این است که دستورات انشعابی که قبلاً اتفاق افتاده‌اند بدون توقف در خط لوله موجودند.

۳- استفاده از بافر حلقه، این بافر یک فایل ثابت کوچک و سریع است که تحت نظر قطعه برداشت دستور خط لوله می‌باشد. وقتی که یک حلقه در برنامه شناسایی شود در یک بافر حلقه به طور کامل همراه با انشعابات ذخیره می‌شود. بنابراین حلقه برنامه می‌تواند مستقیماً و بدون دستیابی به حافظه اجرا شود. تا اینکه آخرین انشعاب موجب خروج از حلقه شود.

۴- پیش بینی انشعاب: یک خط لوله با پیش بینی انشعاب نتیجه حاصل از انشعاب را قبل از اجرای آن به کمک مدارهای منطقی اضافی پیش بینی می‌نماید. خط لوله سپس شروع به دریافت پیش هنگام رشته دستورات مسیر پیش بینی شده می‌نماید. یک پیش بینی صحیح زمان تلف شده به وسیله انشعاب را از بین می‌برد.

۱۱- چهار دستور رادر برنامه زیر در نظر بگیرید. فرض کنید که اولین دستورالعمل از مرحله ۱ در خط لوله شکل ۸ - ۹ آغاز شود. عملیات انجام شده در قطعه چهار، در چهار مرحله را مشخص کنید.

Load $R_1 \leftarrow M[312]$

ADD $R_2 \leftarrow R_2 + M[313]$

INC $R_3 \leftarrow R_3 + 1$

STORE $M[314] \leftarrow R_3$

حل:

مرحله ۴

FI	DA	FO	EX
	FI	DA	FO
		FI	DA
			FI

قطعه EX: انتقال کلمه حافظه به R_1

قطعه FO: خواندن $M[313]$

قطعه DA : دیکد دستور افزایش (INC)

قطعه FI : برداشت دستور STORE از حافظه

۱۲- برنامه‌ای را مثال بزنید که موجب برخورد داده در خط لوله سه قطعه‌ای

بخش ۵ - ۹ گردد.

حل:

Load: $R1 \leftarrow \text{Memory}$

Increment : $R1 \leftarrow R1+1$

۱	۲	۳	
I	A	E	
	I	A	E

R1 در E بارگذاری می‌شود و نمی‌توان همزمان آن را افزایش داد.

۱۳- مثالی ارائه نمایید که در آن در خط لوله سه قطعه‌ای بخش ۵ - ۹ از تأخیر

بار کردن استفاده کند.

حل:

یک دستور NO-OP بین دو دستور مثال سؤال ۱۲ قرار می‌دهیم:

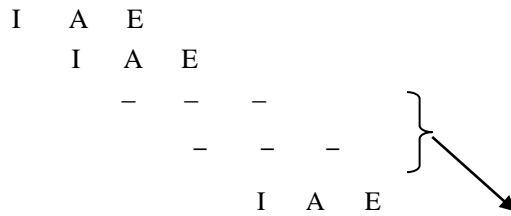
$R1 \leftarrow \text{Memory}$	I	A	E
NOP	-	-	-
$R1 \leftarrow R1+1$		I	A E

۱۴- برنامه‌ای مثال بزنید که موجب جریمه انشعاب در خط لوله سه قطعه‌ای

بخش ۵ - ۹ گردد.

حل:

- 101 Add R_2 to R_3
- 102 Branch to 104
- 103 Increment R_1
- 104 Store R_1



اتلاف زمان ناشی از انشعاب در خط لوله

۱۵- مثالی ارائه دهید که در آن از انشعاب با تأخیر در خط لوله سه قطعه‌ای

بخش ۵ - ۹ استفاده شود.

حل: از مثال مسأله ۹-۱۴ استفاده می‌کنیم.

101	Branch to 105	I	A	E
102	ADD R_2 to R_3	I	A	E
103	NO-operation	I	A	E
104	Increment R_1			
105	Store R_1	I	A	E

۱۶- ضرب دو ماتریس 40×40 را در یک پردازنده برداری در نظر بگیرید.

الف) چند جمله ضرب در هر حاصلضرب داخلی وجود دارد و چند حاصلضرب داخلی باید حساب شود.

در هر ضرب داخلی 40 ضرب نیاز است و $40^2 = 1600$ ضرب داخلی باید محاسبه شود (هر ضرب داخلی برای یک المان ماتریس حاصلضرب)

ب) چند عمل ضرب - جمع برای محاسبه ماتریس لازم است؟

40^2 ضرب داخلی داریم که هر یک 40 عمل ضرب - جمع نیاز دارند:

$$40 \times 40^2 = 40^3 = 64000$$

۱۷- در شکل ۹-۱۲ وقتی که ماتریس‌های 60×60 در هم ضرب شوند چند

سیکل ساعت برای پردازش حاصلضرب داخلی در خط لوله نیاز است؟ چند

حاصلضرب داخلی وجود دارد و چند سیکل ساعت برای محاسبه ماتریس حاصلضرب نیاز است؟

سیکل ساعت برای هر ضرب داخلی $8+60+40=72$
 تعداد ضرب‌های داخلی مورد نیاز $60^2 = 3600$
 جمع ضرب‌های جزئی از خط لوله (جمع‌کننده)
 پایان دستور اول (پر شدن خط لوله)

پس برای محاسبه ماتریس به $3600 \times 72 = 259200$ سیکل ساعت نیاز است.

۱۸- آدرس‌های مربوط به یک آرایه ۱۰۲۴ کلمه، که قرار است در حافظه شکل

۹-۱۳ ذخیره شوند را تعیین نمایید.

حل: با توجه به ساختار شکل ۹-۱۳ آدرس‌های مربوط به آرایه به صورت زیر

هستند:

0 4 8 12 1020 1 6 9 1021 2 6 10 1022

آرایه حافظه ۱ آرایه حافظه ۲ آرایه حافظه ۳

3 7 11 1023

آرایه حافظه ۴

۱۹- محاسبه‌ای برای پیش‌بینی وضع هوا به ۲۵۰ میلیارد عمل ممیز شناور نیاز

دارد. مسأله در یک ابر کامپیوتر با سرعت اجرایی ۱۰۰ مگافلاپس پردازش می‌شود.

چقدر زمان برای انجام محاسبات طول می‌کشد؟

حل:

$$\text{دقیقه } 41,67 = \text{ثانیه } 2500 = \frac{250 \times 10^9}{100 \times 10^6} = \text{زمان اجرا}$$

۲۰- کامپیوتری را با چهار پردازنده خط لوله‌ای ممیز شناور در نظر بگیرید.

فرض کنید که هر پردازنده از سیکل زمان 40 ns استفاده می‌کند. برای اجرای ۴۰۰

عمل ممیز شناور چقدر وقت نیاز است؟ اگر همین عمل با استفاده از یک پردازنده

خط لوله‌ای تکی با سیکل ساعت 10ns انجام شود آیا تفاوتی وجود خواهد

داشت؟

حل:

۴۰۰ عمل را بین چهار پردازنده تقسیم می‌کنیم. داریم:

$$\text{زمان اجرا} = \frac{400}{4} \times 40 = 4000 \text{ ns}$$

با استفاده از یک پردازنده با سیکل 10ns داریم:

$$\text{زمان اجرا} = 400 \times 10 = 4000 \text{ ns}$$

فصل دهم: معماری کامپیوتر و الگوریتم‌های

حسابی

۱- اگر در شکل ۱۰-۱ به جای $A + \bar{B} + 1$ عمل $B + \bar{A}$ (متمم ۱ عدد A بعلاوه B) را انجام دهیم، متمم ساز نیاز نیست. الگوریتمی به شکل فلوچارت برای جمع و تفریق اعداد دودویی ممیز ثابت با نمایش اندازه علامت‌دار بدست آورید که در آن اندازه‌ها با دو ریز عمل $A \leftarrow \bar{A}$ و $\bar{E} \leftarrow A + B$ تفریق شوند.

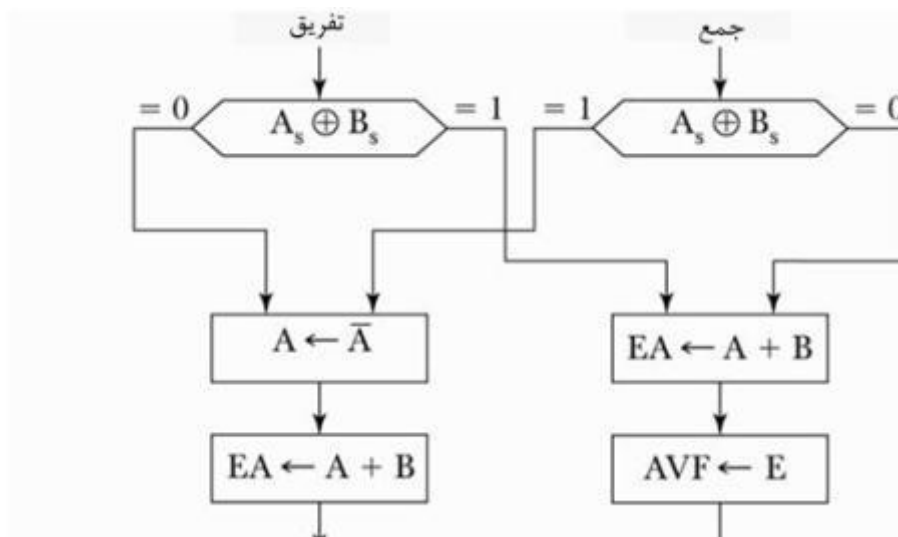
حل: با توجه به تعریف متمم در اعداد داریم: (فرم متمم ۲)

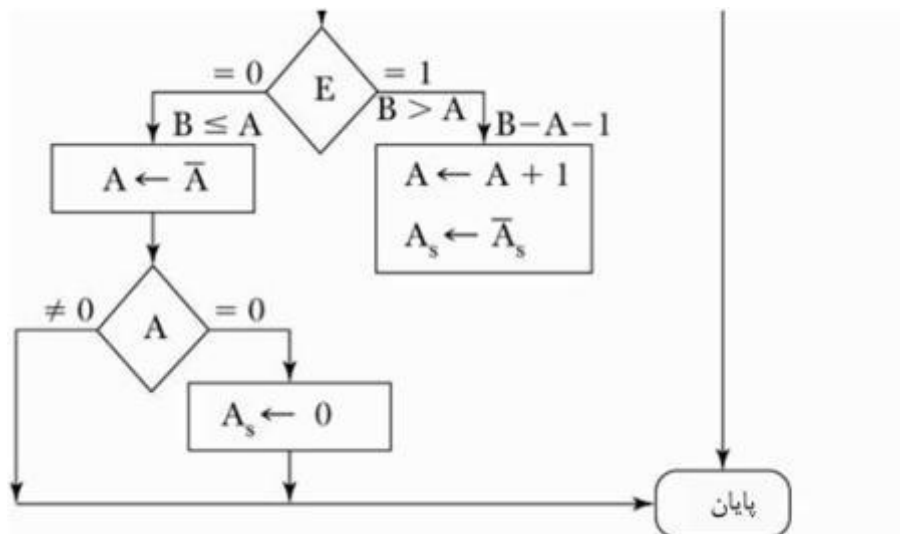
$$A - B = A + \bar{B} + 1$$

بنابراین در صورت سؤال:

$$B + \bar{A} = B - A - 1$$

بنابراین فلوچارت زیر را داریم:

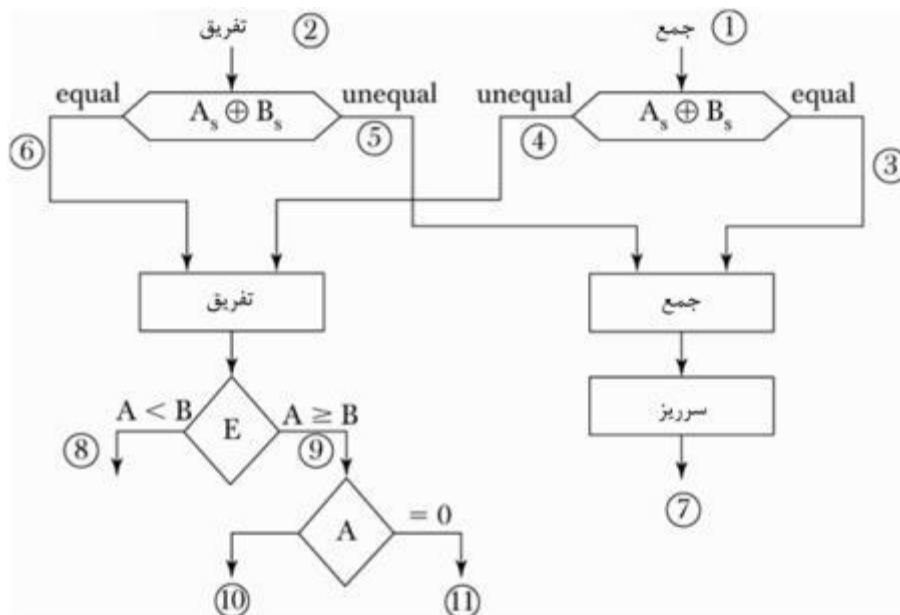




۲- هر یک از مسیرهای فلوچارت شکل ۲-۱۰ را شماره‌گذاری کنید و سپس کل مسیری که الگوریتم هنگام محاسبه اعداد با نمایش اندازه علامت‌دار زیر طی می‌کند را تعیین نمایید. در هر مورد مقدار AVF را بنویسید. بیت سمت چپ در عددهای زیر نماینده بیت علامت است.

- | | |
|--------------------------|------------------------|
| الف) $0101101 + 0011111$ | ب) $1011111 + 1101101$ |
| ج) $0101101 - 0011111$ | د) $0101101 - 0101101$ |
| ه) $1011111 - 0101101$ | |

حل: با توجه به شکل ۲-۱۰، المان‌های محاسباتی را شماره‌گذاری کرده و سپس موارد محاسباتی داده شده در سؤال را با دنباله‌ای از المان‌های محاسباتی مشخص می‌کنیم.



با توجه به ۶ بیتی بودن اعداد، در نمایش متمم (۲)، محدوده ظرفیت نتیجه از $2^6 - 1$ تا -2^6 است. پس نتیجه $-64 \leq$ ، پس نتایج خارج از این محدوده، AVF را (۱) خواهد کرد.

حال داریم:

- (الف) $(+45) + (+31) = 76$ مسیر $(1) \rightarrow (3) \rightarrow (7)$ AVF=1
- (ب) $(-31) + (-45) = -76$ مسیر $(1) \rightarrow (3) \rightarrow (7)$ AVF=1
- (ج) $(+45) - (+31) = 14$ مسیر $(2) \rightarrow (6) \rightarrow (9) \rightarrow (10)$ AVF=0
- (د) $(+45) - (+45) = 0$ مسیر $(2) \rightarrow (6) \rightarrow (9) \rightarrow (10)$ AVF=0
- (هـ) $(-31) - (+45) = -76$ مسیر $(2) \rightarrow (5) \rightarrow 7$ AVF=1

۳- عمل های حسابی زیر را با عددهای دودویی و عددهای منفی در نمایش متمم (۲) علامت دار انجام دهید. برای جای دادن هر عدد همراه با علامتش از ۷ بیت استفاده کنید. در هر حالت، با بررسی نقلی ورودی و خروجی از بیت علامت

وجود سرریز را بررسی کنید.

الف) $(+35) + (+40)$ ب) $(-35) + (-40)$

ج) $(-35) - (+40)$

حل: می‌دانیم

$$A - B = A + (-B)$$

بنابراین موارد ب و ج نتیجه یکسانی را بدست خواهند داد.

الف) E و F، دو رقم نقلی آخر

+35	0 100011			
+40	0 101000	+	E=0, F=1	$E \oplus F = 1 \Rightarrow$
+75	1001011			رخداد سرریز

ب)

-35	1011101			
-40	1011000	+	E=1, F=0	$E \oplus F = 1 \Rightarrow$
-70	0110101			رخداد سرریز

۴- عددهای دودویی را با نمایش متمم ۲ علامت دار در نظر بگیرید. هر عدد n بیت دارد: یک بیت برای علامت و $k = n - 1$ بیت برای اندازه (یا مقدار) عددی منفی مانند $-x$ به صورت $2^k + (2^k - x)$ نمایش داده می‌شود که در آن اولین 2^k نشان‌دهنده بیت علامت و $(2^k - x)$ متمم ۲ عدد x است. عدد مثبت به صورت $0 + x$ نمایش داده می‌شود، که ۰ بیت علامت و x مقدار k بیتی آن است. با این سمبل‌های کلی، ثابت کنید که مجموع $(\pm x) + (\pm y)$ را می‌توان با جمع عددهایی شامل بیت‌های علامت آنها و حذف نقلی خروجی از بیت علامت شکل داد. به بیان دیگر الگوریتم جمع دو عدد دودویی با نمایش متمم ۲ علامت‌دار را ثابت کنید.

حل: برای هر کدام از حالات $(\pm x) + (\pm y)$ سمبل‌ها را جایگزین کرده و اثبات‌ها را انجام می‌دهیم.

ردیف	عبارت	نمایش با سمبل‌های درخواستی سؤال در سیستم مکمل (۲)	نمایش نتیجه مورد انتظار در سیستم مکمل (۲)
۱	$(+x) + (+y)$	$(0+x) + (0+y)$	$0 + (x+y)$
۲	$(+x) + (-y)$	$(0+x) + 2^k + (2^k - y)$	$0 + (x - y) \text{ if } x \geq y$ $2^k + 2^k - (y - x) \text{ if } x < y$
۳	$(-x) + (+y)$	$2^k + (2^k - x) + (0+y)$	$0 + (y - x) \text{ if } y \geq x$ $2^k + 2^k - (x - y) \text{ if } y < x$
۴	$(-x) + (-y)$	$(2^k + 2^k - x) + (2^k + 2^k - y)$	$2^k + 2^k - (x + y)$

حال برای تک‌تک ردیف‌ها داریم:

۱-

$$(0+x) + (0+y) = 0 + (x+y)$$

۲-

$$2^k + 2^k = 2^{k+1} \rightarrow \text{رقم نقلی} \rightarrow (x-y) \text{ صرف نظر از رقم نقلی}$$

۳- مثل مورد ۲

$$2^k + 2^k = 2^{k+1} \rightarrow \text{رقم نقلی} \rightarrow (y-x) \text{ صرف نظر از رقم نقلی}$$

۴-

$$2^{k+1} + 2^{k+1} = 2^{k+2} \rightarrow \text{رقم نقلی} \rightarrow -(x+y) \text{ صرف نظر از رقم نقلی}$$

۵- یک رویه سخت‌افزاری برای آشکارسازی سرریز با مقایسه علامت مجموع

و علامت‌های مضاف و مضاف علیه بسازید. عددها در نمایش متمم (۲) علامت‌دار

هستند.

حل: در عبارت روبرو، در رابطه با تعریف رخداد سرریز داریم:

$A+B$

سرریز زمانی رخ می دهد که:

(۱) هر دو عدد مثبت، علامت مجموع منفی

(۲) هر دو عدد منفی، علامت مجموع مثبت

پس جدول زیر را داریم:

A_S علامت A

B_S علامت B

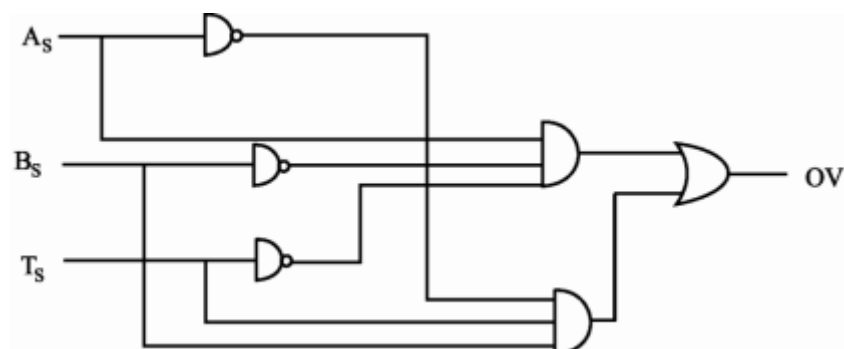
T_S علامت مجموع

T_S	B_S	A_S	OV
۰	۰	۰	۰
۰	۰	۱	۱
۰	۱	۰	۰
۰	۱	۱	۰
۱	۰	۰	۰
۱	۰	۱	۰
۱	۱	۰	۱
۱	۱	۱	۰

پس:

$$OV = T'_S B'_S A_S + T_S B_S A'_S$$

پس مدار منطقی زیر را خواهیم داشت:



۶- الف) $-15 = (-6) + (-9)$ را با عددهای دودویی با نمایش متمم ۱ علامت‌دار فقط پنج بیتی برای هر عدد (با بیت علامت) انجام دهید. نشان دهید که رویه آشکارسازی سرریز با بررسی دو نقلی آخر در این مورد به نتیجه نمی‌رسد.
 ب) رویه اصلاح شده‌ای را برای آشکارسازی سرریز وقتی که اعداد متمم ۱ استفاده می‌شوند پیشنهاد کنید:

حل: الف) برای مثال نقض، مثال زیر را در نظر بگیرید:

$$-9 = (10110)_2$$

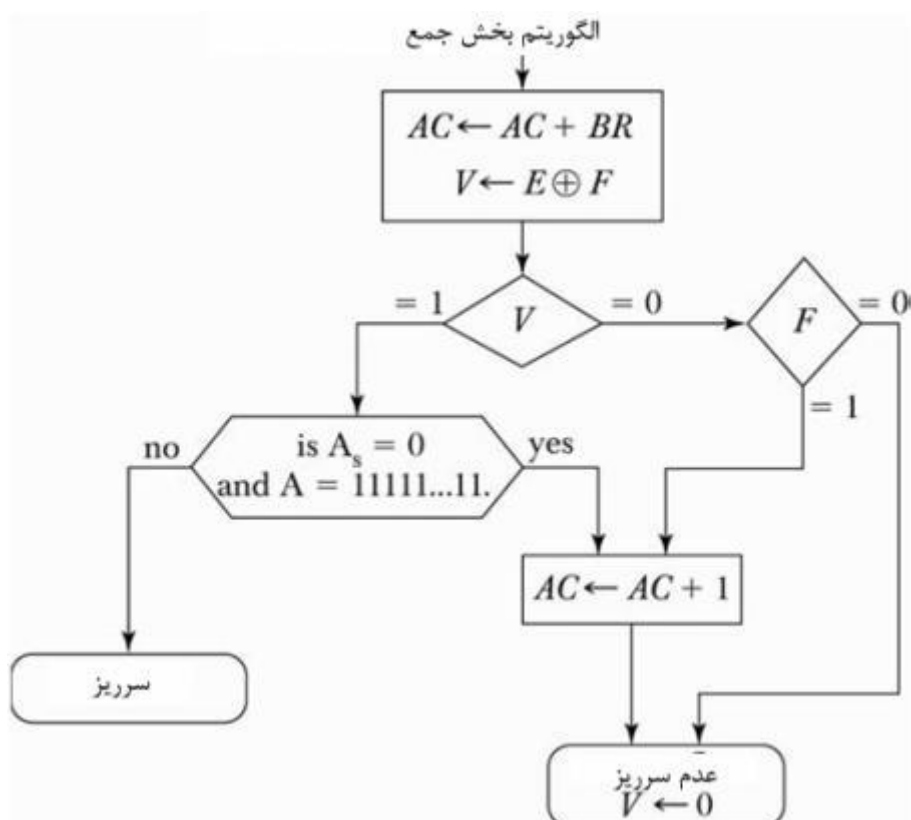
$$-6 = (11001)_2$$

$$\begin{array}{r} -9 \quad 10110 \\ + -6 \quad 11001 \\ \hline -15 \quad 01111 \end{array}$$

در این عمل، دو رقم نقلی آخر متفاوت بوده پس OR انحصاری آنها ۱ برمی‌گرداند. در حالی که قاعدتاً ما نباید سرریز را داشته باشیم زیرا نتیجه منفی است.

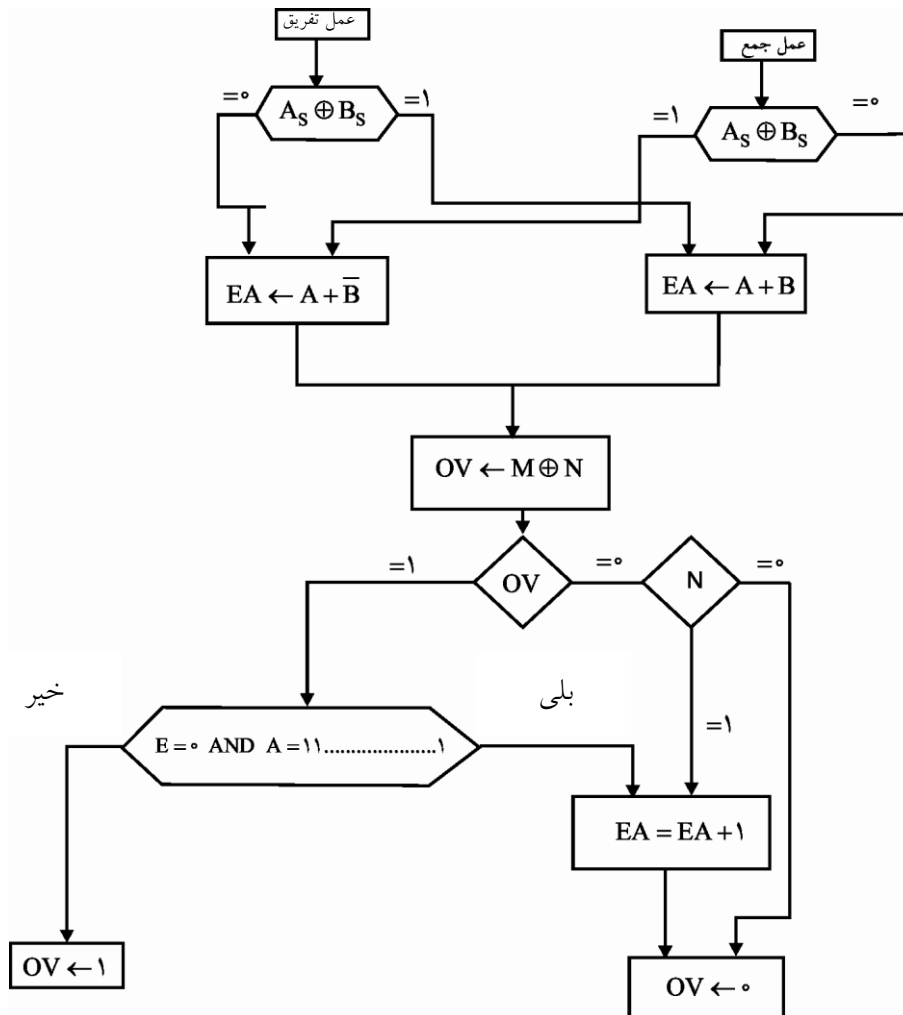
ب) با توجه به مثال نقض بالا، ما نتیجه را برای "1.....0111" شدن چک می‌کنیم.

پس:



۷- الگوریتمی را به شکل فلوچارت برای جمع و تفریق دو عدد دودویی ممیز ثابت وقتی که عددهای منفی به صورت متمم ۱ هستند بدست آورید.

حل: با توجه به حل سؤال قبل و ترکیب آن با الگوریتم جمع و تفریق اعداد منفی متمم ۲ در صفحه ۳۳۶ کتاب، فلوچارت زیر را خواهیم داشت: M و N دو رقم نقلی آخر نتیجه هستند.



سرریز داریم

سرریز نداریم

۸- ثابت کنید ضرب دو عدد n رقمی در پایه r ، حاصلضربی بزرگتر از $2n$ رقم تولید نمی‌کند. نشان دهید که این به معنی آن است که سرریزی در عمل ضرب رخ نمی‌دهد.

حل: می‌دانیم بیشترین مقدار قابل نمایش برای یک عدد n رقمی $r^n - 1$ و برای یک عدد $2n$ رقمی، $r^{2n} - 1$ است. بنابراین ماکزیمم حاصل ضرب دو عدد n رقمی در یکدیگر برابر است با:

$$(r^n - 1) \times (r^n - 1) = r^{2n} - 2r^n + 1 < r^{2n} - 1$$

$$\Rightarrow \dots \Rightarrow r^n > 1$$

با توجه به اینکه مبنا از ۲ شروع می‌شود یعنی $r \geq 2$ و به ازای $n \geq 1$ یعنی اعداد ۱ رقمی به بالا این شرط همیشه سازگار است، سرریز رخ نمی‌دهد.

۹- محتوای ثبات‌های E و A و Q و SC (مطابق جدول ۱۰-۲ کتاب) را در حین روند ضرب دو عدد دودویی ۱۱۱۱ (مضروب) و ۱۰۱۰۱ (مضروب فیه) نشان دهید. علامت‌ها در نظر گرفته نشده‌اند.

حل: همانند روال موجود در جدول ۱۰-۲ داریم:

	E	A	Q	S_C
مضروب فیه در Q	۰	۰۰۰۰۰	۱۰۱۰۱	۱۰۱
		۱۱۱۱۱		
$Q_n = 1$ ، جمع B	۰	۱۱۱۱۱		
شیفت EAQ به راست		۰۱۱۱۱	۱۱۰۱۰	۱۰۰
$Q_n = 0$ و شیفت EAQ به راست		۰۰۱۱۱	۱۱۱۰۱	۰۱۱
$Q_n = 1$ و جمع B		۱۱۱۱۱		
	۱	۰۰۱۱۰		
شیفت EAQ به راست	۰	۱۰۰۱۱	۰۱۱۱۰	۰۱۰
$Q_n = 0$ و شیفت EAQ به راست		۰۱۰۰۱	۱۰۱۱۱	۰۰۱
$Q_n = 1$ و جمع B		۱۱۱۱۱		
	۱	۰۱۰۰۰		
شیفت EAQ به راست		AQ = ۱۰۱۰۰۱۰۱۱ = (651) ₁₀		۰۰۰

۱۰- محتوای ثبات‌های E و A و Q و SC (مطابق شکل ۱۰-۱۲) را در حین روند تقسیم (الف) ۱۰۱۰۰۰۱۱ بر ۱۰۱۱ و (ب) ۰۰۰۰۱۱۱۱ بر ۰۰۱۱ نشان دهید.

(ازمقسوم ۸ بیتی استفاده کنید.)

حل: همانند روال شکل (۱۰-۲)، نتایج به صورت زیر خواهد بود:

الف) $B=1011$, $\bar{B}+1=0101$, $DVF=0$

E	A	Q	SC
۰	۱۰۱۰	۰۰۱۱	۱۰۰
۱	۰۱۰۰	۰۱۱۰	
	<u>۰۱۰۱</u>		
۱	۱۰۰۱	۰۱۱۱	۰۱۱
۱	۰۰۱۰	۱۱۱۰	
	<u>۰۱۰۱</u>		
۱	۰۱۱۱	۱۱۱۱	۰۱۰
۰	۱۱۱۱	۱۱۱۰	
	<u>۰۱۰۱</u>		
۱	۰۱۰۰	۱۱۱۱	۰۰۱
۰	۱۰۰۱	۱۱۱۰	
	<u>۰۱۰۱</u>		
۰	۱۱۱۰	۱۱۱۰	
	<u>۱۰۱۱</u>		
۱	۱۰۰۱	۱۱۱۰	

خارج قسمت باقی مانده

ب) $B=0011$ $\bar{B}+1=1101$

E	A	Q	SC
	۰۰۰۰	۱۱۱۱	۱۰۰
۰	۰۰۰۱	۱۱۱۰	
	<u>۱۱۰۱</u>		
۰	۱۱۱۰	۱۱۱۰	
	<u>۰۰۱۱</u>		
۱	۰۰۰۱		۰۱۱
۰	۰۰۱۱	۱۱۰۰	
	<u>۱۱۰۱</u>		
۱	۰۰۰۰	۱۱۰۱	۰۱۰
۰	۰۰۰۱	۱۰۱۰	
	<u>۱۱۰۱</u>		
۰	۱۱۱۰	۱۰۱۰	
	<u>۰۰۱۱</u>		
۱	۰۰۰۱		۰۰۱
۰	۰۰۱۱	۰۱۰۰	
	<u>۱۱۰۱</u>		
۱	۰۰۰۰	۰۱۰۱	۰۰۰
	باقی مانده	خارج قسمت	

۱۱- نشان دهید که جمع B پس از عمل $A + \bar{B} + 1$ مقدار اولیه A را می دهد. با

رقم نقلی پایانی چه باید کرد؟

حل: طبق تعریف متمم (۲)

$$A + \bar{B} + 1 \Rightarrow A + Z^n - B = Z^n + A - B$$

حال با B جمع می‌کنیم:

$$(2^K + A - B) + B = 2^n + A$$

از رقم نقلی صرف نظر می‌کنیم $A \Leftarrow$

۱۲- چرا علامت باقیمانده پس از تقسیم باید با علامت مقسوم یکسان باشد؟

حل: چهار حالت تقسیم با علامت‌های مختلف را برای مثال زیر در نظر می‌گیریم:

$$\frac{A}{B} = Q + \frac{R}{B}$$

$$\frac{+52}{+5} = +10 + \frac{+2}{+5} = +10/4 \quad (۱)$$

$$\frac{+52}{-5} = -10 + \frac{+2}{-5} = -10/4 \quad (۲)$$

$$\frac{-52}{+5} = -10 + \frac{-2}{+5} = -10/4 \quad (۳)$$

$$\frac{-52}{-5} = +10 + \frac{-2}{-5} = +10/4 \quad (۴)$$

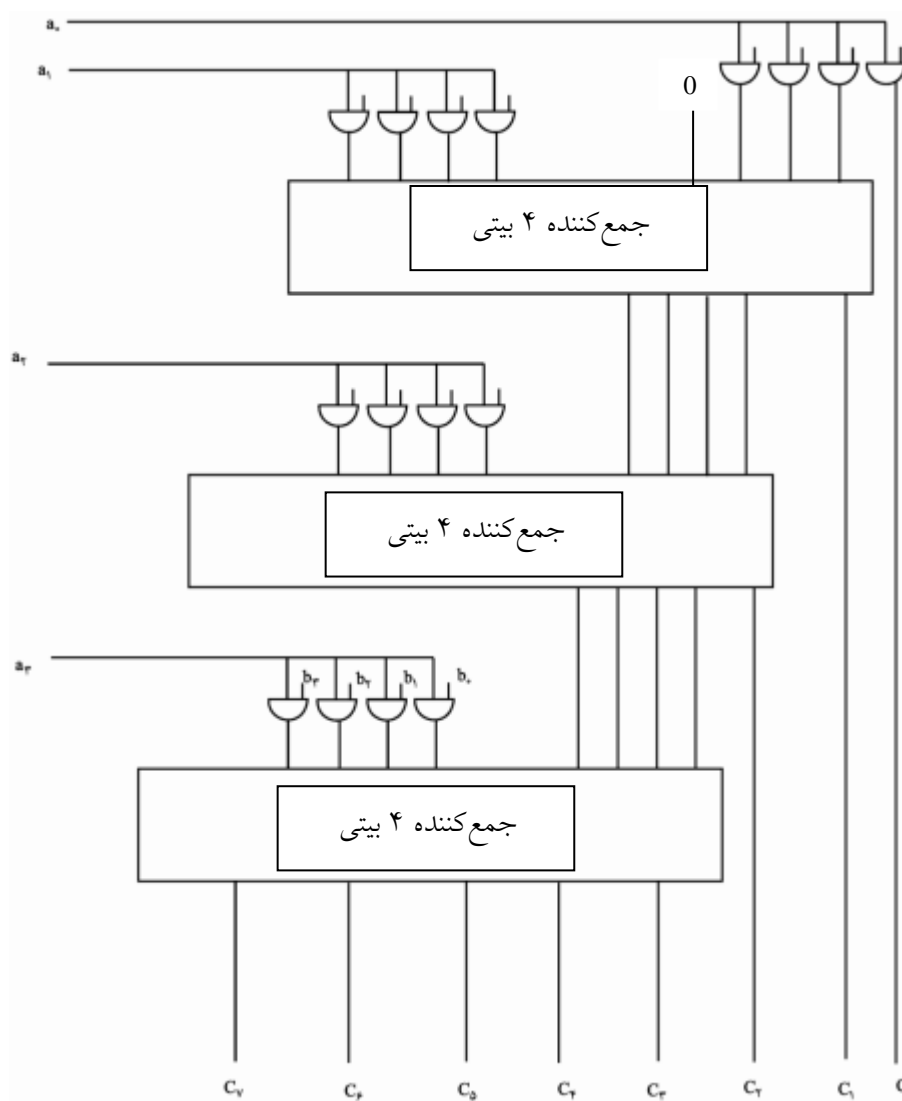
با توجه به حالت‌های مختلفی می‌بینیم علامت باقی مانده (۲) مانند علامت مقسوم (۵۲) است.

۱۳- یک ضرب کننده آرایه‌ای طراحی کنید که دو عدد چهار بیتی را در هم ضرب کند. از گیت AND برای جمع کننده دودویی استفاده کنید.

حل: با توجه به شکل (۱۰-۱۰) مدار زیر را خواهیم داشت (ورودی دوم گیت‌های AND، بیت‌های b_0, b_1, b_2, b_3 هستند):

$$a_3 a_2 a_1 a_0 \quad * \quad b_3 b_2 b_1 b_0$$

$$C_7 C_6 C_5 C_4 C_3 C_2 C_1 C_0 = \text{نتیجه}$$



با توجه به شکل می بینیم که در ضرب ۴ بیت در ۴ بیت، ۱۶ گیت AND و ۳ جمع کننده ۴ بیتی نیاز است. در حالت کلی اگر ضرب شونده k بیتی و ضرب کننده J بیتی باشد، برای ضرب آرایه ای به $J \times K$ تا گیت AND و $J-1$ عدد جمع کننده K بیتی نیاز است.

۱۴- روند قدم به قدم ضرب با الگوریتم بوت را (مثل جدول ۳-۱۰) هنگامی که اعداد دودویی زیر در هم ضرب می‌شوند نشان دهید. فرض کنید ثبات‌های پنج بیتی اعداد علامت‌دار را نگه می‌دارند. مضروب در هر دو حالت ۱۵+ است.

$$\text{الف) } (+13) \times (+15) \quad \text{ب) } (-13) \times (+15)$$

حل: با توجه به تعریف الگوریتم ضرب Booth داریم:

۱- ضرب شونده از حاصلضرب جزئی تفریق می‌شود اگر به اولین یک (از سمت راست) در ضرب‌کننده برسیم.

۲- ضرب شونده به حاصلضرب جزئی افزوده می‌شود اگر به اولین صفر (در یک رشته صفر) در ضرب‌کننده برسیم.

۳- حاصلضرب جزئی تغییر نمی‌کند وقتی بیت ضرب‌کننده با بیت قبلی مشابه باشد. روند را برای مورد (ب) داریم:

$$\text{جواب آخر } (1100111101)_2 = -195 = (-13) \times (+15)$$

$$BR = 011111(+15) \quad \overline{BR} + 1 = 10001(-15), Q_R = 10011(-13)$$

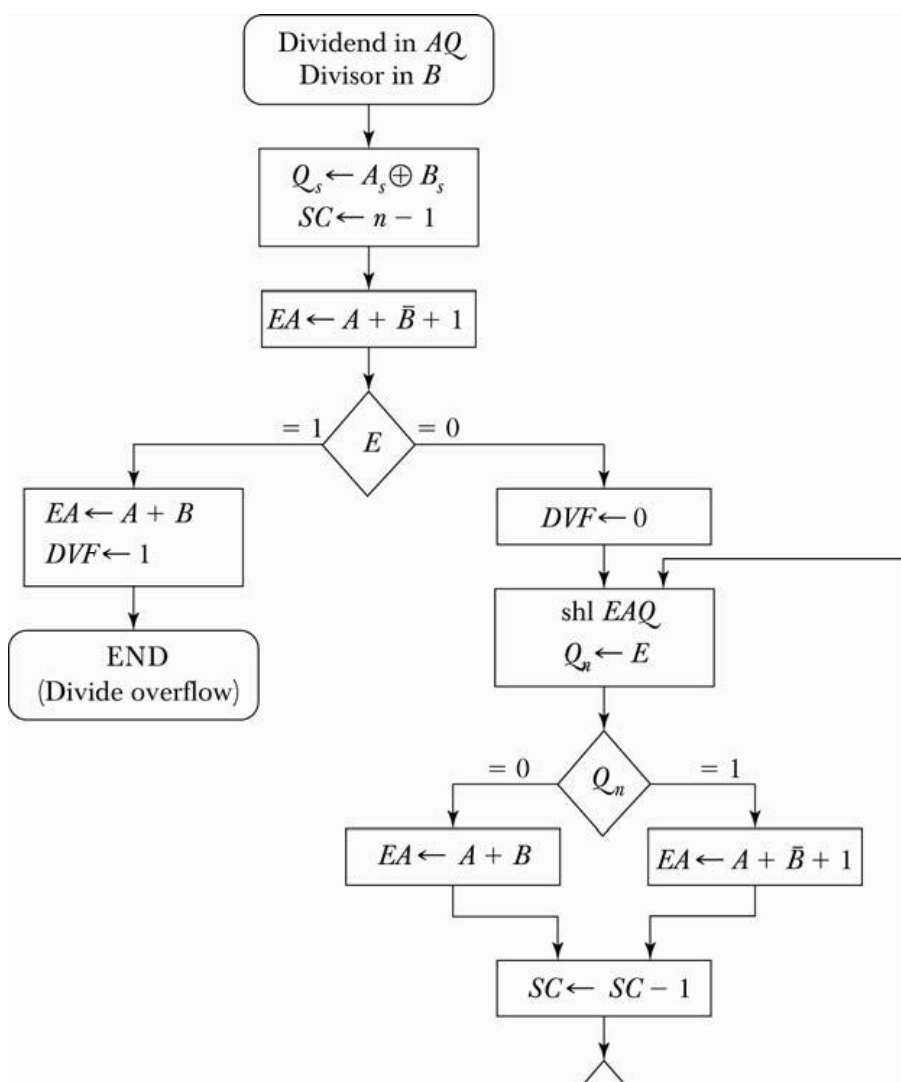
Q_n	Q_{n+1}		AC	QR	Q_{n+1}	SC
			۰۰۰۰۰	۱۰۰۱۱	۰	۱۰۱
۱	۰	تفریق	<u>۱۰۰۰۱</u>			
			۱۰۰۰۱			
		شیفت راست	۱۱۰۰۰	۱۱۰۰۱	۱	۱۰۰
۱	۱	شیفت راست	۱۱۱۰۰	۰۱۱۰۰	۱	۰۱۱
۰	۱	جمع	<u>۰۱۱۱۱</u>			
		شیفت راست	۰۰۱۰۱	۱۰۱۱۰	۰	۰۱۰
۰	۰	شیفت راست	۰۰۰۱۰	۱۱۰۱۱	۰	۰۰۱

۱	۰	تفریق	۱۰۰۰۱			
			۱۰۰۱۱			
		شیفت راست	۱۱۰۰۱	۱۱۱۰۱	۱	۰۰۰
			-۱۹۵			

۱۵- الگوریتمی را به شکل فلوچارت برای روش غیر بازیافتی تقسیم دودویی ممیز ثابت بدست آورید.

حل: دقت کنید، در روش nonrestoring، اگر حاصل تفریق منفی شود، B به آن اضافه نمی‌شود، حاصل منفی به سمت چپ شیفت می‌یابد و پس B به آن اضافه می‌شود.

مثلاً در حالت $A < B$ ، در حالت restoring ابتدا A-B شده، سپس با B جمع می‌شود. یعنی $A-B+B$ که A بازیابی می‌شود، پس $2A-B$ محاسبه می‌شود، اما در حالت A-B nonrestoring را با B جمع نمی‌کنیم. و مستقیم $2A-B$ محاسبه می‌شود. پس:



۱۶- الگوریتمی برای محاسبه ریشه دوم عدد دودویی ممیز ثابت به دست آورید.

حل: اگر روبرو را داشته باشیم:

$$\sqrt{A} = B$$

اگر A, K ، بیتی باشد، برای $B \frac{K}{2}$ (عدد صحیح شده) بیت در نظر خواهیم گرفت.

از طرفی در رابطه با قضیه جذر داریم: $\sqrt{A} \leq \frac{A}{2}$

در حالت کلی داریم:

$$B^2 + R = A$$

الگوریتم پیشنهادی به شرح زیر است:

از عدد $\frac{A}{2}$ (عدد صحیح شده) شروع کرده و تعداد عدد را در A پیدا می‌کنیم. هر گاه تعداد عدد در A با خود عدد برابر شد، عدد همان جذر A خواهد بود. در غیر این صورت یکی از عدد کم کرده و دوباره تست می‌کنیم.

مثال: محاسبه ریشه دوم عدد ۲۸ : $\frac{28}{2} = 14$

از ۱۴ شروع می‌کنیم:

عدد	تعداد در ۲۸	نتیجه
14	2	$14 \neq 2$
13	2	$13 \neq 2$
12	2	$12 \neq 2$
11	2	$11 \neq 2$
10	2	$10 \neq 2$
9	3	$9 \neq 3$
8	3	$8 \neq 3$
7	4	$7 \neq 4$
6	4	$6 \neq 4$
5	5	$5 = 5 \Rightarrow 5^2 + 3 = 28$

۱۷- یک عدد دودویی ممیز شناور هفت بیت برای نمای بایاس شده‌اش دارد.

مقدار ثابت بایاس ۶۴ است.

الف) لیست بایاس شده تمام نماها را از ۶۴- تا ۶۳+ بدست آورید.

ب) نشان دهید که یک مقایسه‌گر مقدار هفت بیتی می‌تواند برای مقایسه اندازه نسبی دو نما بکار رود.

ج) نشان دهید که پس از جمع دو نمای بایاس شده لازم است ۶۴ را کم کنیم تا مجموع نماها به شکل بایاس شده بدست آید. چگونه می‌توان ۶۴ را با جمع کردن مقدار متمم ۲ آن تفریق کرد؟

د) نشان دهید که پس از تفریق دو نمای بایاس شده لازم است ۶۴ را اضافه کنیم تا تفاضل بایاس شده نماها را داشته باشیم.

حل:

(الف)

$$e \quad -64 \quad -63 \quad \dots \quad -1 \quad 0 \quad 1 \quad \dots \quad 62 \quad 63$$

$$e+64 \quad \phi \quad 1 \quad \dots \quad 63 \quad 64 \quad 65 \quad \dots \quad 126 \quad 127$$

ب) بدیهی است تعداد بیت برای نمایش محدوده (63 - 64) برابر با تعداد بیت برای نمایش محدوده (0-127) است.

$$2^7 = 128$$

ج) مشخص است باید ۶۴ کم شود.

$$(e_1 + 64) + (e_2 + 64) = (e_1 + e_2 + 64) + 64$$

د) مشخص است باید ۶۴ اضافه شود.

$$(e_1 + 64) - (e_2 - 64) = e_1 + e_2$$

۱۸- الگوریتمی به شکل فلوچارت برای مقایسه دو عدد دودویی علامت‌دار

وقتی که عددهای منفی به شکل متمم ۲ علامت‌دار باشند به دست آورید.

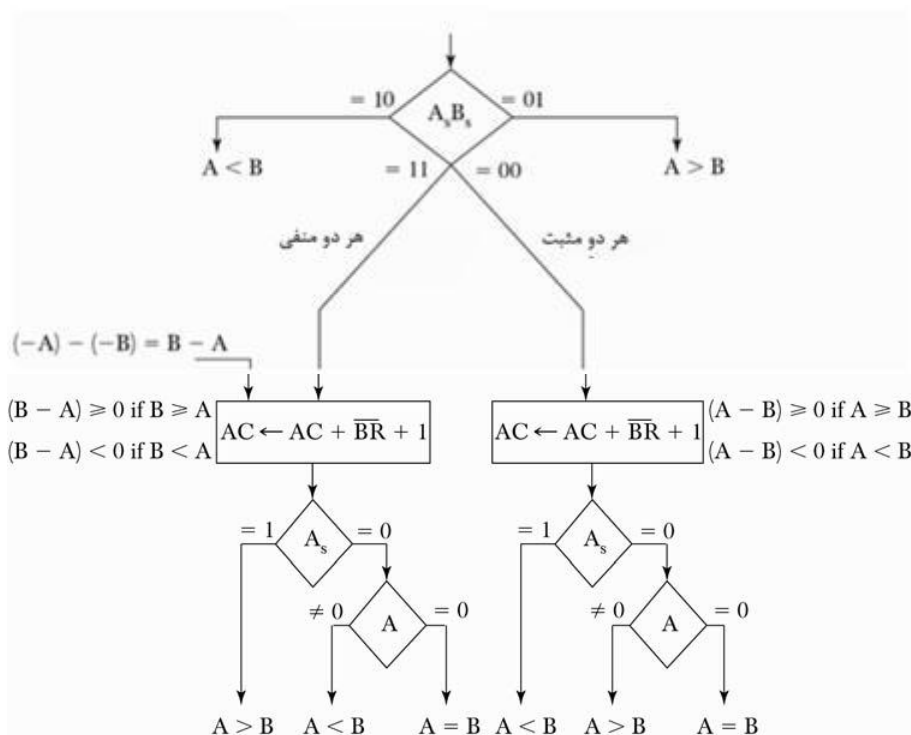
الف) با استفاده از عمل تفریق با عددهای متمم ۲ علامت‌دار

ب) با مرور و مقایسه جفت بیت‌ها از چپ به راست

حل: دقت شود هر گاه علامت دو عدد متفاوت بود، عدد که بیت علامت آن صفر

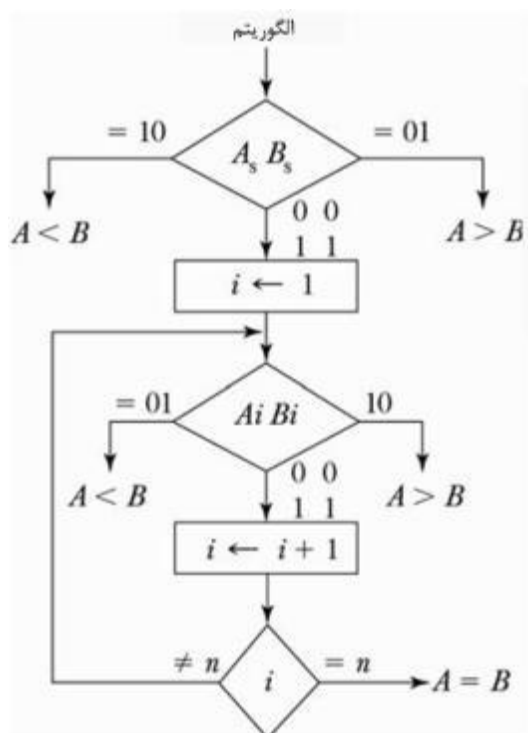
است، بزرگتر است.

الف) در غیر این صورت طبق روال عمل خواهیم کرد.



ب) از سمت بیت‌های با ارزش‌تر تساوی بودن را چک می‌کنیم و به سمت بیت‌های

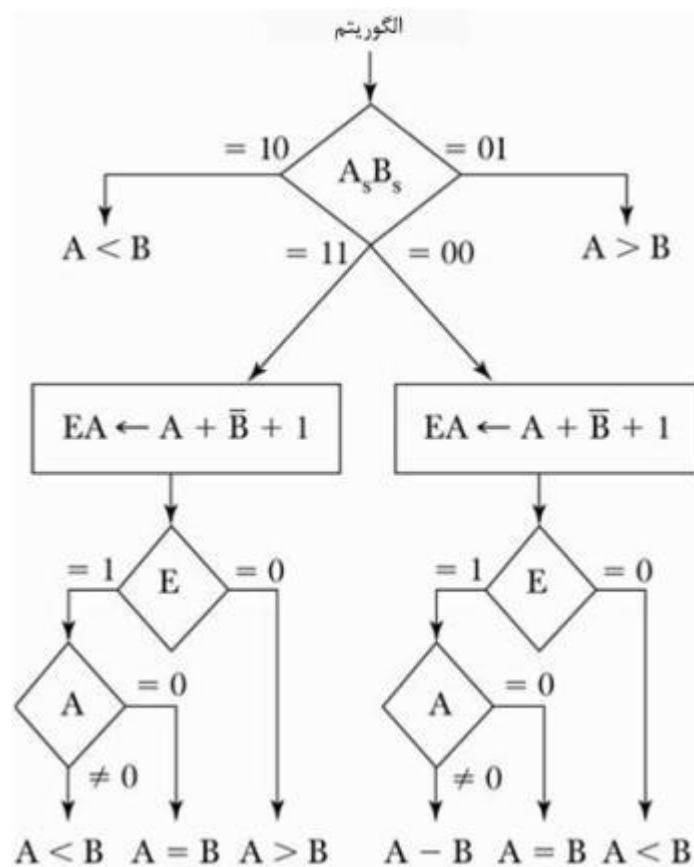
کم ارزش‌تر حرکت می‌کنیم.



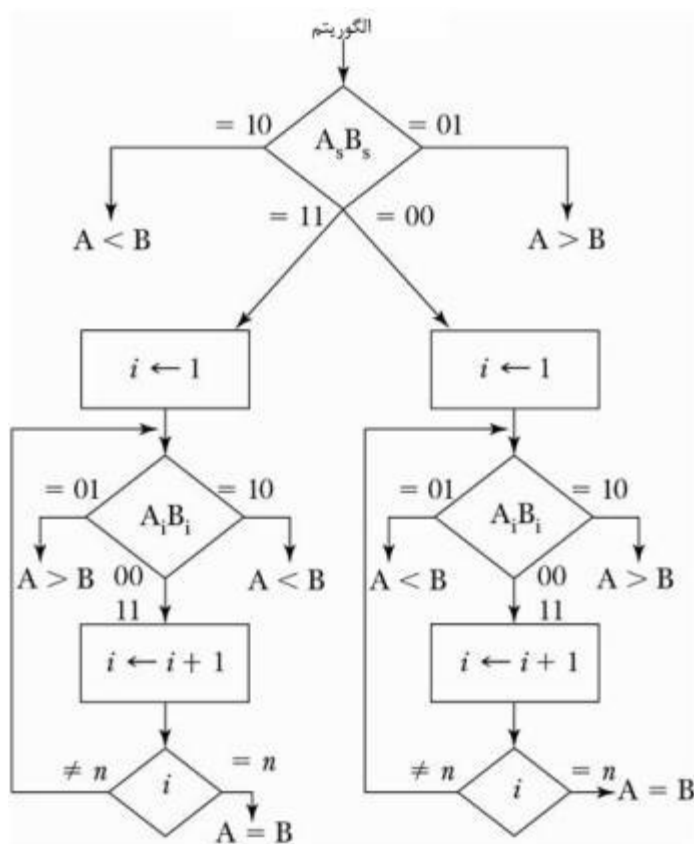
۱۹- مسئله ۱۸ را برای اعداد دودویی مقدار علامت‌دار تکرار کنید.

حل:

(الف)



ب) در رابطه با سیستم مقدار علامت‌دار داریم:
 برای اعداد مثبت آنکه بیت با ارزش‌تر (۱) باشد بزرگ‌تر است.
 برای اعداد منفی آنکه بیت با ارزش‌تر (۱) باشد کوچک‌تر است.



۲۰- فرض کنید n ، تعداد بیت‌های مانتیس در یک عدد دودویی ممیز شناور است. هنگامی که مانتیس‌ها در جمع و تفریق هم ردیف می‌شوند، اختلاف نماها ممکن است بزرگ‌تر از $n-1$ باشد. اگر این حالت رخ دهد، مانتیس با نمای کوچکتر کلاً از ثبات شیف داده می‌شود. هم ردیف کردن مانتیس را در شکل (۱۵-۱۰) با اضافه کردن یک توالی شمارشی که تعداد شیف‌ها را می‌شمارد اصلاح کنید. اگر تعداد شیف‌ها بزرگ‌تر از $n-1$ باشد، بزرگ‌ترین عدد برای تعیین نتیجه بکار خواهد رفت.

حل: این مطلب را با یک مثال نمایش می‌دهیم. فرض شود دو عدد زیر برای عمل

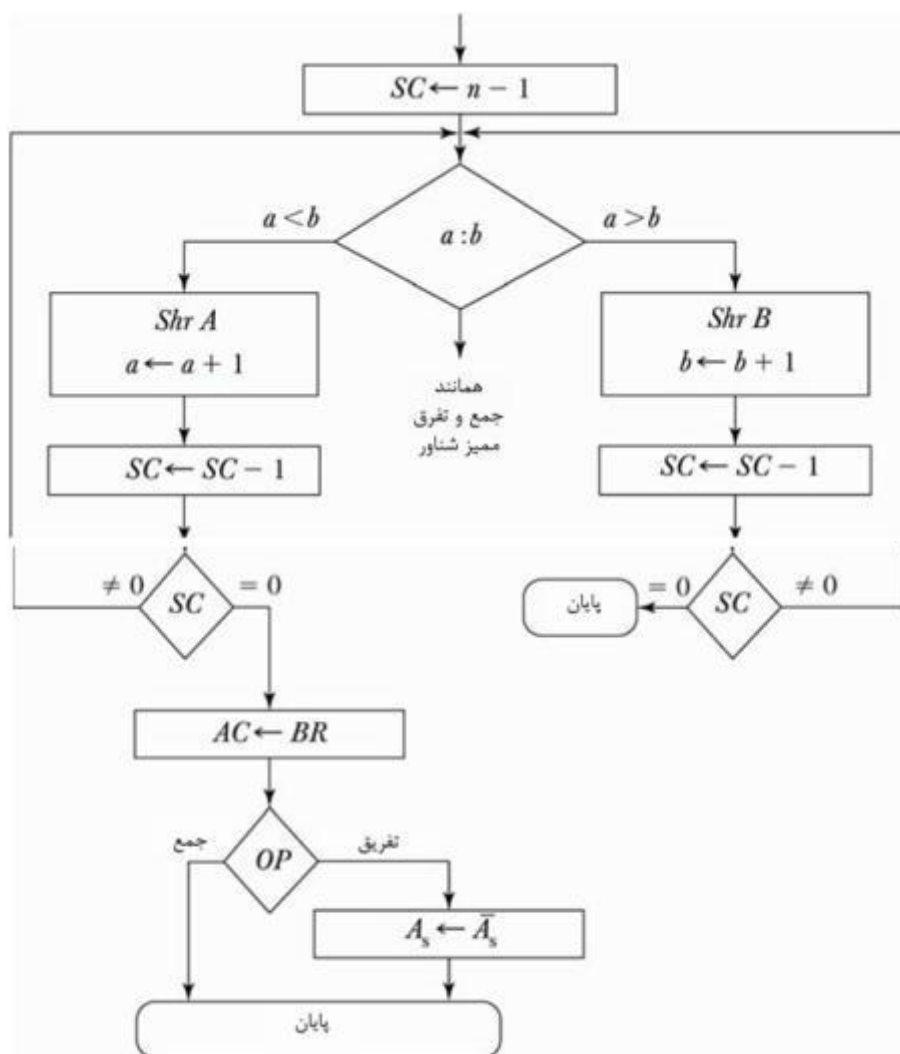
جمع آماده می‌شوند.

$$A = 0,11011 \times 2^5$$

$$B = 0,10111 \times 2^{10}$$

در $A+B$ ، می‌بایست به نمای A ، ۵ واحد اضافه شود و مانتیس A ، ۵ بار شیفت به راست انجام شود که عملاً مانتیس از بین خواهد رفت.

بنابراین با استفاده از شمارنده SC و اضافه کردن حلقه‌های بررسی شیفت مانتیس، در الگوریتم شکل ۱۵-۱۰ کتاب، درخواست سؤال را برآورده می‌کنیم.



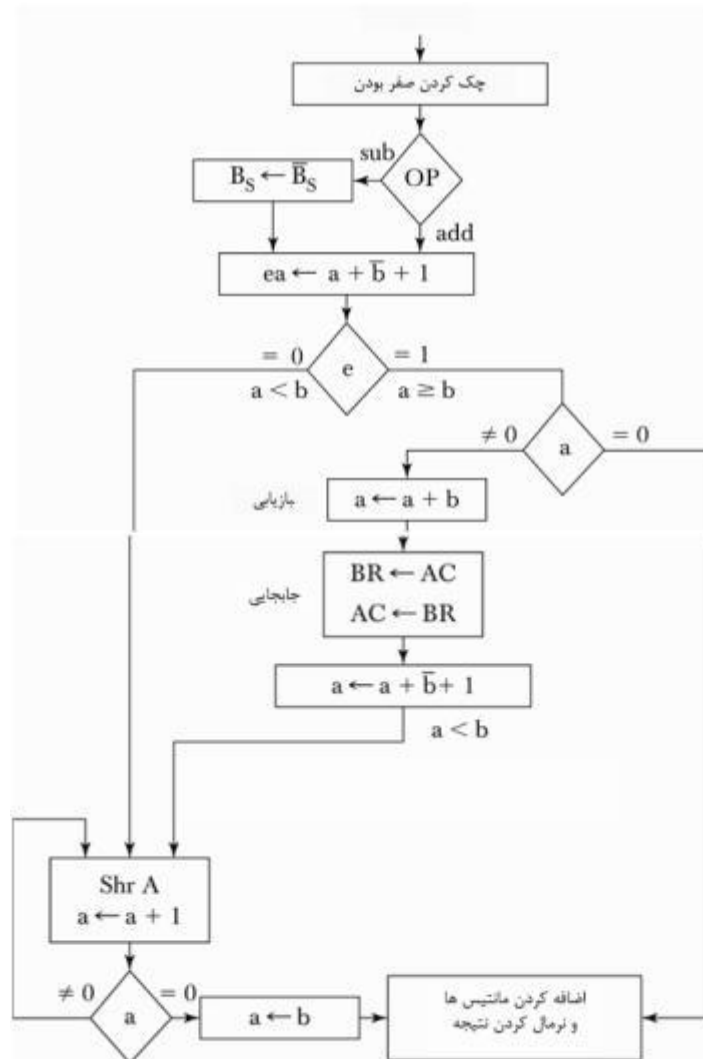
۲۱- رویه هم ردیف کردن ماتریس‌ها در جمع و یا تفریق اعداد ممیز شناور

می‌تواند به صورت زیر بیان شود:

نمای کوچک‌تر را از بزرگ‌تر تفریق کنید و ماتریسی را که دارای نمای کوچک‌تر است به تعداد تفاضل نماها به راست شیفت دهید. نمای مجموع (یا تفاضل) برابر با نمای بزرگ‌تر است. بدون استفاده از مقایسه‌کننده مقادارها، با فرض

نماهای بایاس شده، و در نظر گرفتن اینکه فقط AC را می‌توان جابجا کرد، الگوریتمی را به شکل فلوچارت برای هم ردیف کردن مانتیس‌ها و قرار دادن نمای بزرگ‌تر در AC بدست آورید.

حل:



۲۲- نشان دهید که سرریز مانتیس بعد از عمل ضرب ممکن نیست.

حل: از دو جنبه می توان این نتیجه را بررسی کرد.

۱- مانتیس های نرمال شده، عدد اعشای کوچکتر از ۱ هستند. مثلاً $0/1001$ یا $0/1101$ ؛ از طرفی می دانیم ضرب اعداد کوچکتر از ۱ همیشه کوچکتر از ۱ خواهد بود بنابراین سرریز پیش نخواهد آمد و تنها زیر ریز پیش خواهد آمد.

۲- زمانی که دو عدد n بیتی در هم ضرب شوند، نتیجه حداکثر $2n$ بیتی خواهد بود. این قضیه را در سؤال ۸ همین فصل اثبات کرده ایم.

۲۳- نشان دهید در تقسیم دو عدد نرمال شده ممیز شناور با مانتیس کسری همواره خارج قسمت نرمال شده خواهد داد؛ به شرط اینکه قبل از عمل تقسیم، هم ردیف کردن مقسوم انجام شود.

حل: دقت شود هر دو مانتیس ها نرمالیزه هستند.

$$\frac{A}{B} = \frac{0/1\ldots\ldots}{0/1\ldots\ldots}$$

دو شرط زیر را بررسی می کنیم:

1) $A \geq B \Rightarrow$ جواب به صورت روبرو خواهد بود. $\Rightarrow 1/0\ldots\ldots \Rightarrow$ شیفت به راست

2) $A < B \Rightarrow$ جواب به صورت روبرو خواهد بود. $\Rightarrow 0/1\ldots\ldots$

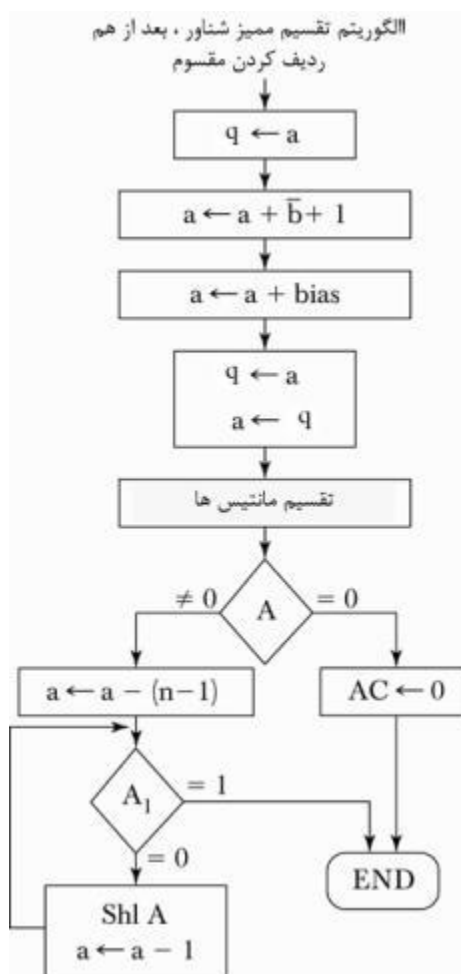
۲۴- فلوچارت شکل ۱۷-۱۰ را برای تهیه باقیمانده نرمال شده ممیز شناور در

AC تعمیم دهید. مانتیس باید یک کسر باشد.

حل: در تقسیم ممیز شناور داریم:

$$\begin{aligned} \frac{A}{B} &= \frac{m_A \times 2^{e_A}}{m_B \times 2^{e_B}} = \frac{0/1xxxx}{0/lyyyy} \times 2^{e_A - e_B} \\ &= 0/1zzzz \times 2^{e_A - e_B} + \frac{0,000000rrrr}{0/lyyyy} \times 2^{e_A - e_B} \end{aligned}$$

حال داریم:



- ۲۵- در الگوریتم‌های عمل‌های حسابی ممیز شناور در بخش ۵ - ۱۰ امکان سرریز یا فروریز در نما نادیده گرفته می‌شود.
- الف) سه فلوچارت را مرور کنید و جاهایی را که سرریز نما ممکن است رخ دهد پیدا کنید.
- ب) قسمت الف) را برای فروریز تکرار کنید. فروریز نما هنگامی رخ می‌دهد که نما از کوچکترین عددی که بتواند در ثبات جا گیرد کمتر شود.

ج) نشان دهید که سرریز یا فروریز نما می تواند با سخت افزار آشکار شود.
حل:

الف) جاهایی که نماها اضافه می شوند (جمع می شوند).

جاهایی که نماها افزایش می یابند (یکی زیادتیر می شوند).

ب) جاهایی که نماها منها می شوند.

جاهایی که نماها کاهش می یابند (یکی کم می شوند).

ج) می توان با چک کردن رقم نقلی نهایی بعد از عمل جمع و رقم نقلی بعد از اعمال کاهش یا افزایش، سرریز یا فروریز را آشکار کرد.

۲۶- اگر نمایش عدد صحیح را برای مانتیس عددهای ممیز شناور در نظر بگیریم، در حین ضرب و تقسیم با مشکلات خاصی روبرو می شویم. اجازه بدهید تعداد بیت های بخش مانتیس $(n-1)$ باشد.

برای نمایش صحیح:

الف) نشان دهید که اگر حاصلضرب با دقت معمولی بکار رود، باید $(n-1)$ به نمای حاصلضرب در AC اضافه شود.

ب) نشان دهید که اگر از مقسوم با مانتیس دقت معمولی استفاده شود، باید هنگامی که Q برابر ۰ می شود. $(n-1)$ از نمای مقسوم کم شود.

حل: الف) می دانیم ضرب دو عدد $(n-1)$ بیتی، یک عدد حداکثر $2(n-1)$ بیتی بوجود خواهد آورد. با توجه به مفهوم نرمال بودن و همچنین ظرفیت مانتیس که $(n-1)$ بیتی است، می بایست از $(n-1)$ بیت سمت راست صرف نظر کرده و $(n-1)$ به نما اضافه کنیم.

مثال:

بیت $n-1=3$

در ضرب روبرو

$$(111 \times 2^2) * (110 \times 2^2)$$

$$111 \times 110 = 101010 \times 2^4 = 101 \times 2^7$$

ب) در رابطه با تقسیم هم همین قضیه وجود دارد.

۲۷- سخت‌افزاری را که برای جمع و تفریق دو عدد دهدهی در نمایش اندازه

علامت‌دار به کار می‌رود نشان دهید. چگونگی آشکارسازی سرریز را نشان دهید.

حل:

۲۷) حل: سخت‌افزار جمع و تفریق شبیه به سخت‌افزار ضرب و تقسیم است (شکل ۱۰-۲۱)

تا این تفاوت که به A_e ، B_e ، Q و SC نیاز داریم. همین رقم شش به صورت مستقیم وارد E می‌شود. تا این سخت‌افزار جمع و تفریق به صورت زیر طراحی می‌شود:

$B_3 \quad \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} B$

$E \leftarrow \text{و اندر حساب BCD}$

$A_3 \quad \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} A$

بخش دادن سرریز تا این از علامت A (A_e)، علامت B (B_e)، نوع عمل (جمع و تفریق) و شماره رقم فعلی (E) می‌باشد. برای هرست آوردن این تابع، جدول حسی با تمثيل در هم و سپس با استفاده از جدول کارنو ساده سازی می‌کنیم.

E	A_3	$B_3(0)$	عمل	سرریز (V)
0	0	0	+	0
0	0	1	+	0
0	0	1	-	0
0	1	0	+	0
0	1	0	-	0
0	1	1	+	0
0	1	1	-	0
1	0	0	+	1
1	0	0	-	1
1	0	1	+	1
1	0	1	-	1
1	1	0	+	1
1	1	0	-	1
1	1	1	+	1
1	1	1	-	1

جدول کارنو:

		B_3		
		0	1	
E	0	0	0	0
	1	0	0	0
A_3	0	0	1	1
	1	1	1	1

نتیجه: $V = E(0 \odot A_3 \odot B_3)$

۲۸- نشان دهید که $673 - 356$ با جمع 673 با متمم 10 عدد 356 و چشم‌پوشی از نقلی پایانی انجام می‌شود. بلاک دیاگرام سه مرحله از واحد حسابی دهدهی را رسم کنید و نشان دهید چگونه این عمل پیاده‌سازی می‌شود. تمام بیت‌های ورودی و خروجی واحد حساب را لیست کنید.

حل:

$$673 - 356 = 317 \quad 10 \text{ مکمل } (356) = 644$$

$$\Rightarrow 673$$

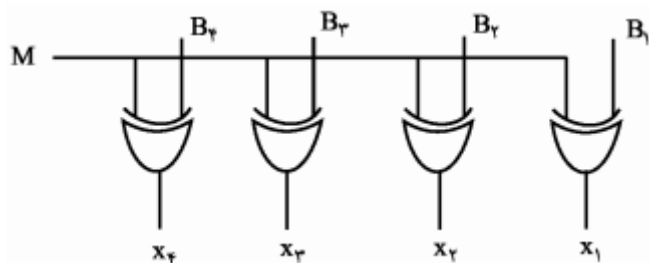
$$+ \underline{644}$$

$$317$$

و نقلی ۱

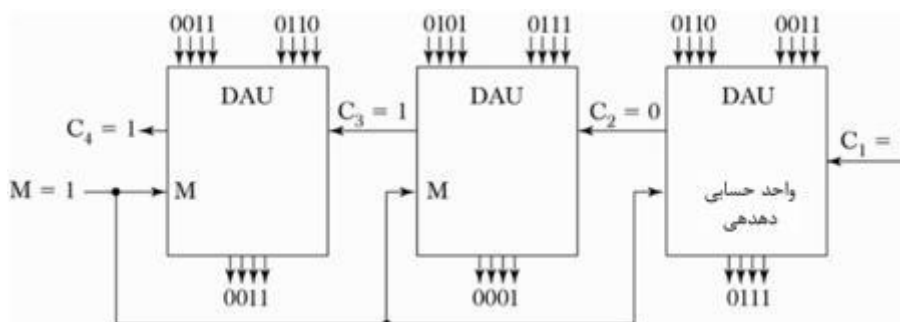
با استفاده از یک سویچ M مشخص می‌کنیم تفریق یا جمع انجام می‌شود به شکل

روبرو:

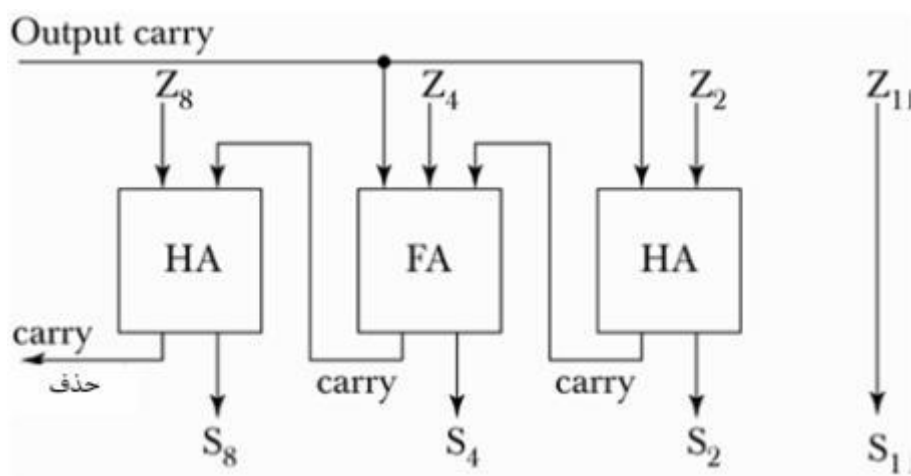


$A \pm B$

برای سه واحد حسابی دهدهی داریم:



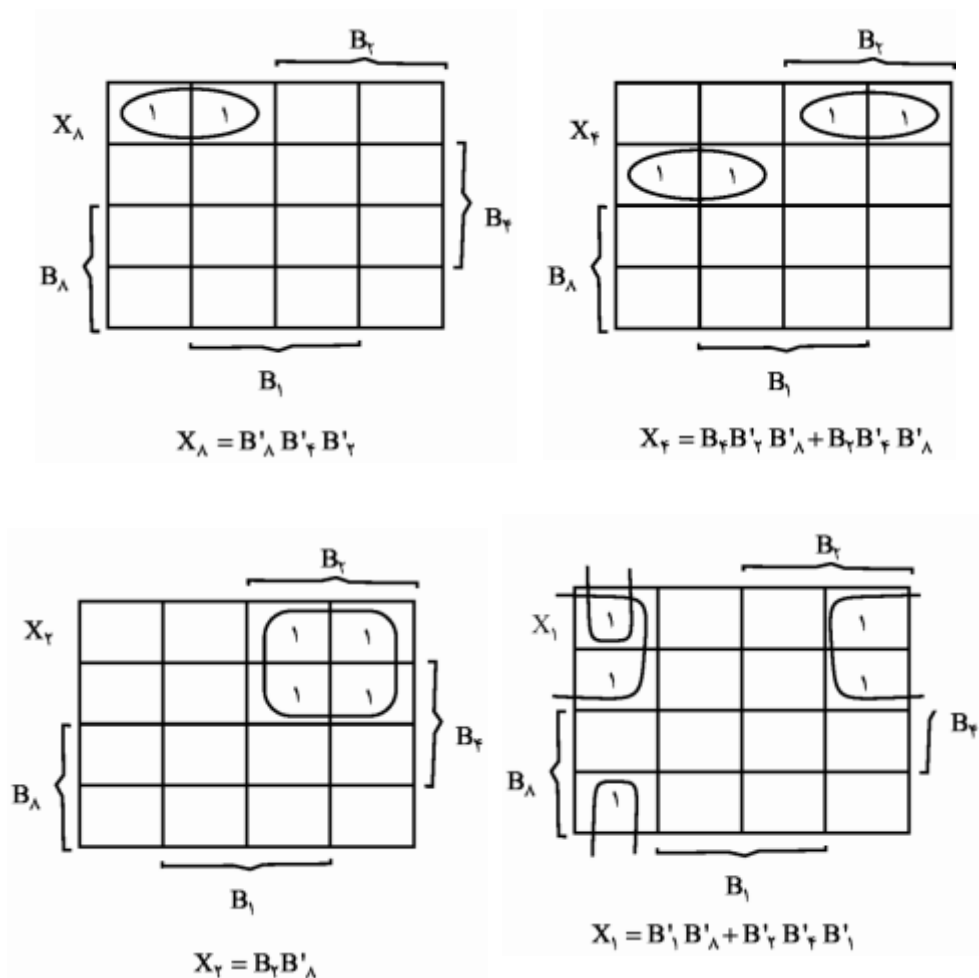
- ۲۹- نشان دهید که به جای جمع کننده دودویی چهار بیتی پایین رتبه در شکل ۱۰-۱۰ می توان یک تمام جمع کننده و دو نیم جمع کننده گذاشت.
- حل: دقت شود زمانی می توانیم به جای F.A. از H.A. استفاده کنیم که یا رقم نقلی ورودی نداشته باشیم و یا رقم نقلی خروجی نداشته باشیم.
- با توجه به شکل ۱۰-۱۰ کتاب، می توانیم داشته باشیم:



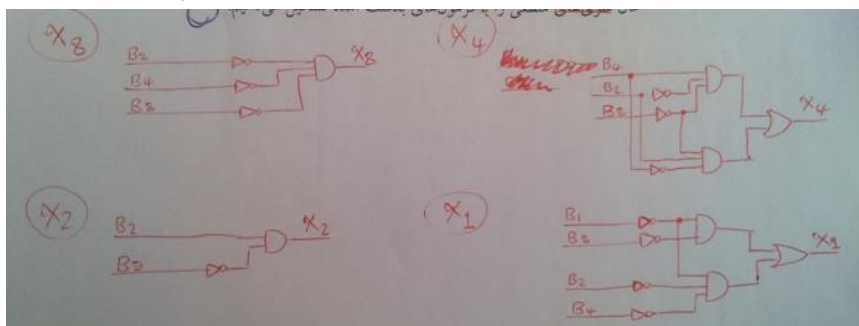
۳۰- با تکنیک‌های طراحی مدارهای ترکیبی، توابع بولی متمم‌ساز 9 و اعداد BCD شکل ۱۹-۱۰ را بدست آورید. دیاگرام منطقی آن را رسم کنید.
حل: جدول زیر را داریم:

	ورودی‌ها				خروجی‌ها				
	B ₈	B ₄	B ₂	B ₁	X ₈	X ₄	X ₂	X ₁	
0	0	0	0	0	1	0	0	1	9
1	0	0	0	1	1	0	0	0	8
2	0	0	1	0	0	1	1	1	7
3	0	0	1	1	0	1	1	0	6
4	0	1	0	0	0	1	0	1	5
5	0	1	0	1	0	1	0	0	4
6	0	1	1	0	0	0	1	1	3
7	0	1	1	1	0	0	1	0	2
8	1	0	0	0	0	0	0	1	1
9	1	0	0	1	0	0	0	0	0

طبق جدول، برای X_1, X_2, X_4, X_8 جداول کارنو را تشکیل داده تا مدار منطقی بهینه شده را بدست آوریم:



حال مدارهای منطقی را با فرمولهای بدست آمده تشکیل می‌دهیم.



۳۱- نیاز است برای دو رقم دهدهی در کد افزونی ۳ (جدول ۶ - ۳) یک جمع کننده طراحی شود. نشان دهید که تصحیح پس از جمع دو رقم با جمع کننده چهار بیتی به صورت زیر است:

الف) رقم نقلی خروجی برابر نقلی تصحیح نشده است.

ب) اگر نقلی خروجی برابر ۰ باشد، ۱۱۰۱ اضافه می شود و نقلی حاصل از این جمع نادیده گرفته می شود. نشان دهید که جمع کننده افزونی ۳ می تواند با هفت جمع کننده و دو وارونگر ساخته شود.

حل:

Dec (جمع دو عدد)	Z تصحیح نشده	Y تصحیح شده (خروجی)
0	0110	0011
1	0111	0100
2	1000	0101
3	1001	0110
4	1010	0111
5	1011	1000
6	1100	1001
7	1101	1010
8	1110	1011
9	1111	1100

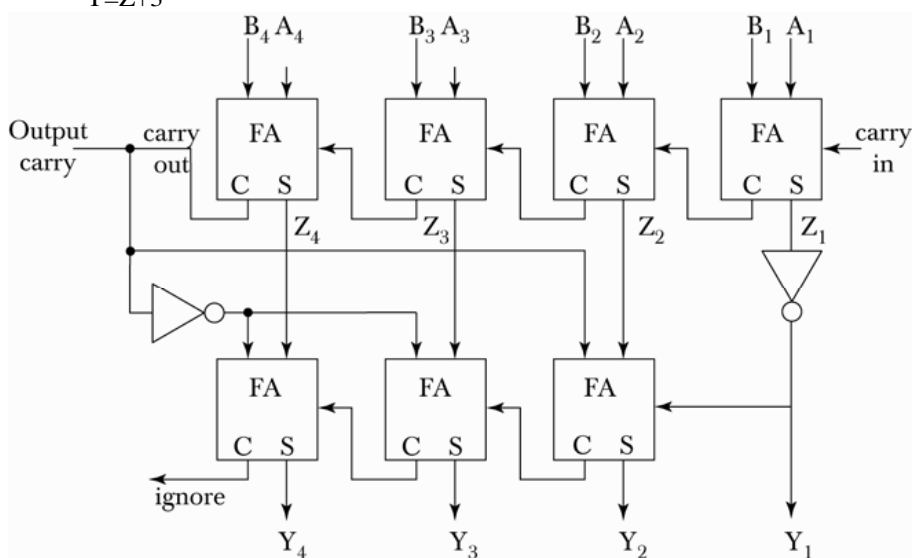
10	10000	10011
11	10001	10100
12	10010	10101
13	10011	10110
14	10100	10111
15	10101	11000
16	10110	11001
17	10111	11010
18	11000	11011
19	11001	11100

دقت شود زمانی که جمع دو عدد بین ۰ تا ۹ دسیمال شود، رقم نقلی نداریم.

(نادیده گرفتن رقم نقلی) $Y=Z-3 = Z+13-16$

رقم نقلی خروجی = رقم نقلی تصحیح نشده

$$Y=Z+3$$

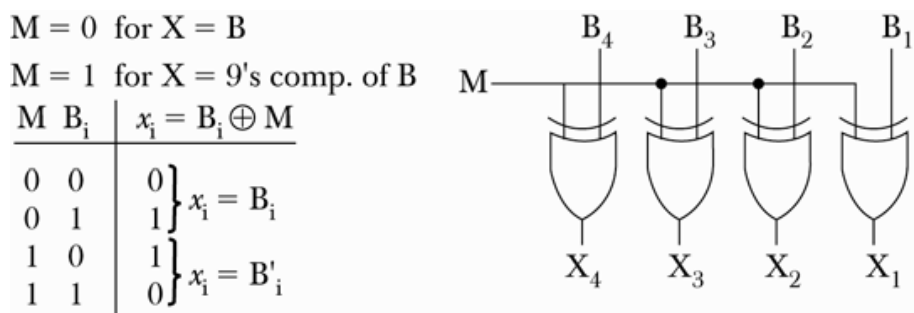


۳۲- مداری برای یک متمم ساز ۹، هنگامی که ارقام دهدهی به شکل کد افزونی

۳ نمایش داده می شوند، بدست آورید. یک ورودی کنترل مد تعیین می کند آیا رقم

متمم شده است یا نه. مزیت این کار نسبت به کد BCD چیست؟

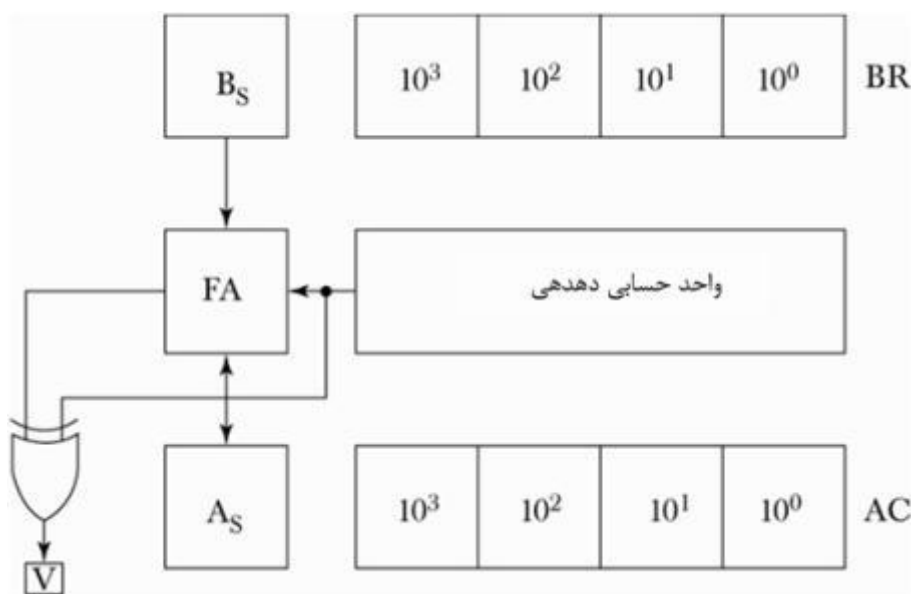
حل: در ارتباط با مزیت کد افزونی سه نسبت به کد BCD برای استفاده در ساخت متمم ساز 9، می‌دانیم کد افزونی ۳، خود مکمل است. بنابراین برای بدست آوردن متمم 9، کافی است متمم هر رقم را بنویسیم.
برای مدار داریم:



در این مدار با استفاده از سویچ M و ۱ کردن آن می‌توان مکمل هر بیت را بدست آورد.

۳۳- سخت‌افزار به کار رفته برای جمع و تفریق دو عدد دهدهی با نمایش متمم ۱۰ علامت‌دار عددهای منفی را نشان دهید. نشان دهید چگونه سرریز آشکار می‌شود. الگوریتمی به شکل فلوچارت تهیه کنید و خود را از درستی تولید پاسخ آن مطمئن سازید.

حل: اساس کار الگوریتم درخواستی مانند الگوریتم داده شده برای جمع و تفریق در سیستم متمم (۲) دودویی است. (فلوچارت شکل ۲-۱۰)
در ضمن در ارتباط با تشخیص خطای سرریز داریم:



۳۴- محتوای ثبات‌های A و E و Q و SC را در حین ضرب دهمی (شکل ۲۲-۱۰) الف) 470×152 و ب) 999×199 بدست آورید. ثبات‌ها را سه رقمی فرض کنید و دومین عدد را مضروب فیه در نظر بگیرید.

حل: برای نمونه (الف) را حل می‌نماییم

الف) $B = 470$ مضروب فیه

	Ae	A	Q	sc
Initial	0	000	152	3
$Q_L \neq 0$ -----	0	470	151	
$Q_L \neq 0$ -----	0	940	150	
$Q_L = 0$, d shr -----	0	094	015	2
	0	564	014	
$Q_L \neq 0$ -----	1	034	013	
	1	504	012	
	1	974	011	
	2	444	010	
$Q_L = 0$, dshr -----	0	244	401	1
$Q_L \neq 0$ -----	0	714	400	
$Q_L = 0$, dshr -----	0	071	440	0

۳۵- محتوای ثبات‌های A و E و Q و SC را در حین تقسیم دهدهی (شکل

۲۳-۱۰) ۱۶۸۰/۳۲ نشان دهید. ثبات‌ها را دو رقمی در نظر بگیرید.

حل:

E	AC	A	Q	SC
0	0	16	80	2
	1	68	00	
	9	68		
1	1	36	01	
	9	68		
1	1	04	02	
	9	68		
1	0	72	03	
	9	68		
1	0	40	04	
	9	68		
1	0	08	05	
	9	68		
0	9	76		
	0	32		
1	0	08	05	1

E	AC	A	Q	SC
--	0	80	50	1
--	9	68		
1	0	48	51	
	9	68		
1	0	16	52	
	9	68		
0	9	84		
	0	32		
1	0	16	52	0

خارج قسمت باقی مانده

۳۶- نشان دهید که زیر ثبات A_e در شکل (۲۱-۱۰) در پایان الف) ضرب دهنده‌ی که در شکل (۲۲-۱۰) مشخص است، و (ب) در تقسیم شکل (۲۳-۱۰)، صفر است.

حل: الف) در شکل (۲۲-۱۰)، در پایان عملیات ضرب، محتویات رجیستر A ، شیفت به راست داده می‌شود تا مقدار A_e صفر شود.

ب) در شکل (۲۳-۱۰) در پایان عملیات تقسیم، B ، به مقدار $(A + \bar{B} + 1)$ اضافه می‌شود. طبق سیستم اعداد BCD، A_e مقدار ۹ را دارد. وقتی $Be=0$ اضافه می‌شود، رقم نقلی تولید می‌شود و $A_e=0$ می‌شود.

۳۷- الگوریتم‌های حساب ممیز شناور در بخش‌های (۵-۱۰) را از داده‌های

دودویی به داده‌های دهمی تغییر دهید. در یک جدول شیوه تفسیر هر یک از نمادهای ریز عمل‌ها را بنویسید.

حل: با در نظر گرفتن جدول (۱-۱۰) سمبل‌ها را انتخاب می‌کنیم و همچنین با در نظر گرفتن الگوریتم‌های بخش (۴-۱۰)، و در نظر گرفتن مانتیس در الگوریتم‌های ضرب و تقسیم، تبدیلات لازم را انجام می‌دهیم.



فصل یازدهم: سازمان ورودی-خروجی

۱- آدرس‌های واگذار شده به چهار ثبات واسط I/O در شکل ۱۱-۲ برابر با معادل دودویی اعداد ۱۲، ۱۳، ۱۴ و ۱۵ است. مدار خارجی لازم بین یک آدرس I/O هشت بیتی از CPU و در ورودی‌های CS، RS1 و RS0 از مدار واسطه وصل شوند را نشان دهید.

حل:

$$\begin{array}{rcl}
 & A_7 \dots A_2 & A_1 A_0 \\
 12 = & 000011 & 0 \ 0 \\
 13 = & 000011 & 0 \ 1 \\
 14 = & 000011 & 1 \ 0 \\
 15 = & 000011 & 1 \ 1 \\
 & & \begin{array}{cc} \swarrow & \searrow \\ RS_1 & RS_0 \end{array}
 \end{array}
 \Rightarrow
 \begin{array}{l}
 CS = A_2 A_3 A'_4 A'_5 A'_6 A'_7 \\
 RS_1 = A_1 \\
 RS_0 = A_0
 \end{array}$$

(بیت‌های A_7 تا A_2 یکسان‌اند که با استفاده از یک AND 6 ورودی به CS متصل می‌شوند و دو بیت آخر همان RS1 و RS0 هستند.)

۲- شش واحد واسط از نوعی که در شکل ۱۱-۲ نشان داده شده به یک CPU که از آدرس I/O هشت بیتی استفاده می‌کند متصل‌اند. هر یک از شش ورودی انتخاب تراشه CS به خط آدرس متفاوتی متصل است، بنابراین خط با ارزش‌تر آدرس به ورودی CS اولین واحد واسط و ششمین خط آدرس به ششمین واحد واسط متصل است. دو خط آدرس کم ارزش‌تر به RS1 و RS0 از هر شش واحد واسط متصل‌اند. آدرس هشت بیتی ثبات‌های هر واسطه را تعیین کنید.

حل:

واسط	درگاه A	درگاه B	ثبات کنترل	ثبات وضعیت
1 ≠	10000000	10000001	10000010	10000011
2	01000000	01000001	01000010	01000011
3	00100000	00100001	00100010	00100011
4	00010000	00010001	00010010	00010011
5	00001000	00001001	00001010	00001011
6	00000100	00000101	00000110	00000111

۳- چهار وسیله جانبی که خروجی قابل درکی برای انسان داشته باشد نام ببرید.

حل: چاپگر - پلاتر - صفحه نمایش - خروجی صدا

۴- نام کامل خود را به ASCII با هشت بیت برای هر کاراکتر و یک ۰ در سمت چپ‌ترین مکان بنویسید. بین قسمت‌های مختلف نام یک فاصله بگذارید و نیز در صورت به کار بردن مخفف، پس از آن نقطه بگذارید.

حل:

M A N O (MANO)
01001101 01000001 01001110 01001111

۵- تفاوت I/O مجزا با I/O نگاهت چیست؟ خوبی‌ها و بدی‌های هر یک را

بگویید.

حل: در I/O مجزا برای هر دو بخش حافظه و I/O از یک گذرگاه مشترک استفاده می‌شود ولی هر کدام خطوط کنترل جداگانه دارند و برای نوشتن و خواندن از I/O دستورالعمل‌های مخصوص وجود دارد اما در I/O نگاهت هر دو بخش حافظه و I/O از گذرگاه و خطوط کنترل مشترک استفاده می‌کنند و دستورالعمل جداگانه‌ای برای ورودی و خروجی وجود ندارد.

I/O مجزا این مزیت را دارد که آدرس‌های حافظه و I/O را طوری از هم جدا می‌کند که حافظه به وسیله آدرس‌های تخصیص یافته در مدار واسط تحت تأثیر قرار نمی‌گیرد زیرا حافظه و I/O هر کدام فضای آدرس مربوط به خود را دارند. عیب این روش وجود دستورالعمل‌های مجزا برای خواندن و نوشتن در I/O است. در I/O نگاهت دستورالعمل خاصی برای I/O وجود ندارد و CPU قادر است داده‌های I/O واقع در

ثبات واسطه را با همان دستوراتی که برای حافظه به کار می‌روند دستکاری کند. عیب این روش آن است که در صورت بروز خطا در محاسبه آدرس حافظه، I/O تحت تأثیر قرار می‌گیرد و برعکس.

۶ - مشخص کنید کدام یک از موارد زیر یک فرمان کنترل، وضعیت یا انتقال داده است.

(الف) گذر از دستورالعمل بعدی اگر پرچم یک باشد. ← فرمان وضعیت (وضعیت یک پرچم چک می‌شود).

(ب) جستجوی یک رکورد خاص روی یک دیسک مغناطیسی ← فرمان کنترل (هد مغناطیسی را روی دیسک جابه‌جا می‌کند).

(ج) بررسی آماده بودن یا نبودن وسیله I/O ← فرمان وضعیت (آماده بودن وسیله را بررسی می‌کند)

(د) حرکت کاغذ چاپگر به ابتدای صفحه بعد ← فرمان کنترل (کاغذ را جابه‌جا می‌کند).

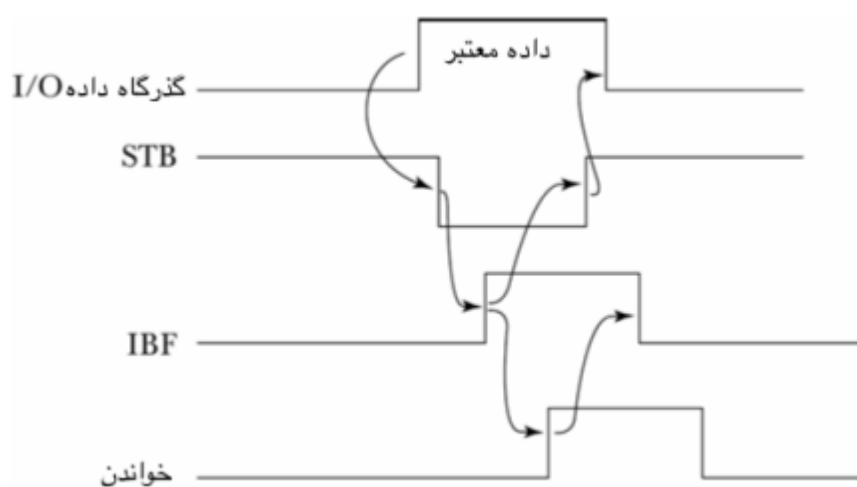
(ه) خواندن ثبات وضعیت واسط ← فرمان انتقال داده (مقدار یک ثبات خوانده می‌شود).

۷- یک واحد واسط تجاری برای خطوط دست‌دهی مربوط به انتقال داده‌ها از وسیله I/O به واحد واسط از نام‌های مختلفی استفاده می‌کند. خط دست‌دهی ورودی واسط STB (مخفف استروب یا فعال‌ساز) و خط دست‌دهی خروجی واسط IBF (مخفف پر بودن بافر ورودی) نام‌گذاری شده است. وجود یک سیگنال سطح بالا روی IBF نشان می‌دهد که داده توسط واسط پذیرفته شده است. پس از سیگنال خواندن I/O از طرف CPU از طریق ثبات داده، IBF به سطح پایین می‌رود.

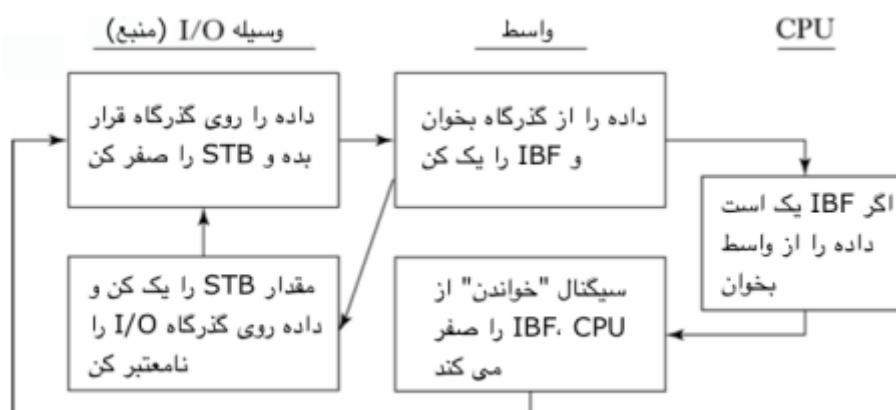
(الف) یک بلاک دیاگرام که CPU، واسط و وسیله I/O همراه با اتصالات لازم بین آنها را نشان دهد رسم کنید.



ب) یک دیاگرام زمانی برای انتقال دسته‌ی رسم کنید.



ج) برای انتقال از وسیله به واسط و از واسط به CPU یک نمودار دنباله رخدادها رسم کنید.

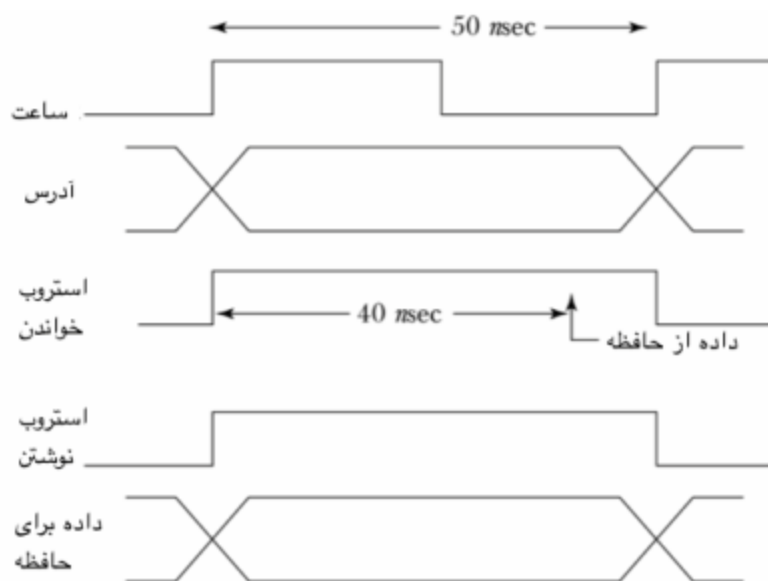


۸- یک CPU با فرکانس ساعت 20 MHz به یک واحد حافظه که زمان دسترسی آن ۴۰ns است وصل شده است. نمودارهای زمانی خواندن و نوشتن آنها را با فعال‌کننده‌های READ و WRITE تهیه کنید. در نمودار زمانی آدرس را نیز نشان دهید.

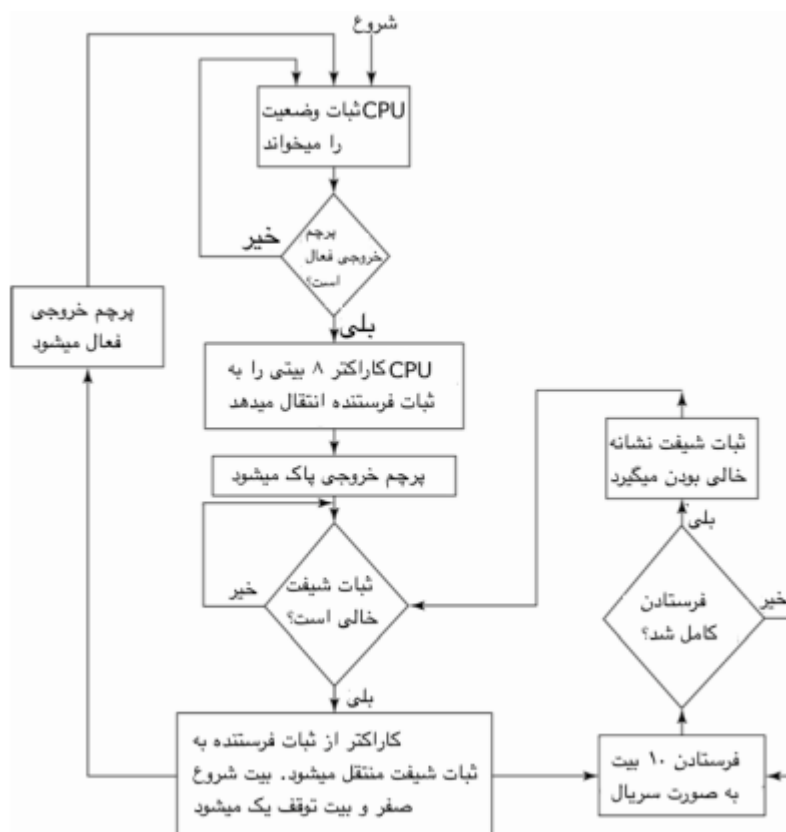
حل:

$$20 \text{ MHz} = 20 \times 10^6 \text{ Hz} \Rightarrow T = \frac{10^{-6}}{20} = 50 \text{ ns}$$

بنابراین دیاگرام به صورت زیر است:



- ۹- مدار واسط ارتباطی غیرهمگام شکل ۸-۱۱ بین یک CPU و یک چاپگر وصل شده است. فلوچارتی رسم کنید که دنباله عمل های فرستادن کاراکتر از CPU به چاپگر را نشان دهد.
- حل: (ثبات ها مربوط به شکل ۸-۱۱ هستند. پرچم خروجی یک بیت در ثبات وضعیت است.)



۱۰- حداقل شش وضعیت مختلف را برای ۱ کردن بیت‌های ثبات وضعیت در یک واسط ارتباطی غیرهمگام ارائه دهید.

حل:

۱- پرچم خروجی زمانی که ثبات انتقال خالی است.

۲- پرچم ورودی زمانی که ثبات دریافت پر است.

۳- فعال کردن وقفه اگر یک پرچم فعال است.

۴- خطای توازن

۵- خطای کادربندی

۶- خطای بازنویسی

۱۱- اگر واسط به پایانه‌ای وصل باشد که به یک بیت توقف نیاز دارد، چند بیت در ثبات شیفت فرستنده شکل ۸- ۱۱ وجود دارد؟ با استفاده از کد اسکی توازن زوج بیت‌های ثبات شیفت را وقتی که حرف W ارسال شود مشخص کنید.
حل:

۱۰ بیت : بیت شروع + ۷ بیت اسکی + بیت توازن + بیت توقف

از جدول ۱۱-۱ داریم:

کد اسکی W : ۱۰۱۰۱۱۱

با توازن زوج : ۱۱۰۱۰۱۱۱

با بیت شروع و توقف: ۱۱۱۰۱۰۱۱۱۰

۱۲- بر روی یک خط 1200-Baud با هر یک از روش‌های زیر چند کاراکتر در هر ثانیه فرستاده می‌شود؟ (کد کاراکترها را هشت بیتی فرض کنید).
الف) ارسال سری همگام (همزمان)

$$\frac{1200}{8} = 150 \text{ کاراکتر بر ثانیه}$$

ب) ارسال غیر همگام (با دو بیت توقف)

$$\frac{1200}{11} = 109 \text{ کاراکتر بر ثانیه}$$

ج) ارسال سری غیر همگام (با یک بیت توقف)

$$\frac{1200}{10} = 120 \text{ کاراکتر بر ثانیه}$$

۱۳- اطلاعاتی با سرعت m بایت در ثانیه به یک بافر FIFO وارد می‌شود. اطلاعات پاک شده از آن نیز n بایت در ثانیه است، بیشینه ظرفیت بافر ۱۲ بایت است.

الف) اگر $m > n$ باشد چه مدت طول می‌کشد تا بافر خالی، پر شود؟

ظرفیت بافر

بایت k

اختلاف سرعت ورود و پاک شدن

ثانیه / بایت (m-n)

زمان پر شده = _____ = _____

$$\text{ثانیه} = \frac{12}{m-n} = \text{زمان پر شدن بافر}$$

ب) اگر $m < n$ چقدر طول می‌کشد تا بافر پر، خالی شود؟

همانند زمان پر شدن داریم:

$$\text{ثانیه} = \frac{12}{n-m} = \text{زمان خالی شدن بافر}$$

ج) اگر $m = n$ باشد آیا بافر FIFO لازم است؟

خیر، زمانی به بافر نیاز داریم که اختلاف سرعت وجود داشته باشد و لازم باشد یکی منتظر دیگری بماند.

۱۴- بیت‌های ثبات کنترل FIFO در شکل ۹-۱۱ عبارتند از $F_1F_2F_3F_4 = 0011$

رشته عمل‌های داخلی را هر گاه یک داده از FIFO حذف و سپس داده جدیدی وارد شود تعیین کنید.

حل: با توجه به شکل ۹-۱۱ داریم:

حالت اولیه	$F = 0011$	$\text{output} \leftarrow R_4$
بعد از حذف ۱	$F = 0010$	
بعد از حذف ۰	$F = 0001$	$R_4 \leftarrow R_3$
بعد از ورود ۱	$F = 1001$	$R_1 \leftarrow \text{input}$
داده وارد شده به انتهای صف می‌رود.	$F = 0101$	$R_2 \leftarrow R_1$
	$F = 0011$	$R_3 \leftarrow R_2$

۱۵- مقدارهای «ورودی آماده» و «خروجی آماده» و بیت‌های کنترل F_1 تا F_4 در

شکل ۹-۱۱ در وضعیت‌های زیر چیست؟

«ورودی»	«خروجی»	$F_1 - F_4$	
«آماده»	«آماده»		
۱	۰	۰۰۰۰	الف) وقتی بافر خالی است.
۰	۱	۱۱۱۱	ب) وقتی بافر پر است.
۱	۱	۰۰۱۱	ج) وقتی بافر حاوی دو قلم داده است.

۱۶- یک بلاک دیاگرام مشابه شکل ۱۰-۱۱ را برای جابجایی داده از CPU به یک واسطه و سپس به یک وسیله I/O رسم کنید. رویه‌ای برای ۱ کردن و ۰ کردن بیت پرچم ارائه دهید.

حل:



رویه ۱ کردن و صفر کردن پرچم.

flag = 0 اگر ثبات داده پر باشد. (بعد از نوشتن توسط CPU)

flag = 1 اگر ثبات داده خالی باشد (بعد از انتقال به وسیله جانبی)

هنگامی که flag صفر می‌شود، Data Ready فعال شده و داده روی گذرگاه I/O

قرار می‌گیرد. هنگامی که acknowledge فعال شود؛ flag را ۱ کرده و خط دست‌دهی

Ready را غیرفعال می‌کند.

۱۷- با پیکربندی شکل مسأله ۱۶-۱۱ فلوچارتی (مشابه شکل ۱۱-۱۱) برای

برنامه CPU جهت خروجی داده بدست آورید.

حل:



۱۸- خوبی اصلی استفاده از جابجایی داده به کمک وقفه در برابر جابجایی

تحت کنترل برنامه بدون وقفه چیست؟

حل: انتقال تحت کنترل برنامه به دلیل تفاوت سرعت انتقال اطلاعات بین CPU و وسیله I/O روش کارایی نیست (زمان CPU را هدر می‌دهد) اما در انتقال به کمک وقفه CPU مادامی که مشغول اجرای یک برنامه است پرچم را واریسی نمی‌کند و هنگام ۱ شدن پرچم، وقفه‌ای اجرای برنامه جاری را متوقف می‌کند و ۱ شدن پرچم را به CPU اطلاع می‌دهد در این حالت CPU روال سرویس وقفه را انجام می‌دهد و دوباره به برنامه اصلی باز می‌گردد که باعث تلف نشدن وقت CPU برای بررسی فعال شدن

پرچم می‌شود.

۱۹- در بیشتر کامپیوترها وقفه فقط پس از اجرای دستورالعمل جاری تشخیص داده می‌شود. امکان پذیرش وقفه را در هر زمانی در حین اجرای دستور بررسی کنید. در مورد مشکلاتی که ممکن است بروز کند بحث کنید.

حل: اگر وقفه‌ای در حین اجرای یک دستور تشخیص داده شود، لازم است که علاوه بر ثبات‌های پردازنده، تمام اطلاعات ثبات‌های کنترل نیز ذخیره شود (در پایان اجرای دستور مقدار ثبات‌های کنترل مهم نیستند و نیازی به ذخیره آنها نیست) این کار باعث می‌شود ذخیره‌ی وضعیت CPU بسیار پیچیده‌تر شود. همچنین اجرای دستوراتی مانند ION که باید بطور کامل انجام شوند نیز ممکن است دچار اختلال شود.

۲۰- در سیستم وقفه اولویت‌دار چرخشی شکل ۱۲-۱۱، هر گاه پس از درخواست وقفه وسیله ۲ به CPU و قبل از تصدیق به وسیله CPU، وسیله ۱ درخواست وقفه کند چه می‌شود؟

حل:

- ۱- در ابتدا وسیله ۲ یک درخواست وقفه به CPU می‌فرستد.
- ۲- قبل از این که CPU تصدیق کند وسیله ۱ درخواست وقفه به CPU می‌فرستد.
- ۳- بعد از این که CPU تصدیق می‌کند، وسیله ۱ دارای اولویت است و آدرس آن فعال شده و به CPU ارسال می‌شود.

۲۱- کامپیوتری را بدون سخت‌افزار وقفه اولویت‌دار در نظر بگیرید. هر یک از چند منبع می‌توانند به کامپیوتر وقفه دهند، هر درخواست وقفه موجب ذخیره آدرس برگشت و انشعاب به یک روال مشترک وقفه می‌گردد. توضیح دهید چگونه می‌توان در برنامه سرویس‌دهی وقفه، اولویت برقرار کرد.

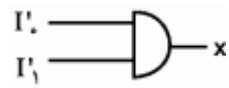
حل:

برای این کار، در روال مشترک وقفه، هنگام چک کردن منبع وقفه (شرط if-else)،

منابع
همز ما

۲
را برا

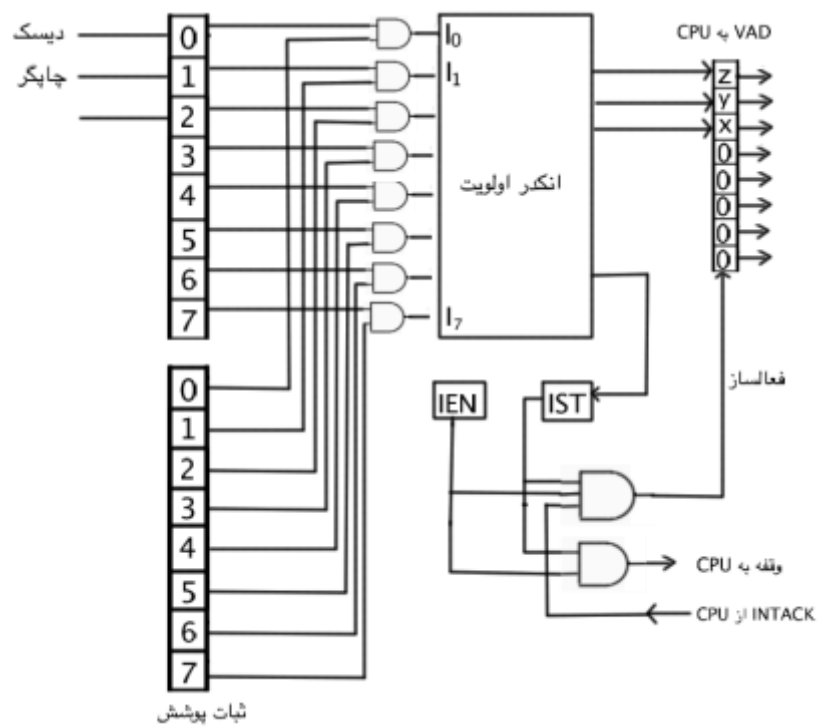
7



۲۳- سخت افزار وقفه اولویت دار موازی را برای سیستمی با هشت منبع وقفه

طراحی کنید.

حل:



۲۴- جدول درستی انکدر 8×3 اولویت را بدست آورید. فرض کنید که سه خروجی xyz از انکدر اولویت برای تهیه آدرس بردار به و به شکل $101xyz00$ بکار روند. هشت آدرس بردار را با شروع از آن که بالاترین اولویت را دارد بنویسید.

حل:

l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7	x	y	z	lst
1	x	x	x	x	x	x	x	0	0	0	1
0	1	x	x	x	x	x	x	0	0	1	1
0	0	1	x	x	x	x	x	0	1	0	1
0	0	0	1	x	x	x	x	0	1	1	1
0	0	0	0	1	x	x	x	1	0	0	1
0	0	0	0	0	1	x	x	1	0	1	1
0	0	0	0	0	0	1	x	1	1	0	1
0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	x	x	x	0

دودویی	مبنای 16
1010 0000	A0
1010 0100	A4
1010 1000	A8
1010 1100	AC
1011 0000	B0
1011 0100	B4
1011 1000	B8
101 11100	BC

۲۵- در شکل ۱۴-۱۱ چه باید کرد تا چهار مقدار VAD برابر با ۷۶، ۷۷، ۷۸، ۷۹

گردند؟

حل:

$$76 = (01001100)_2$$

پس کافی است ۶ عدد صفر را با ۰۱۰۰۱۱ جایگزین کنیم تا اعداد مورد نظر بدست آیند.

۲۶- برای بررسی زمانی که منبع در حال سرویس دهی به وقفه قبلی به کامپیوتر

به وسیله همان منبع، وقفه می دهد، چه برنامه ای نیاز است؟

حل: ابتدا بیت ماسک مربوط به منبع وقفه را فعال می کنیم که بتواند دوباره وقفه بدهد. برای تشخیص، در شروع روال سرویس مقدار آدرس بازگشت را در پشته بررسی می کنیم. اگر این مقدار، آدرسی درون برنامه سرویس منبع باشد، بدین معنی است که همان منبع در هنگام سرویس دهی دوباره وقفه ایجاد کرده است.

۲۷- چرا خط های کنترل خواندن و نوشتن در کنترل کننده DMA دو طرفه

است؟ در چه شرایطی و به چه منظوری از آنها به عنوان ورودی استفاده می شود؟

در چه شرایطی و به چه منظوری از آنها به عنوان خروجی استفاده می شود؟

حل: هنگامی که CPU با کنترل کننده DMA ارتباط برقرار می کند، خطوط خواندن و نوشتن به عنوان ورودی از CPU به DMA استفاده می شوند. این کار امکان برقراری ارتباط با ثبات های DMA را برای CPU فراهم می کند.

هنگامی که کنترل کننده DMA با حافظه ارتباط برقرار می کند، خطوط خواندن و نوشتن به عنوان خروجی از DMA به حافظه استفاده می شوند و عمل خواندن یا نوشتن را مشخص می کنند.

۲۸- می خواهیم ۲۵۶ کلمه را از یک دیسک مغناطیسی به بخشی از حافظه که از

آدرس ۱۲۳۰ شروع می شود جابجا کنیم. جابجایی با DMA طبق شکل ۱۸-۱۱

صورت می‌گیرد، روند این کار چگونه خواهد بود؟

حل: الف) CPU، DMA را با انتقال مقادیر زیر مقداردهی اولیه می‌کند:

۲۵۶ به ثبات شمارنده کلمه

۱۲۳۰ به ثبات آدرس DMA

بیت‌هایی به ثبات کنترل برای مشخص کردن عمل نوشتن

ب) ۱- وسیله I/O یک درخواست DMA می‌فرستد.

۲- DMA درخواست گذرگاه (BR) به CPU می‌فرستد.

۳- CPU با اعطای گذرگاه (BG) پاسخ می‌دهد.

۴- محتوای ثبات آدرس DMA روی گذرگاه آدرس قرار می‌گیرد.

۵- DMA سیگنال «تصدیق DMA» را به وسیله I/O می‌فرستد و خط کنترل نوشتن حافظه را فعال می‌کند.

۶- کلمه داده توسط وسیله I/O روی گذرگاه داده قرار می‌گیرد.

۷- ثبات آدرس DMA یک واحد افزایش پیدا کرده و ثبات شمارنده کلمه DMA یک واحد کاهش می‌یابد.

۸- مراحل ۴ تا ۷ برای انتقال هر کلمه تکرار می‌شود.

۲۹- یک کنترل‌کننده DMA کلمه‌های ۱۶ بیتی را با سرقت سیکل به حافظه منتقل می‌کند. این کلمه‌ها از وسیله‌ای دریافت می‌شود که کاراکترها را با سرعت ۲۴۰۰ کاراکتر در ثانیه می‌فرستد. CPU دستورها را با سرعت یک میلیون در ثانیه برداشت و اجرا می‌کند. CPU در اثر جابجایی به شیوه DMA تا چه اندازه کند می‌شود؟

حل: CPU در هر $1\mu s (1/10^6)$ یک بار (یا بیشتر) بر حافظه مراجعه می‌کند. کاراکترها در هر $1/2400 = 4\frac{2}{3}\mu s$ دریافت می‌شوند. دو کاراکتر ۸ بیتی بسته‌بندی شده در کلمه ۱۶ بیتی هر $2 \times 416,6 = 833,3\mu s$ دریافت می‌شوند. پس هر $833,3/3$ سیکل از CPU، یک سیکل برای دریافت کاراکتر استفاده می‌شود. بنابراین حداکثر کاهش سرعت

CPU برابر است با:

$$\frac{1}{833,3} \times 100 = 0,12\%$$

۳۰- چرا DMA به هنگام درخواست جابجایی به حافظه نسبت به CPU به هنگام

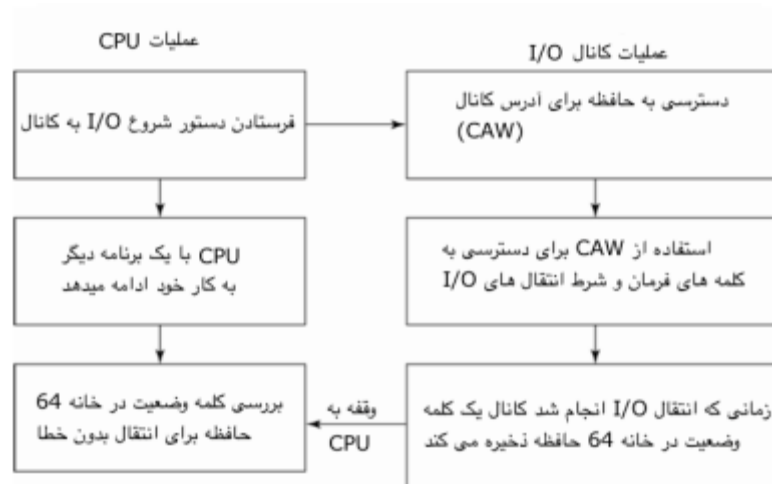
درخواست جابجایی به حافظه اولویت دارد؟

حل: CPU می‌تواند برای برداشت دستورها و داده از حافظه بدون هیچ آسیبی به جز از دست دادن زمان منتظر بماند اما DMA معمولاً داده را از وسیله‌ای انتقال می‌دهد که نمی‌تواند متوقف شود چون جریان ورودی اطلاعات ادامه دارد و ممکن است داده‌هایی از بین برود.

۳۱- فلوچارتی شبیه ۲۰-۱۱ برای IBM 370 رسم کنید و ارتباط CPU با گذرگاه

جابجایی I/O را توضیح دهید.

حل:



CPU مستقیماً از طریق خطوط کنترل اختصاصی با کانال‌ها و از طریق نواحی رزرو

شده ذخیره‌سازی در حافظه بطور غیرمستقیم تبادل اطلاعات می‌کند. شکل ۲۱-۱۱ در

کتاب قالب کلمات مربوط به عملکرد کانال را نشان می‌دهد.

۳۲- آدرس یک پایانه متصل به یک پردازنده انتقال داده از دو حرف الفبای انگلیسی یا یک حرف و به دنبال آن یکی از ده رقم تشکیل شده است. چند آدرس مختلف می‌توان ایجاد کرد؟

۲۶ حرف و ۱۰ رقم وجود دارد:

$$26 \times 26 + 26 \times 10 = 936 = \text{تعداد آدرس ممکن}$$

۳۳- رویه‌ای را برای خط جابجایی دنباله کاراکترها، به منظور ارتباط بین یک پردازنده جابجایی داده و یک پایانه دوردست طراحی کنید. پردازنده می‌پرسد که پایانه در حال کار است یا خیر. پایانه با بله و خیر به آن پاسخ می‌دهد. اگر پاسخ بلی باشد پردازنده بلاکی از متن برای آن می‌فرستد.

حل: پردازنده آدرس ترمینال را همراه با کد ENQ (پرسش) (۰۱ ۰۱ ۰۰۰۰)

می‌فرستد.

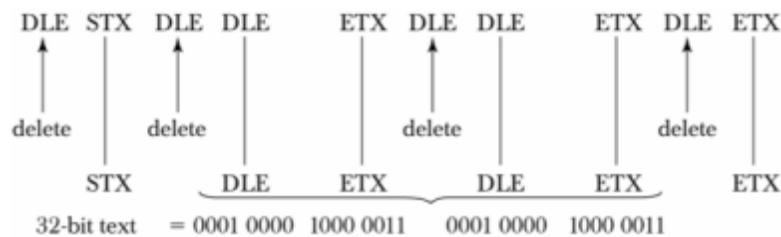
ترمینال با ACK (تصدیق) یا NAK (عدم تصدیق) پاسخ می‌دهد یا اینکه در طول یک فاصله زمانی پاسخ نمی‌دهد. اگر پردازنده یک ACK دریافت کرد، یک بلاک از متن را می‌فرستد.

۳۴- یک ارتباط جابجایی داده از قرارداد مبتنی بر کاراکتر با شفافیت داده‌های بکارگیری کاراکتر DLE استفاده می‌کند. پیام متنی که فرستنده بین STX و ETX می‌فرستد به صورت زیر است:

DLE STX DLE DLE ETX DLE DLE ETX DLE ETX

داده‌های شفاف متنی را به صورت دودویی بنویسید.

حل:



۳۵- کمترین تعداد بیت‌هایی که یک کادر در قرارداد مبتنی بر بیت دارد چقدر

است؟

حل: ۳۲ بیت بین پرچم‌ها؛ ۴۸ بیت همراه با پرچم‌ها (با توجه به قرارداد مبتنی بر

بیت)

۳۶- نشان دهید چگونه روش درج صفر در قرارداد مبتنی بر بیت، وقتی که یک

صفر و به دنبال آن ده بیت معادل دودویی ۱۰۲۳ فرستاده می‌شود کار می‌کند.

حل:

اطلاعاتی که قرار است فرستاده شود (۱۰۲۳): ۰۱۱۱۱۱۱۱۱

بعد از درج صفر، اطلاعاتی که فرستاده می‌شود: ۰۱۱۱۱۱۰۱۱۱۱۱۰

اطلاعات دریافت شده بعد از حذف صفر: ۰۱۱۱۱۱۱۱۱۱

فصل دوازدهم: سازمان حافظه

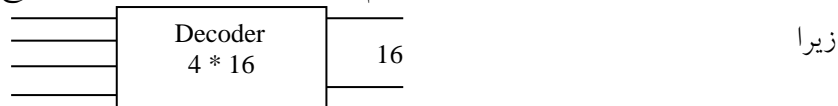
۱- الف) چند تراشه Ram ، 128×8 برای ساخت یک حافظه با ظرفیت ۲۰۴۸ لازم است؟

ب) چند خط گذرگاه آدرس نیاز است تا ۲۰۴۸ بایت حافظه دستیابی شود؟
چند خط از این خط ها برای همه تراشه ها مشترک است؟
پ) چند خط برای انتخاب تراشه باید دیکد شود؟ اندازه دیکدر را تعیین کنید.
حل:

$$\frac{2048}{128} = 16 \text{ الف) تعداد تراشه Ram مورد نیاز:}$$

$$2048 = 2^{11} \Rightarrow \text{ب) ۱۱ خط آدرس نیاز داریم}$$

برای دیکدر انتخاب و فعال سازی هر کدام از تراشه های Ram، ۴ خط احتیاج است



پس خطوط آدرس مشترک $11 - 4 = 7$

پ) همان طور که گفته شد، دیکدر ۴ به ۱۶ نیاز خواهد بود.

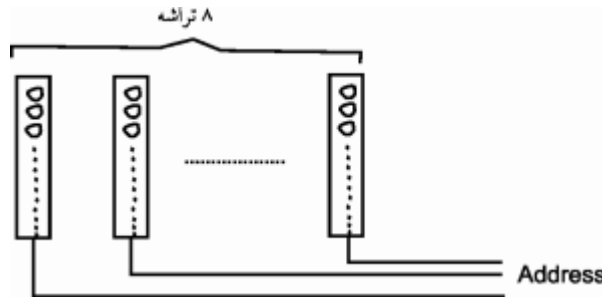
۲- کامپیوتری از یک RAM با ظرفیت 1024×1 استفاده می کند.

الف) چند تراشه نیاز است و چگونه خط های آنها باید متصل شود تا ۱۰۲۴ بایت حافظه حاصل شود.

ب) چند تراشه برای تهیه ۱۶k بایت حافظه نیاز است؟ توضیح دهید که چگونه تراشه ها باید به گذرگاه حافظه وصل شوند.

حل:

الف) برای ساخت تراشه RAM ، 1024×8 از تراشه رم 1024×1 ، ۸ تراشه به صورت موازی به هم متصل می شوند. مانند شکل زیر:



ب) در کل، ۱۲۸ تراشه Ram 1024×1 لازم است. $16 \times 8 = 128 \Rightarrow$

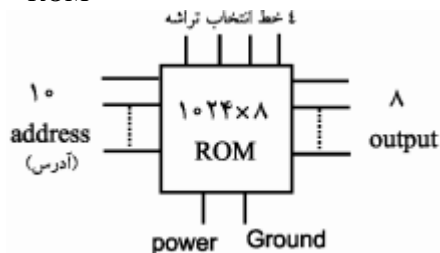
۱۴ خط آدرس خواهیم داشت: $16k = 2^4 \times 2^{10} = 2^{14} \Rightarrow$

که ۱۰ خط آدرس برای آدرس‌دهی تراشه‌ها و ۴ خط آدرس برای انتخاب یکی از ۱۶ تراشه به وسیله یک دیکدر ۴ به ۱۶ استفاده می‌شود.

۳- یک تراشه ROM با 1024×8 بیت دارای چهار ورودی انتخاب بوده و با منبع تغذیه ۵ ولتی کار می‌کند برای بسته IC چند پایه نیاز است. بلاک دیاگرام را رسم کرده و پایانه‌های ورودی و خروجی را در ROM نام‌گذاری کنید.
حل:

۱۰ خط آدرس $\Rightarrow 1024 = 2^{10}$

$\frac{1024 \times 8}{ROM} \Rightarrow$ ۸ خط خروجی



پس داریم:

۴- سیستم حافظه شکل ۴-۱۲ را به ۱۶

گسترش دهید. نقشه حافظه - آدرس را رسم کنید. تعیین کنید که اندازه دیکدر باید چقدر باشد.

حل:

تعداد تراشه RAM:

$$\frac{4096}{128} = 32$$

تعداد تراشه ROM:

$$\frac{4096}{512} = 8$$

از ۱۶ خط آدرس که CPU در اختیار می‌گذارد به ۱۲ خط آدرس نیاز داریم زیرا:

$$4096 = 2^{12}$$

همچنین از بیت ۱۳ خط آدرس برای انتخاب ROM استفاده می‌شود.

پس داریم:

بیت‌های آدرس

تراشه	محدوده آدرس	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷-۱
RAM	0000-0FFF	۰	۰	۰	۰	دیکدر ۵ به ۳۲					×
ROM	1FFF-4000	۰	۰	۰	۱	دیکدر ۳ به ۸			×		

در این بحث یک نکته مهم وجود دارد. در این مثال ما از بیت‌های با ارزش کمتر برای خطوط مشترک آدرس‌دهی و از بیت‌های با ارزش بیشتر برای انتخاب تراشه‌ها استفاده کردیم. به این روش High order Interleaving می‌گویند. حسن این روش در مفهوم تحمل خطا در خرابی احتمالی تراشه‌ها می‌باشد.

روش دیگر که بر عکس روش قبلی، از بیت‌های با ارزش کمتر برای انتخاب تراشه‌ها استفاده می‌شود Low order Interleaving نام دارد. حسن این روش در امکان بکارگیری آدرس‌های درخواستی دسترسی به حافظه‌ها به صورت موازی و در یک زمان است. زیرا آدرس‌های متوالی در تراشه‌های متفاوت قرار دارند.

۵ - کامپیوتری از تراشه‌های RAM با ۲۵۶×۸ و ROM با ۱۰۲۴×۸ استفاده

می‌کند. سیستم کامپیوتر به ۲k بیت از RAM ، ۴k بیت از ROM و چهار واحد واسطه، که هر یک چهار ثبات دارد، نیاز دارد. از پیکربندی I/O نگاشت حافظه استفاده شده است. به دو بیت با ارزش‌تر گذرگاه آدرس ۰۰ برای RAM ۰۱ برای ROM و ۱۰ برای ثبات‌های واسطه اختصاص داده می‌شود.

الف) چند تراشه RAM و ROM لازم است.

(ب) نقشه حافظه - آدرس را برای سیستم رسم کنید.

پ) محدوده آدرس را برحسب شانزده شانزدهی برای RAM و ROM و مدار واسط تعیین کنید.

حل:

تعداد تراشه‌های RAM مورد نیاز:

$$\frac{2048}{256}=8 \quad , \quad 2048=2^{11} \quad , \quad 256=2^8$$

تعداد تراشه‌های ROM مورد نیاز:

$$\frac{4096}{1024}=4, \quad 4096=2^{12}, \quad 1024=2^{10}$$

چهار واحد واسطه که هر کدام ۴ رجیستر دارد.

$$4 \times 4 = 16 = 2^4$$

پس داریم:

		بیت های آدرس																
ترتیب و نام	محدوده آدرس	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
RAM	0000 - 07FF	0	0	0	0	0	← ۸ بیت، Data →			X	X	X	X	X	X	X	X	
ROM	4000 - 4FFF	0	1	0	0	← ۸ بیت، Data →			X	X	X	X	X	X	X	X	X	
Interface	8000 - 800F	1	0	0	0	0	0	0	0	0	0	0	0	← ۸ بیت، Data →			X	X

نوع پورت ۹31

۶- یک کامپیوتر دارای گذرگاه آدرس ۱۶ بیتی است. ۱۵ خط اول آدرس‌ها برای انتخاب یک بانک ۳۲k بایتی حافظه به کار رفته است. بیت با ارزش‌تر آدرس برای انتخاب یک ثبات که محتوای گذرگاه داده را دریافت می‌کند بکار رفته است. توضیح دهید چگونه می‌توان از این پیکربندی استفاده کرده و ظرفیت حافظه را به هشت بانک ۳۲k بایتی افزایش داد تا یک بانک ۲۵۶k بایتی حاصل شود.

حل: $32k = 2^5 \times 2^{10} = 2^{15}$ ۱۵ خط آدرس

پس ۱۵ خط آدرس برای دسترسی به داده‌های هر کدام از بانک‌های ۳۲k بایتی داریم. برای استفاده از بیت با ارزش‌ترین، برای انتخاب ثبات محتوای گذرگاه به صورت زیر عمل می‌کنیم:

شماره بانک ۳۲K بیتی	ثبات محتوای گذرگاه
۱	۰۰۰۰۰۰۰۱
۲	۰۰۰۰۰۰۱۰
۳	۰۰۰۰۰۱۰۰
۴	۰۰۰۰۱۰۰۰
۵	۰۰۰۱۰۰۰۰
۶	۰۰۱۰۰۰۰۰
۷	۰۱۰۰۰۰۰۰
۸	۱۰۰۰۰۰۰۰

۷- یک سیستم دیسک مغناطیسی دارای پارامترهای زیر است:

T_s = زمان متوسط برای قرار گرفتن هد مغناطیسی روی یک شیار

R = سرعت چرخش دیسک برحسب دور بر ثانیه

N_t = تعداد بیت‌ها در شیار

N_s = تعداد بیت‌ها در قطاع

زمان متوسط T_a که برای خواندن یک قطاع نیاز است چقدر است؟

حل: زمان متوسط =

زمان متوسط لازم برای قرار گرفتن هد مغناطیسی روی یک شیار

+ متوسط زمانی که طول می‌کشد دیسک با چرخش خود، هد را در مکان مناسب

جای دهد = نصف یک چرخش کامل

+ زمانی که طول می‌کشد، اطلاعات از سکتور خوانده شود.

پس:

$$T_a = T_s + \frac{1}{2R} + \frac{N_s}{N_t} \times \frac{1}{R}$$

۸ - سرعت جابجایی یک نوار مغناطیسی هشت شیاره که سرعتش ۱۲۰ اینچ در

ثانیه و تراکم آن ۱۶۰۰ بیت در اینچ است چقدر است؟

حل: دقت شود که ۸ بیت مربوط به یک کاراکتر، دقیقاً همزمان خوانده می‌شود (به

دلیل ماهیت نوار مغناطیسی) بنابراین داریم:

کاراکتر در ثانیه $1600 \times 120 = 192000$ = نرخ انتقال

۹- تابع متمم منطق انطباق یک کلمه را در حافظه انجمنی بدست آورید. به بیان

دیگر، نشان دهید که M'_i مجموع توابع OR انحصاری است. دیاگرام منطقی را

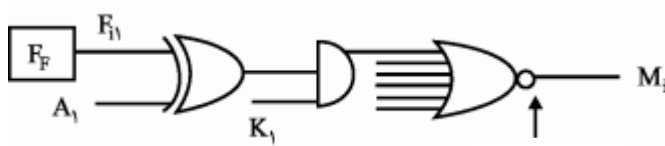
برای M'_i رسم نموده و با یک وارونگر آن را پایان دهید تا M_i بدست آید.

حل: در ابتدا باید گفت مفهوم $\prod_{g=1}^n$ ضرب همه عبارت‌ها به ازای g های ۱ تا n

می‌باشد. همچنین مفهوم $\sum_{g=1}^n$ ، مجموع همه عبارت‌ها به ازای گ‌های از ۱ تا n می‌باشد.
داریم:

$$M_i = \prod_{j=1}^n ((A_j F_{ij} + A'_j F'_{ij}) + K'_j) \\ = \prod_{j=1}^n ((A_j \oplus F_{ij})' + K'_j) \\ M'_i = \sum_{j=1}^n (A_j \oplus F_{ij}) K_j \quad \text{طبق قانون کرفان}$$

برای رسم مدار منطقی، برای یک بیت داریم: (بیت ۱)



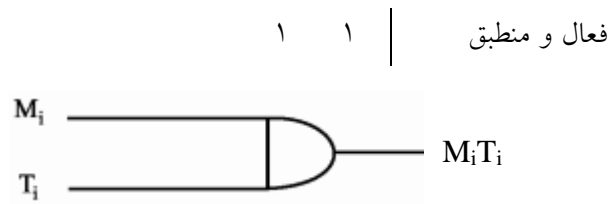
وارونگر

۱۰- تابع بول را برای منطق انطباق یک کلمه با یک حافظه انجمنی با در نظر گرفتن یک بیت نشانه که مشخص می‌کند کلمه فعال است یا غیرفعال را بدست آورید.

حل: این بیت نشانه را به عنوان یکی از پایه‌های یک گیت منطقی AND به کار می‌بریم تا فعال بودن یا نبودن کلمه، مشخص شود. پس:

T_i : بیت نشانه

$M_i T_i$		وضعیت
۰	۰	نه فعال و نه منطبق
۰	۱	فعال و غیر منطبق
۱	۰	منطبق و غیر فعال



۱۱- چه مدار منطقی اضافی نیاز است تا نتیجه عدم انطباق را برای یک کلمه در حافظه انجمی، وقتی که همه بیت‌های کلید صفر است مشخص کند؟

حل: با توجه به مدار داده شده برای حافظه انجمی «تداعیگر»، هر گاه $k_j = \phi$ باشد، مقایسه بیت‌ها تأثیر نخواهد داشت. پس اگر همه بیت‌های key، صفر باشد، تطابق حاصل خواهد شد. برای برآورده کردن خواسته سؤال، صرفاً لازم است حداقل یکی از بیت‌های key، ۱ باشد پس:

$$M_{i(\text{new})} = \left(\prod_{g=1}^n (A_i F_{ig} + A'_g F'_{ig} + K'_g) \right) \cdot (K_1 + K_2 + \dots + K_n)$$

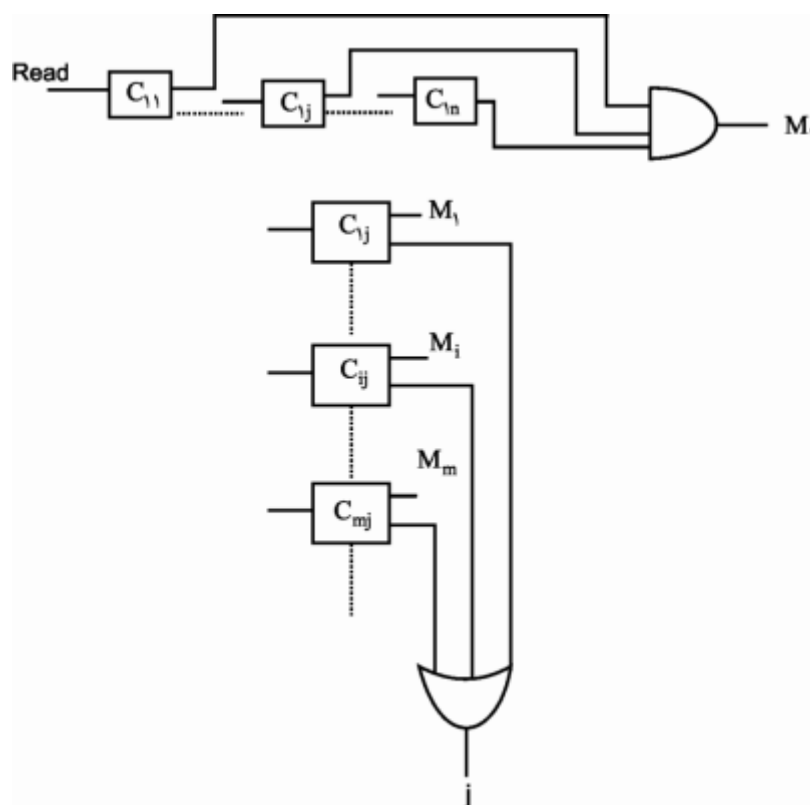
۱۲- الف) دیاگرام منطقی تمام سلول‌های یک کلمه در حافظه انجمی را رسم کنید. منطق خواندن و نوشتن شکل ۸-۱۲ و منطق انطباق شکل ۹-۱۲ را نیز اضافه کنید.

ب) دیاگرام منطقی تمام سلول‌های عمودی یک ستون (ستون J) را در حافظه انجمی رسم کنید. یک خط خروجی مشترک برای همه بیت‌هایی که در یک ستون وجود دارند اضافه کنید.

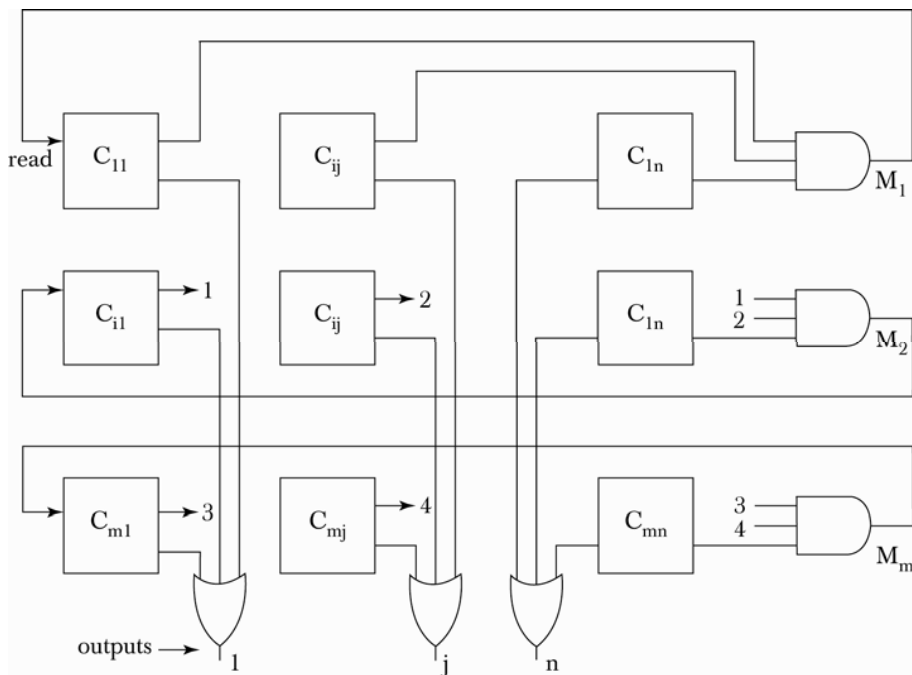
ج) با استفاده از دیاگرام‌های الف و ب نشان دهید که اگر خروجی M_i به خط خواندن همان کلمه وصل شود، کلمه منطبق خوانده خواهد شد، مشروط بر اینکه فقط یک کلمه با آرگومان پوشانده شده مطابقت داشته باشد.

حل:

الف و ب) شکل اول برای کلمه اول، و شکل دوم برای ستون j:



ج) لازم به توضیح است، هر گاه M_i و ۱ شود، پایه Read مربوط به C_i فعال شده و خواندن آن صورت می‌پذیرد. پس:

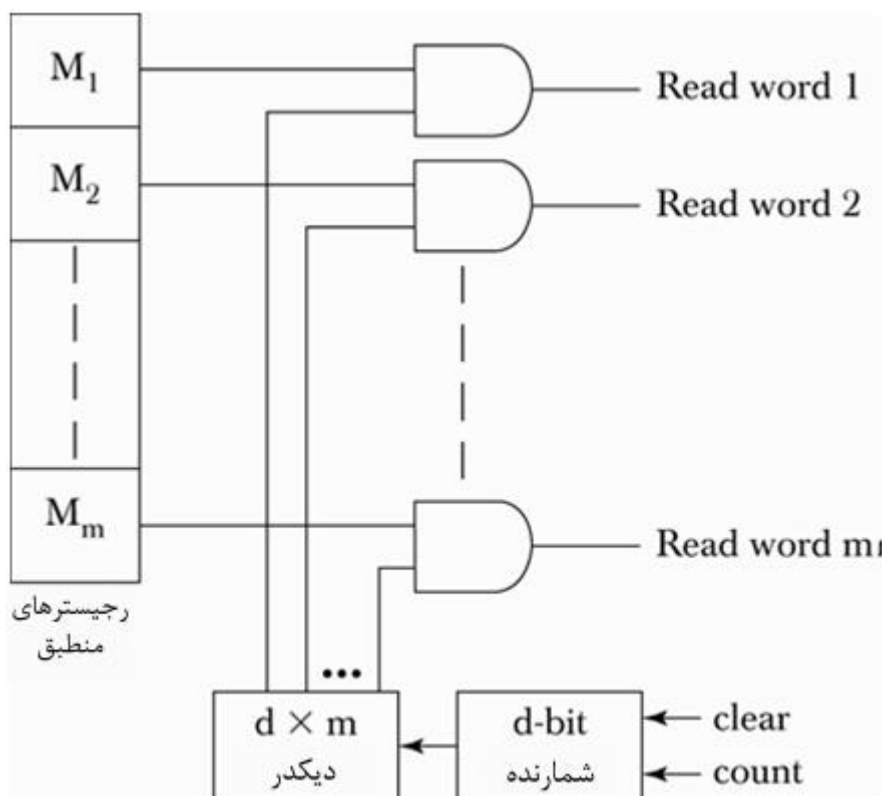


۱۳- با استفاده از بلاک دیاگرام نشان دهید چگونه می‌توان کلمات منطبق متعدد

را از حافظه انجمنی خواند؟

حل: ذکر این نکته مهم است که قاعده‌تاً امکان خواندن تعداد بیش از ۱ کلمه از حافظه انجمنی بطور همزمان و در یک لحظه وجود ندارد. برای این کار باید ترتیبی دهیم تا با هر سیکل، یکی از کلمات منطبق از حافظه خوانده شود. بنابراین از یک شمارنده که بتواند در هر سیکل، یکی از کلمات منطبق را برای خروج انتخاب کند، استفاده می‌کنیم.

پس:



۱۴- منطق یک سلول و نیز یک کلمه کامل را برای یک حافظه انجمنی، همراه با نشانگری برای مواقعی که آرگومان پوشش نیافته بزرگتر از (اما نه مساوی با) کلمه موجود در حافظه انجمنی باشد، بدست آورید.

حل: در رابطه با نشانگر مورد نظر، هدف مقایسه F_i و A ها است. بنابراین

می‌توانیم موارد زیر را داشته باشیم:

$$G_i = 1 \quad \text{IF} \left\{ \begin{array}{l} \text{بیت اول } A, \text{ و } F_{i1} = 0 \text{ است} \end{array} \right.$$

یا

بیت اول A و F_{i1} برابر و بیت دوم A، ۱ و $F_{i2}=0$

یا

بیت اول و دوم A با F_{i1} و F_{i2} برابر و بیت سوم A، ۱ و $F_{i3}=0$

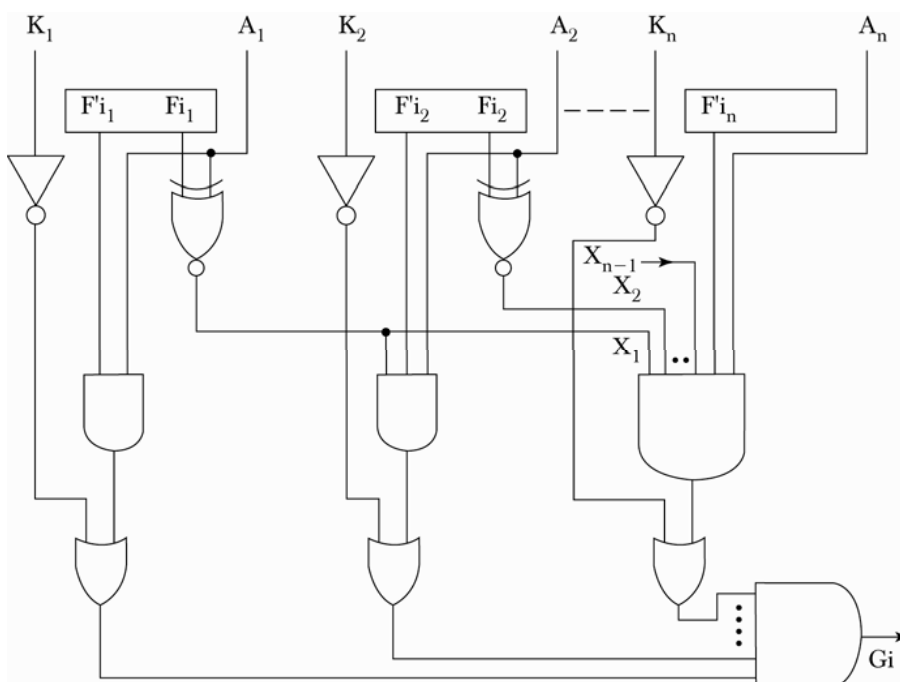
..... }
}

پس داریم:

$$X_j = A_j F_{ij} + A'_j F'_{ij}$$

$$G_i = (A_i F'_{i1} + K'_1)(X_1 A_2 F'_{i2} + K'_2)(X_1 X_2 A_3 F'_{i3} + K'_3) \dots (X_1 X_2 \dots X_{n-1} A_n F'_{in} + K'_n)$$

پس برای مدار داریم:



۱۵- یک حافظه نهان انجمنی دو تایی از بلاک‌های چهار کلمه‌ای استفاده می‌کند. حافظه نهان می‌تواند در مجموع ۲۰۴۸ کلمه از حافظه اصلی را در خود جای دهد. اندازه حافظه اصلی ۱۲۸×۳۲ * است.

الف) همه اطلاعات لازم برای ساختن این حافظه نهان را فرمول بندی کنید.

ب) اندازه حافظه نهان چقدر است؟

حل: حافظه نهان از نوع انجمنی "Associative" و دوتایی "two-way"، و به صورت بلاک‌های ۴ کلمه‌ای است.

* در متن انگلیسی کتاب معماری کامپیوتر، اندازه حافظه اصلی، $۱۲۸k \times ۳۲$ عنوان شده است. پس داریم:

$$128k = 2^{17}$$

الف) تعداد سطرهای حافظه نهان $= 2^{10} \Rightarrow$ در هر سطر حافظه نهان دو کلمه و

$$2048 = 2^{11}$$

پس:

Index	TAG
۱۰ بیت	۷ بیت

ب) اندازه حافظه حافظه نهان برحسب تعداد بیت مدنظر است. یک سطر حافظه

نهان را رسم می‌کنیم.

Tag1	Data2	Tag2	Data 2
۷ بیت	۳۲ بیت	۷ بیت	۳۲ بیت

در هر سطر حافظه نهان، $۲(۷+۳۲)$ بیت داده وجود دارد. پس اندازه حافظه نهان

برحسب تعداد بیت:

$$1024 \times 2(7 + 32) = 1024 \times 78 = 79872 \text{ بیت}$$

۱۶- زمان دستیابی یک حافظه نهان $۱۰۰ns$ و حافظه اصلی $۱۰۰۰ns$ است.

پیش‌بینی می‌شود ۸۰ درصد درخواستهای حافظه برای خواندن و ۲۰ درصد بقیه برای نوشتن می‌باشند. نسبت برد برای دستیابی‌های خواندن فقط ۰/۹ است. رویه کامل‌نویسی استفاده می‌شود.

الف) زمان دستیابی متوسط سیستم فقط با در نظر گرفتن سیکل خواندن چقدر است؟

ب) زمان دستیابی متوسط سیستم برای درخواست‌های خواندن و نوشتن چقدر است؟

ج) نسبت برد با در نظر گرفتن سیکل‌های نوشتن چقدر است؟

حل: منظور از نسبت برد، Hit Ratio است.

در حالت کلی برای زمان دستیابی متوسط سیستم داریم:

$$t_a = ht_c + (1-h)(t_c + t_m)$$

t_c = زمان دستیابی حافظه نهان

t_m = زمان دستیابی حافظه اصلی

h = احتمال وجود داده در حافظه نهان

حال داریم:

الف)

$$t_a = ht_c + (1-h)(t_c + t_m) = 0,9 \times 100 + 0,1 \times 1100 = 200 \text{ ns}$$

ب) از قسمت الف، برای خواندن، زمان متوسط دستیابی را 200 ns بدست آوریم.

حال داریم: (خواندن 200ns و نوشتن 1000ns)

$$t_a = 0,2 \times 1000 + 0,8 \times 200 = 360 \text{ ns}$$

ج) از ۸۰ درصد دستورات که خواندن هستند، ۰/۹ آنها در حافظه نهان هستند پس:

$$h = 0,8 \times 0,9 = 0,72$$

۱۷- یک مجموعه حافظه نهان انجمنی چهار تایی در هر مجموعه چهار کلمه

دارد. یک رویه جایگزینی بر پایه الگوریتم قدیمی‌ترین مورد استفاده (LRU) با

شمارنده‌های دو بیتی متناظر با هر یک از کلمه‌های مجموعه پیاده‌سازی می‌شود. بنابراین مقداری در محدوده ۰ تا ۳ برای هر کلمه ثبت می‌شود. وقتی بر دی رخ دهد (Hit)، شمارنده متناظر با کلمه دسترسی ۰ می‌گردد. شمارنده‌هایی که مقدار قبلی آنها کمتر از مقدار شمارنده مورد دسترسی است، ۱ واحد افزایش می‌یابند و بقیه تغییر نمی‌کنند. اگر باخت رخ دهد (Miss)، کلمه‌ای که مقدار شمارنده آن ۳ است حذف می‌شود، کلمه جدید در محل آن قرار داده می‌شود و شمارنده آن ۰ می‌گردد. سه شمارنده دیگر ۱ واحد افزایش می‌یابند. برای دنباله دسترسی‌های زیر روند را انجام دهید.

A B C D B E D A C E C E →

حل:

داریم:

مقادیر شمارنده	۰	۱	۲	۳	ورود
ABCD ورود	D	C	B	A	
Hit B	B	D	C	A	
Miss E	E	B	D	C	
Hit D	D	E	B	C	
Miss A	A	D	E	B	
Miss C	C	A	D	E	
Hit E	E	C	A	D	
Hit C	C	E	A	D	
Hit E	E	C	A	D	

۱۸- یک کامپیوتر دیجیتال دارای واحد حافظه $16 \times 64k$ و یک حافظه نهان $1k$ کلمه‌ای است. حافظه نهان از نگاشت مستقیم (Direct) استفاده می‌کند و سایز بلاک

چهار کلمه است.

الف) در میدان‌های نشانه، شاخص، بلاک، و کلمه قالب آدرس چند بیت وجود دارد؟

ب) در هر کلمه حافظه نهان چند بیت وجود دارد، و آنها چگونه به توابع مختلف کاری تقسیم شده‌اند؟

ج) حافظه نهان چند بلاک را در خود جای می‌دهد.
حل:

حافظه اصلی ۱۶ خط آدرس و ۱۶ خط داده $\Rightarrow 2^{16} = 2^{10} \times 2^6 = 64k$ حافظه

الف) دقت شود برای آدرس، ما TAG و Index و تعداد کلمه‌های در یک بلاک را در نظر می‌گیریم.

$10 = \text{تعداد بیت‌های آدرس‌دهی حافظه نهان} \Rightarrow 1k = 2^{10}$

تعداد بیت‌های TAG $16 - 10 = 6 \Rightarrow$

$$16 = 2 \text{ بیت} + 8 \text{ بیت} + 6 \text{ بیت}$$

>> سایز بلاک چهار کلمه

WORD	Block	TAG
------	-------	-----

ب) دقت شود برای کلمه حافظه نهان، TAG L، داده و بیت VALID را در نظر می‌گیریم.

$$23 = 1 + 6 + 18 \text{ بیت}$$

۱ بیت	۶ بیت	۱۶ بیت
V	TAG	DATA

ج)

$1k = 1024 = \text{حافظه نهان}$

بلاک $256 = \frac{1024}{4} = \text{تعداد بلاک} \Rightarrow$ در هر بلاک ۴ کلمه

۱۹- یک فضای آدرس با ۲۴ بیت و فضای حافظه متناظر با آن با ۱۶ بیت

مشخص می‌شود.

الف) چند کلمه در فضای آدرس وجود دارد؟

ب) چند کلمه در فضای حافظه وجود دارد؟

ج) اگر یک صفحه از ۲k کلمه ساخته شده باشد، چند صفحه و بلاک در فضای حافظه وجود دارد؟

حل: در بحث حافظه مجازی سیستم‌های کامپیوتری، منظور از فضای حافظه متناظر، فضای آدرس‌دهی حافظه فیزیکی است. یعنی حافظه اصلی یا RAM منظور از فضای آدرس، حافظه مجازی یا منطقی، و به عبارت دیگر اندازه برنامه (process) در کامپیوتر می‌باشد. بنابراین داریم:

الف) فضای آدرس: ۲۴ بیت \Rightarrow

$$2^{24} = 16M \quad \text{کلمه}$$

ب) فضای حافظه: ۱۶ بیت \Rightarrow

$$2^{16} = 64K \quad \text{کلمه}$$

ج) طبق تعاریف بالا:

$$\text{تعداد صفحه} = \frac{16M}{2K} = 8K \text{ pages}$$

$$\text{قاب} = \frac{64K}{2K} = 32 \quad \text{تعداد بلاک یا قاب}$$

۲۰- یک حافظه مجازی دارای یک صفحه ۱K کلمه‌ای است. در این حافظه ۸

صفحه و چهار بلاک وجود دارد. جدول صفحه‌های حافظه انجمنی حاوی داده‌های زیر است.

لیستی از آدرس‌های مجازی (به دهدهی) تهیه کنید که در صورت استفاده CPU از آن موجب فقدان صفحه (Page Fault) شوند.

صفحه	بلاک
۰	۳
۱	۱
۴	۲
۶	۰

حل: سایز صفحات، ۱K می‌باشد؛ بنابراین سایر قاب‌های حافظه اصلی نیز ۱K می‌باشد. با توجه به جدول داده شده می‌توانیم به طور دقیق مشخص کنیم که صفحاتی که الان نام برده شده‌اند، چه آدرس‌هایی را پوشش می‌دهند:

صفحه	بلاک	محدوده آدرس پوشاننده (دهدهی)
۰	۳	۰-۱۰۲۳
۱	۱	۱۰۲۴-۲۰۴۷
۴	۲	۴۰۹۶-۵۱۱۹
۶	۰	۶۱۴۴-۷۱۶۷

یعنی آدرس‌های مجازی که CPU اعلان می‌دهد، اگر در محدوده‌های بالا باشد، صفحات متناظر وجود خواهند داشت. بنابراین می‌توان گفت آدرس‌های مربوط به سایر صفحات که در جدول صفحه نیستند موجب بروز فقدان صفحه می‌شوند:

صفحه	محدوده آدرس مربوطه که موجب فقدان صفحه می‌شوند.
۲	۲۰۴۸-۳۰۷۱
۳	۳۰۷۲-۴۰۹۵
۵	۵۱۲۰-۶۱۴۳
۷	۷۱۶۸-۸۱۹۱

۲۱- یک سیستم حافظه مجازی دارای فضای آدرس ۸K کلمه یک فضای حافظه

۴K کلمه، و صفحات و بلاک‌های ۱K کلمه است (شکل ۱۸-۱۲) تغییرات دسترسی صفحه زیر در طول یک فاصله زمانی معین رخ می‌دهد. (فقط تغییر صفحه‌ها نوشته شده است. اگر یک صفحه دوباره مورد دسترسی قرار گرفته باشد دو بار نوشته نشده است.)

4 2 0 1 2 6 1 4 0 1 0 2 3 5 7 →

چهار صفحه مستقیم در حافظه اصلی را پس از هر تغییر دسترسی صفحات تعیین کنید به شرطی که الگوریتم جایگزینی مورد استفاده (الف) FIFO و (ب) LRU باشد.

حل: برای هر کدام از الگوریتم‌ها، جدول صفحه را رسم می‌نماییم:

منظور از H، Hit و منظور از M، Miss صفحات است

(الف) FIFO (First In First Out)

مراجعه	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
	۴	۲	۰	۱	۱	۶	۶	۴	۴	۴	۴	۲	۳	۵	۷
	-	۴	۲	۰	۰	۱	۱	۶	۶	۶	۶	۴	۲	۳	۵
	-	-	۴	۲	۲	۰	۰	۱	۱	۱	۱	۶	۴	۲	۳
	-	-	-	۴	۴	۲	۲	۰	۰	۰	۰	۱	۶	۴	۲
Page Fault	M	M	M	M	H	M	H	M	H	H	H	M	M	M	M

(ب) LRU (Least Recently Used)

مراجعه	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
	۴	۲	۰	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۵	۵
	-	۴	۲	۰	۰	۰	۰	۴	۴	۴	۴	۴	۳	۳	۳

	۷	۰	۰	۰	۰	۰	۰	۲	۲	۲	۲	۲	۲	۲	۲	۴	-	-
	۲	۲	۲	۲	۶	۶	۶	۶	۶	۶	۶	۶	۶	۶	۶	۴	-	-
Page Fault	M	M	M	M	H	M	H	M	H	H	H	M	M	M	M	M	M	M

۲۲- دو آدرس منطقی را از شکل ۱۲-۲۴ (الف) که سبب دستیابی به حافظه

فیزیکی در آدرس فیزیکی شانزده شانزدهمی ۰۱۲AF می شود تعیین کنید.

حل: با توجه به قسمت الف شکل ۱۲-۲۴، محدوده آدرس (01200 – 012FF) شامل

بلاک ۱۲ حافظه فیزیکی می باشد. حال مطابق با جدول صفحه، صفحات ۳۵ و A3 به

این بلاک اشاره می کنند. مطابق با جدول قطعات، قطعات 6 و F به شامل این دو صفحه

می شوند. بنابراین می توان دو آدرس منطقی زیر را نام برد:

600 AF

F00AF

۲۳- فضای آدرس منطقی در یک سیستم کامپیوتری دارای ۱۲۸ قطعه است. هر

قطعه می تواند تا ۳۲ صفحه ۴K کلمه ای داشته باشد. حافظه فیزیکی ۴K بلاک ۴K

کلمه در هر بلاک است. قالب آدرس های فیزیکی و منطقی را تنظیم کنید.

حل: قالب منطقی

۷ بیت $\Rightarrow 2^7 = 128$ قطعه

۵ بیت $\Rightarrow 2^5 = 32$ صفحه

۱۲ بیت $\Rightarrow 2^{12} = 2^2 \times 2^{10} = 4K$ کلمه

پس :

۷ بیت	۵ بیت	۱۲ بیت
Segment	Page	Word

قالب فیزیکی:

۱۲ بیت $4K = 2^2 \times 2^{10} = 2^{12}$ بلاک

۱۲ بیت $\Rightarrow 4K = 2^{12}$ کلمه

پس:

۱۲ بیت	۱۲ بیت
Block	Word

۲۴- معادل دودویی آدرس منطقی تنظیم شده در مسئله ۲۳-۱۲ را برای قطعه ۳۶

و کلمه ۲۰۰۰ در صفحه ۱۵ بنویسید.

حل: با توجه به سؤال ۲۳-۱۲، آدرس منطقی، ۲۴ بیت می‌باشد. قطعه ۷ بیت، صفحه

۵ بیت و کلمه ۱۲ بیت را اختصاص می‌دهد.

قطعه ۳۶ $\Leftarrow (0100100)_2$

صفحه ۱۵ $\Leftarrow (01111)_2$

کلمه ۲۰۰۰ $\Leftarrow (011\ 111010000)_2$

پس: آدرس منطقی برابر است با:

(0100100 01111 01111010000)

۲۴ بیت

تست‌های فصل ۴ «انتقال ثبات‌ها و ریز عمل‌ها»

- ۱- به منظور ساخت یک گذرگاه مشترک، که تعداد ۳۲ ثبات (Register) ۱۶ بیتی را به همدیگر وصل بنماید، حداقل سخت‌افزار لازم چیست؟
(مهندسی کامپیوتر - سراسری ۸۰)

الف) ۱۶ عدد MUX هر کدام با ۴ خط انتخاب

ب) ۳۲ عدد MUX هر کدام با ۴ خط انتخاب

ج) ۱۶ عدد MUX هر کدام با ۵ خط انتخاب

د) ۳۲ عدد MUX هر کدام با ۵ خط انتخاب

- ۲- کدام یک از انتقال بین رجیسترها غلط است؟

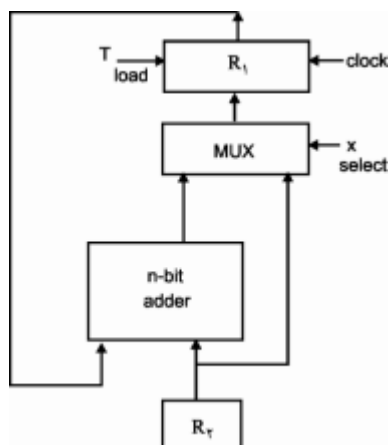
(آزاد - ۸۱)

الف) $XT: AR \leftarrow \overline{AR}, AR \leftarrow 0$ ب) $YT: R_1 \leftarrow R_2, R_2 \leftarrow R_1$

ج) $ZT: PC \leftarrow AR, AR \leftarrow AR - 1$ د) $NT: PC \leftarrow PC + 1$

- ۳- در شکل زیر کدام یک از موارد انتقال رجیستری (RTL) قابل اجرا است؟

(مهندسی کامپیوتر - سراسری ۸۳)



الف) $X: R_1 \leftarrow R_2$

ب) $XT: R_1 \leftarrow R_2$

ج) $\overline{XT}: R_1 \leftarrow R_2$

د) $T\overline{X}: R_1 \leftarrow R_1 + R_2$

۴- کدام عبارت صحیح است؟

(مهندسی IT - سراسری ۸۷)

الف) برای پیاده‌سازی گذرگاه مشترک با وجود ۸ ثبات ۱۶ بیتی، جهت پیاده‌سازی با روش بافر سه حالت، به ۸ بافر ۱۶ بیتی و یک رمزگشای 3×8 نیاز است و جهت پیاده‌سازی با MUX به ۱۶ MUX 8×1 نیاز است.

ب) برای پیاده‌سازی گذرگاه مشترک با وجود ۸ ثبات ۱۶ بیتی، جهت پیاده‌سازی با روش بافر سه حالت به ۸ بافر ۱۶ بیتی و یک رمزگشای 4×16 نیاز است و جهت پیاده‌سازی با MUX به ۱۶ MUX 8×1 نیاز است.

ج) برای پیاده‌سازی گذرگاه مشترک با وجود ۸ ثبات ۱۶ بیتی، جهت پیاده‌سازی با روش بافر سه حالت به ۱۶ بافر ۸ بیتی و یک رمزگشای 4×16 نیاز است و جهت پیاده‌سازی با MUX، به ۱۶ MUX 8×1 نیاز است.

د) برای پیاده‌سازی گذرگاه مشترک با وجود ۸ ثبات ۱۶ بیتی، جهت پیاده‌سازی با روش بافر سه حالت به ۱۶ بافر ۸ بیتی و یک رمزگشای 3×8 نیاز است و جهت پیاده‌سازی با MUX، به ۱۶ MUX 16×1 نیاز است.

حل تست‌های فصل ۴

۱- گزینه‌ی ج صحیح است.

در حالت کلی اگر K رجیستر n بیتی داشته باشیم، برای ساخت گذرگاه مشترک به تعداد n Mux که هر کدام $k \times 1$ هستند نیاز داریم. در ضمن تعداد خطوط انتخاب MUX ها، $\lceil \log_2^n \rceil$ خواهد بود.

۲- گزینه‌ی الف صحیح است.

امکان ندارد در یک زمان، متمم یک رجیستر و مقدار صفر را در همان رجیستر ذخیره کرد. در اینگونه موارد تداخل (conflict) پیش می‌آید. مانند

$$XT : PC \leftarrow AR, PC \leftarrow PC + 1$$

لازم به ذکر است امکان جابجایی محتوای دو رجیستر در یک زمان وجود دارد.

۳- گزینه‌ی ج صحیح است.

با توجه به مدار اگر x ، $+$ باشد، R_2 و اگر 1 باشد، $R_1 + R_2$ ، آماده انتقال است. در ضمن در صورتی انتقال انجام می‌شود که پایه load (T) فعال باشد.

۴- گزینه‌ی ج صحیح است.

در حالت کلی اگر k رجیستر n بیتی داشته باشیم برای ساخت گذرگاه مشترک با استفاده از روش بافر، به یک رمزگشای $n \times (\log_2^n)$ و همچنین $m \times n$ بافر سه حالته نیاز داریم.

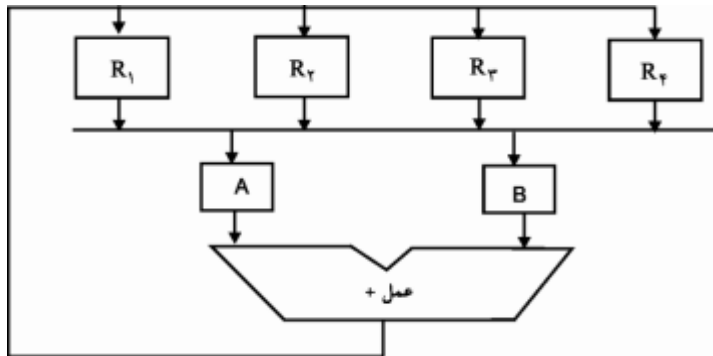
تست‌های فصل ۵ «سازمان و طراحی یک کامپیوتر

پایه»

۱- در شکل داده شده جهت انجام عملیات $R_1 \leftarrow R_4 + R_3, R_1 \leftarrow R_1 + R_2$ به ترتیب چند پالس ساعت لازم است؟

(مهندسی کامپیوتر - سراسری ۸۱)

(مشابه مهندسی کامپیوتر - سراسری ۷۹)



الف) ۲ و ۳

ب) ۳ و ۳

ج) ۳ و ۴

د) ۳ و ۴

۲- در طراحی قالب دستورالعمل‌های یک کامپیوتر، ۴ بیت برای رمز عمل (opcode)، ۲ بیت برای ثبات و دو فیلد ۱۵ بیتی برای آدرس وجود دارد. با توجه به قالب دستورالعمل، مشخصات این کامپیوتر چیست؟

(آزاد - ۸۲)

الف) ۲۵۶ دستورالعمل مختلف، ۲ ثبات قابل استفاده توسط کاربر، فضای آدرس

حافظه 32KB

ب) ۱۶ دستورالعمل مختلف، ۱۲۸ ثبات قابل استفاده توسط کاربر، فضای آدرس

حافظه 4KB

ج) ۱۶ دستورالعمل مختلف، ۶ ثبات قابل استفاده توسط کاربر، و فضای آدرس حافظه 2KB

د) هیچکدام

۳- با فرض اینکه در طی فازهای T_0 و T_1 عمل واکنشی (Fetch) و دیکد دستورالعمل به صورت مقابل انجام شده است؟

$T_0 : DR \leftarrow M[PC]$

$T_1 : IR \leftarrow DR, PC \leftarrow PC + 1, \text{Decode}$

مشخص کنید دستور STA Addr کدام مجموعه‌ی گزاره‌های زیر است؟

(مهندسی IT – سراسری ۸۴)

الف) $T_2 : AR \leftarrow DR, PC \leftarrow PC + 1$ ب) $T_2 : DR \leftarrow M[PC], DR \leftarrow AC$

$T_3 : M[AR] \leftarrow DR, \text{goto } T_0$ $T_3 : M[AR] \leftarrow AC, \text{goto } T_0$

$T_2 : DR \leftarrow M[PC]$

$T_2 : DR \leftarrow M[PC]$

د) $T_3 : AC \leftarrow DR, PC \leftarrow PC + 1$

ج) $T_3 : AR \leftarrow DR, PC \leftarrow PC + 1$

$T_4 : M[AR] \leftarrow AC, \text{goto } T_0$

$T_4 : M[AR] \leftarrow AC, \text{goto } T_0$

۴- کدام دستورالعمل غیرکنترلی است؟

(آزاد – ۸۴)

الف) CALL ب) SKIP ج) MORE د) JUMP

۵- برای قالب دستورالعمل

آدرس	کد عمل	I, کد عمل ۰۰۰۱
------	--------	----------------

 با شرح

AC < 0 then
کدام نماد تداعی می‌شود؟ (EA آدرس موثر است)
(PC ← EA)

(آزاد – ۸۳)

الف) ADD ب) STORE ج) EXCHANGE د) BRANCH

حل تست‌های فصل ۵

۱- گزینه‌ی ب صحیح است.

در هر دو عملیات، تعداد یکسانی پالس ساعت نیاز است. در پالس اول، مقدار رجیستر اول آماده عملیات می‌شود، در پالس دوم، مقدار رجیستر دوم آماده عملیات می‌شود، و در پالس سوم عمل جمع انجام شده و نتیجه بازنویسی می‌شود.

۲- گزینه‌ی د صحیح است.

با توجه به توضیحات، قابل دستورالعمل به صورت زیر است.

۴	۲	۱۵	۱۵
opcode	ثبات	آدرس ۱	آدرس ۲

پس داریم:

$$2^2 = 4 = \text{تعداد ثبات برای کار}$$

$$2^4 = 16 = \text{تعداد دستورالعمل مختلف}$$

$$2^{15} = 32\text{KB} = \text{فضای آدرس‌دهی حافظه}$$

که هیچکدام از گزینه‌های ۱، ۲ و ۳ را شامل نمی‌شود.

۳- گزینه‌ی ج صحیح است.

دستور STA، محتوای انباره (AC) را در حافظه ذخیره می‌کند. در این سؤال، دستورات دو کلمه‌ای هستند، و کلمه دوم، آدرس را مشخص می‌کند. بنابراین، ابتدا فیلد آدرس از حافظه خوانده شده و پس محتوای انباره، در آن آدرس از حافظه ذخیره می‌شود.

۴- گزینه‌ی ج صحیح است.

هر سه دستور CALL, JUMP, SKIP، مسیر عادی اجرای برنامه را عوض می‌کنند.

CALL: رفتن به آدرس شروع زیر برنامه

JUMP: پرش به آدرس مورد نظر از حافظه

SKIP: پرش از دستور بعدی در صورت پذیرفته شده شرط

۵- گزینه‌ی د صحیح است.

با توجه به عبارت $if (AC < 0) then PC \leftarrow EA$ ، دستور یک پرش شرطی خواهد

بود.

تست‌های فصل ۶ «برنامه‌نویسی کامپیوتر پایه»

۱- مجموعه سه دستورالعمل زیر در پردازنده ۸۰۸۶ چه می‌کند؟

XOR AX, BX

XOR BX, AX

XOR AX, BX

(آزاد - ۷۸)

الف) محتوای ثبات AX را با AX+BX تبدیل می‌کند.

ب) محتوای ثبات AX را با BX عوض می‌کند.

ج) محتوای ثبات BX را با BX+AX تبدیل می‌کند.

د) در نهایت هیچ تغییری در محتوای دو ثبات داده نمی‌شود.

۲- مجموعه دستورات زیر چه عمل را انجام می‌دهند؟

1- MOV bx, ax

2- SHR ax

3- SHR ax

4- SHR ax

5- AND bx, 07

(علوم کامپیوتر - ۸۲)

الف) محتوای رجیستر ax را بر ۸ تقسیم کرده و خارج قسمت را در رجیستر bx و باقی‌مانده را در رجیستر ax قرار می‌دهد.

ب) محتویات رجیستر ax را بر ۸ تقسیم می‌کند و خارج قسمت را در رجیستر ax قرار داده و تمام بیت‌های خارج قسمت به جز سه بیت اول را صفر می‌کند.

ج) محتویات رجیستر ax را بر ۸ تقسیم می‌کند و فقط خارج قسمت را در رجیستر ax قرار می‌دهد و باقیمانده را محاسبه نمی‌کند.

د) محتویات رجیستر ax را بر ۸ تقسیم کرده و خارج قسمت را در رجیستر ax و

باقیمانده را در رجیستر bx قرار می‌دهد.

حل تست‌های فصل ۶

۱- گزینه‌ی ب صحیح است.

از طریق جبر بول می‌توان اثبات کرد که نتیجه اجرای برنامه، تعویض محتوای دو رجیستر AX و BX است برای مثال داریم، اگر AX و BX ۴ بیتی باشند:

$$AX = 1001$$

$$BX = 0011$$

$$XOR \ AX, BX \Rightarrow AX = 1010, BX = 0011$$

$$XOR \ BX, AX \Rightarrow AX = 1010, BX = 1001$$

$$XOR \ AX, BX \Rightarrow AX = 0011, BX = 1001$$

۲- گزینه‌ی د صحیح است.

می‌دانیم در نتیجه یک شیفت به چپ، ضرب در ۲، و در نتیجه یک شیفت به راست، تقسیم بر ۲ حاصل می‌شود. بنابراین ax بر ۸ تقسیم می‌شود. در ضمن bx, 07 and ، باعث صفر شدن ۵ بیت با ارزش بیشتر محتوای رجیستر bx ، که همان ax اولیه است، می‌شود و ۳ بیت کم ارزش ax که بیانگر باقی مانده تقسیم بر ۸ است را نگه می‌دارد.

تست‌های فصل ۷ «کنترل ریز برنامه‌نویسی شده»

۱- دلیل استفاده از ریز برنامه‌سازی (Microprogramming) در طراحی واحد

کنترل CPU چیست؟

(مهندسی کامپیوتر - سراسری ۷۱)

الف) طراحی واحد کنترل سیستماتیک شود.

ب) بتوان به آسانی خطاهای طراحی را برطرف نمود و فهرست دستورهای زبان ماشین را تغییر داد.

ج) بتوان برنامه نوشته شده به زبان ماشین کامپیوتر دیگری را اجرا نمود.

د) هر سه مورد

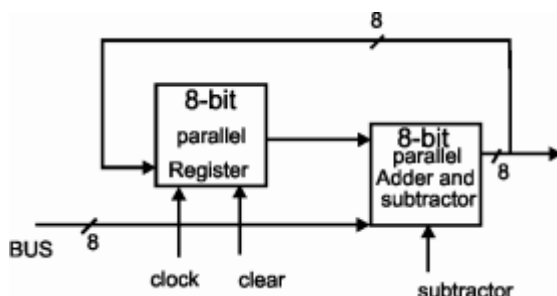
۲- نمودار بلوکی مدار و سیگنال‌های کنترل لازم برای انجام عملیات جمع و

تفریق شکل زیر نشان داده شده است. در صورتی که در طراحی واحد کنترل آن از

روش ریز برنامه‌سازی استفاده کنیم. تعداد ریز دستورالعمل‌های (Microinstruction)

لازم برای انجام عمل A-B را به دست آورید.

(مهندسی کامپیوتر - سراسری ۷۳)



الف) یک ریز دستورالعمل

ب) دو ریز دستورالعمل

ج) سه ریز دستورالعمل

د) چهار ریز دستورالعمل

۳- کدام یک از عبارات زیر در مورد ساختارهای ریز برنامه‌ریزی شده

(Microprogrammed) و سیم‌کشی (Hard wired) درست است؟

(مهندسی کامپیوتر – سراسری ۷۸)

الف) پردازنده‌های از نوع “Microprogrammed” قابلیت انعطاف بیشتری از لحاظ تغییرات احتمالی در آینده دارند و معمولاً دستورالعمل‌های پیچیده‌تری و سرعت کمتری نسبت به نوع “Hard wired” دارند.

ب) پردازنده‌های از نوع “Microprogrammed” نمی‌توانند از امکان pipeline استفاده کنند. در صورتی که نوع “Hard wired” این امکان را دارد و علت سرعت بیشتر آن نیز همین است.

ج) پردازنده‌های از نوع “Microprogrammed” سرعت بیشتری از دید اجرای دستورالعمل نسبت به نوع “Hard wired” دارند.

د) پردازنده‌های از نوع “Hard wired” دارای سرعت بیشتری نسبت به نوع “Microprogrammed” هستند که علت عمده آن استفاده از “pipelining” در ساختار اینگونه پردازنده‌ها است.

۴- در مورد ریز برنامه‌سازی “Microprogramming” کدام یک از جمله‌های زیر صحیح است؟

(آزاد – ۷۸)**(مشابه مهندسی IT – سراسری ۸۴)**

الف) ریز برنامه‌سازی برای سیستماتیک کردن طراحی واحد کنترل به کار می‌رود و در طراحی پردازنده‌های با کارایی بالای امروزی کاربرد زیاد دارد.

ب) ریز برنامه‌سازی برای مهیاسازی زمینه تغییر و اضافه کردن دستورات زبان ماشین مناسب است ولی به دلیل مشکل بودن، در طراحی کامپیوترهای امروزی به کار نمی‌رود.

ج) ریز برنامه‌سازی مزایای فراوانی دارد، ولی به دلیل سرعت کم در طراحی پردازنده‌های سریع امروزی به کار نمی‌رود.

د) ریز برنامه‌سازی این امکان را در اختیار می‌گذارد که بتوان برنامه نوشته شده به زبان ماشین کامپیوتر دیگری را اجرا نمود و به همین دلیل در طراحی واحد کنترل پردازنده‌های امروزی به کار گرفته می‌شود.

حل تست‌های فصل ۷

۱- گزینه‌ی د صحیح است.

کنترل دنباله عملیات به وسیله ریز دستورالعمل‌ها، مستقل از زبان ماشین کامپیوترهای مختلف است و دارای قابلیت تغییر، توسعه و رفع خطا است.

۲- گزینه‌ی ج صحیح است.

ابتدا رجیستر صفر شده، پس با A جمع شده، و پس از B کم می‌شود تا A-B انجام شود.

1) $R \leftarrow 0$

2) $R \leftarrow R + A$

3) $R \leftarrow R - B$

۳- گزینه‌ی الف صحیح است.

در حالت کلی در ارتباط با مقایسه پردازنده‌های ریز برنامه‌نویسی شده و سیم‌بندی داریم:

واحد کنترل ریز برنامه‌سازی شده	واحد کنترل سیم‌بندی	معیارها
کم	زیاد	سرعت اجرا
کم	زیاد	پیچیدگی سخت‌افزار
زیاد	کم	سهولت تغییر و توسعه
کم	زیاد	میزان استفاده کنونی
بله	بله	امکان استفاده pipeline در آنها

۴- گزینه‌ی ج صحیح است.

با توجه به توضیحات سؤال قبل

تست‌های فصل ۸ «واحد مرکزی پردازش»

۱- برای محاسبه تابع $(8+2 \times 5)/(1+3 \times 2-4)$ با استفاده از stack، چه تعداد دستورالعمل push و حداقل تعداد کلمه لازم در stack برای انجام این عمل چقدر است؟

(مهندسی کامپیوتر - سراسری ۷۴)

الف) ۷ مرتبه push و حداقل ۵ کلمه ب) ۶ مرتبه push و حداقل ۳ کلمه

ج) ۸ مرتبه push و حداقل ۴ کلمه د) ۷ مرتبه push و حداقل ۴ کلمه

۲- با فرض مقادیر خانه‌های حافظه زیر و داشتن یک پردازنده دارای یک انباشتگر (Accumulator) و میدان تک آدرس در دستورالعمل، انباشتگر پس از اجرای هر یک از دستورات زیر چه مقداری خواهد داشت؟

(مهندسی کامپیوتر - سراسری ۷۵)

۲۰ کلمه ۲۰ حافظه مقدار ۴۰ دارد. a: load immediate ۲۰

۳۰ کلمه ۳۰ حافظه مقدار ۵۰ دارد. b: load indirect ۲۰

۴۰ کلمه ۴۰ حافظه مقدار ۶۰ دارد. c: load direct ۳۰

۵۰ کلمه ۵۰ حافظه مقدار ۷۰ دارد. d: load indirect ۳۰

الف) ۲۰ و ۳۰ و ۶۰ و ۷۰ ب) ۴۰ و ۵۰ و ۶۰ و ۷۰

ج) ۲۰ و ۴۰ و ۳۰ و ۵۰ د) ۲۰ و ۶۰ و ۵۰ و ۷۰

۳- با فرض داشتن یک ماشین پشته‌ای (stack machine) مشخص کنید دستورات زیر کدام عبارت را اجرا می‌کند؟

(مهندسی کامپیوتر - سراسری ۷۶)

الف) $X = (A * B / D - C) + (C + A - B * F / E)$

ب) $X = (A * B - C) / D + (C / E + A - B * F)$

ج) $X = (A / D * B - C) + (A - B * F * C / E)$

$$X = (A * B / C - D + A - B * F * C / E) \quad (د)$$

Push A
Push B
Mul
Push C
SUB
Push D
DIV
Push C
Push E
DIV
Push A
ADD
Push B
Push F
MUL
SUB
ADD
Popx

۴- کدام مجموعه از عملیات ذیل می‌تواند یک پشته (stack) را پیاده‌سازی

نماید؟ عملیات push و pop برای یک ثبات به نام A صورت می‌پذیرد؟

(مهندسی کامپیوتر - سراسری ۷۹)

Push A	Pop A	
$SP \leftarrow SP - 1$ $Mem[SP] \leftarrow A$	$A \leftarrow Mem[SP]$ $SP \leftarrow SP + 1$	(۱)
$SP \leftarrow SP + 1$ $Mem[SP] \leftarrow A$	$A \leftarrow Mem[SP]$ $SP \leftarrow SP - 1$	(۲)
$Mem[SP] \leftarrow A$ $SP \leftarrow SP - 1$	$SP \leftarrow SP + 1$ $A \leftarrow Mem[SP]$	(۳)

الف) ۱ و ۲ ب) ۲ و ۳ ج) ۱ و ۳ د) هر سه

۵- تفاوت کامپیوترهای RISC و CISC کدام است؟ CPI یعنی تعداد متوسط

پالس ساعت به ازاء اجرای هر دستورالعمل)

(مهندسی کامپیوتر - سراسری ۸۰)

الف) در RISC تعداد دستورات کمتر، طول برنامه بزرگتر و CPI کمتر از CISC است.

ب) در RISC تعداد دستورات بیشتر و طول برنامه تقریباً هم اندازه و CPI کمتر از CISC است.

ج) در RISC تعداد دستورات بیشتر، طول برنامه کوچکتر و CPI بیشتر از CISC است.

د) در RISC تعداد دستورات کمتر، طول برنامه کوچکتر و CPI کمتر از CISC است.

حل تست‌های فصل ۸

۱- گزینه‌ی د صحیح است.

با توجه به توضیحات حل مسئله (۸-۹) فصل واحد مرکزی پردازش، تعداد push در پشته برابر تعداد عملوندهای عبارت است. برای تشخیص حداقل ظرفیت پشته فرم RPN عبارت را تشکیل داده، و محاسبه را انجام می‌دهیم.

فرم RPN :

$$825 \times + 132 \times + 4 - /$$

وضعیت پشته: حداقل سایز پشته = ۴

۲- گزینه‌ی د صحیح است.

$$a : \text{load immediate } 20 \Rightarrow AC = 20$$

$$b : \text{load indirect } 20 \Rightarrow AC = M[M[20]] = 60$$

$$c : \text{load direct } 30 \Rightarrow AC = M[30] = 50$$

$$d : \text{load indirect } 30 \Rightarrow AC = M[M[30]] = 70$$

۳- گزینه‌ی ب صحیح است.

ابتدا از روی دنباله برنامه، فرم PRN عبارت را پیدا می‌کنیم.

$$\text{RPN عبارت} = AB * C - D / CE / A + BF * - +$$

حال طبق توضیحات حل مسئله ۸ از فصل ۸، فرم میانوندی را از فرم RPN بدست

آمده، بدست می‌آوریم.

$$\text{عبارت میانوندی} = (A * B - C) / D + (C / E + A - B * F)$$

۴- گزینه‌ی د صحیح است.

حالت (۱): پشته از آدرس‌های بالا به پایین رشد می‌کند و SP به بالاترین عنصر پشته اشاره می‌کند.

حالت (۲): پشته از آدرس‌های پایین به بالا رشد می‌کند و SP به بالاترین عنصر پشته اشاره می‌کند.

حالت (۳): پشته از آدرس‌های بالا به پایین رشد می‌کند و SP به مکان بالای بالاترین عنصر پشته اشاره می‌کند.

۵ - گزینه‌ی الف صحیح است.

مفهوم CPI به معنای تعداد کلاک مورد نیاز برای اجرای هر کدام از دستورات است. و در کل به ساده‌تر بودن یک دستور اشاره می‌کند.

علاوه بر این، در رابطه با مقایسه کامپیوترهای RISC و CISC داریم:

معیار	RISC	CISC
تعداد دستورات	کم	زیاد
طول دستورات	ثابت	متفاوت
(تعداد خطوط کد برنامه) طول برنامه	زیاد	کم
تعداد ثبات پردازنده	زیاد	کم
استفاده از پنجره‌های ثبات همپوشان	بلی	خیر
تعداد مد آدرس‌دهی	کم	زیاد

تست‌های فصل ۹ «پردازش خط لوله‌ای و

برداری»

۱- یک واحد محاسباتی لوله‌ای (pipeline) دارای پنج قسمت (stage) با زمان اجرای ۳۶، ۲۳، ۲۸ و ۶۴ نانوثانیه است. اگر از ثبات‌هایی با تأخیر ۱ نانوثانیه در بین قسمت‌های مختلف لوله استفاده شده باشد، حداکثر تسریع این واحد محاسباتی نسبت به تأخیر غیر خط لوله‌ای چقدر است؟

(مهندسی کامپیوتر - سراسری ۷۹)

(مشابه مهندسی کامپیوتر - سراسری ۸۵)

الف) $2/92$ ب) $3/5$ ج) 5 د) $7/92$

۲- یک برنامه موازی به ۱۰۰ ثانیه برای اجرا روی یک کامپیوتر با یک پردازنده دارد. اگر ۴۰٪ محاسبات برنامه ذاتاً سریال باشد، بهترین زمان برای اجرای این برنامه روی یک سیستم با ۲ و ۴ پردازنده چیست؟

(علوم کامپیوتر - ۸۱)

الف) ۲ پردازنده برابر ۷۰ ثانیه، ۴ پردازنده برابر ۵۵ ثانیه

ب) ۲ پردازنده برابر ۲۰ ثانیه، ۴ پردازنده برابر ۱۰ ثانیه

ج) ۲ پردازنده برابر ۳۰ ثانیه، ۴ پردازنده برابر ۱۵ ثانیه

د) ۲ پردازنده برابر ۵۰ ثانیه، ۴ پردازنده برابر ۲۵ ثانیه

۳- اگر k تعداد مراحل یک سیستم خط لوله‌ای (pipelined) باشد. آنگاه این سیستم خط لوله‌ای نسبت به یک سیستم بدون خط لوله حداکثر چند برابر سریعتر می‌باشد؟

(مهندسی IT - آزاد ۸۴)

الف) $2k+1$ ب) k ج) $2k$ د) $2k-1$

۴- مشکلات خط لوله (pipeline) دستورالعمل کدام است؟

(مهندسی کامپیوتر - آزاد ۸۴)

الف) مشکل همزمانی دسترسی به حافظه

ب) وابستگی داده‌ها

ج) مشکلات انشعاب

د) هر سه گزینه درست است.

حل تست‌های فصل ۹

۱- گزینه‌ی الف صحیح است.

زمان اجرا بدون خط لوله: در این محاسبه، زمان‌های داده شده را با هم جمع می‌کنیم و از تأخیر لچ‌ها صرف‌نظر می‌کنیم زیرا نیاز نیستند.

$$\text{زمان اجرای } n \text{ کار بدون خط لوله} = (64 + 28 + 23 + 39 + 36)n = 190n$$

زمان اجرا با راه‌اندازی خط لوله = ابتدا کلاک را تعیین می‌کنیم:

$$T = 64 + 1 = 65 \text{ ns}$$

$$\text{زمان اجرای } n \text{ کار با خط لوله} = (k + (n-1))T = (5 + (n-1))65$$

تسریع برابر است با:

$$S_{\max} = \frac{190n}{5 \times 65 + (n-1)65}, \quad n \rightarrow \infty \Rightarrow S_{\max} = \frac{190}{65} = 2.92$$

۲- گزینه‌ی الف صحیح است.

اگر t زمان اجرا روی یک پردازنده باشد، زمان اجرا با P پردازنده برابر است با:

$$t_p = \left(f + \frac{1-f}{p} \right) t$$

که f درصد از برنامه است که ذاتاً سریال می‌باشد.

$$f = \frac{40}{100}$$

$$\Rightarrow t_p = \left(\frac{40}{100} + \frac{\frac{60}{100}}{2} \right) 100 = 70$$

$$t_p = \left(\frac{40}{100} + \frac{\frac{60}{100}}{4} \right) 100 = 55$$

$$p = 4$$

۳- گزینه‌ی ب صحیح است.

حداکثر تسریع در استفاده از خط لوله زمانی بدست می‌آید که اولاً زمان اجرای مرحله‌ها با هم برابر باشد و ثانیاً تعداد کارها (n) را به سمت بی‌نهایت سوق دهیم.

$$\lim_{n \rightarrow \infty} \frac{nkT}{(k + (n-1))T} = k$$

۴- گزینه‌ی د صحیح است.

انواع مشکلات موجود در خط لوله عبارت‌اند از:

- ۱- همزمانی دسترسی به منابع مشترک مثل حافظه یا ALU یا...
- ۲- وابستگی داده‌ای به طوری که دستور کنونی به نتیجه اجرای دستور قبلی نیاز دارد.

۳- مشکل مربوط به دستورات انشعاب

تست‌های فصل ۱۰ «معماری کامپیوتر و

الگوریتم‌های حسابی»

۱- نتیجه حاصل از انجام محاسبات در ALU ممکن است باعث set شدن برخی فلیپ فلاپ‌ها در رجیستر وضعیت شود. اگر برای نمایش اطلاعات از سیستم مکمل ۲ استفاده شود، کدام یک از عبارات زیر به ترتیب وجود overflow در جمع و نتیجه صفر را تشخیص می‌دهد؟

(مهندسی کامپیوتر - سراسری ۷۱)

الف) رقم نقلی نتیجه، AND منطقی بیت‌های نتیجه

ب) OR انحصاری ارقام نقلی تولید شده از موضع sign - bit و ماقبل آن، NOR بیت‌های نتیجه

ج) OR انحصاری ارقام نقلی تولید شده از موضع sign - bit و ماقبل آن، AND بیت‌های نتیجه

د) رقم نقلی نتیجه، NAND منطقی بیت‌های نتیجه

۲- می‌خواهیم دو عدد علامت‌دار را (با قرار دادن مکمل ۲ها) در هم ضرب کنیم.

Multiplicand = A

Multiplier = B

(مهندسی کامپیوتر - سراسری ۷۱)

الف) علامت A و B را مثبت می‌کنیم و سپس ضرب می‌کنیم و پس از آن در صورت لزوم علامت را اصلاح می‌کنیم.

ب) اگر A منفی بود آن را مثبت می‌کنیم، ضرب را با گسترش علامت انجام می‌دهیم، حاصلضرب را در صورت لزوم تغییر علامت می‌دهیم.

ج) اگر B منفی بود، آن را مثبت کرده، ضرب را با گسترش علامت انجام می‌دهیم، حاصلضرب را در صورت لزوم تغییر علامت می‌دهیم.

د) ضرب را با گسترش علامت انجام می‌دهیم.

۳- الگوریتم Booth دارای کدام یک از ویژگی‌های زیر است؟

(مهندسی کامپیوتر - سراسری ۷۳)

الف) سرعت محاسبه با تعداد ۱ها متناسب است.

ب) تعداد عملیات جمع برابر نصف تعداد ۱ها و عملیات جابجایی (shift) برابر تعداد بیت‌ها است.

ج) تعداد زوج بیت‌های صفر و یک، یا یک و صفر، تعداد جمع‌ها و تعداد کل بیت‌ها، تعداد عملیات جابجایی را تعیین می‌کند.

د) سرعت محاسبه با تعداد بیت‌ها متناسب است.

۴- در عمل تقسیم در چه صورت divide - overflow رخ می‌دهد؟

(علوم کامپیوتر - ۸۰)

الف) فقط زمانی که تقسیم به صفر انجام شود.

ب) عدد موجود در نیمه بالای خارج قسمت از مقسوم علیه بزرگتر باشد.

ج) باقیمانده یک عدد منفی باشد.

د) هیچکدام

۵- در انجام کدام یک از چهار عمل اصلی روی مانتیس اعداد با ممیز شناور،

سرریز رخ نمی‌دهد؟

(آزاد - ۷۹)

الف) جمع ب) ضرب ج) تفریق د) تقسیم

۶- در یک سیستم نمایش اعداد حقیقی به شکل ممیز شناور به صورت زیر

است:

۰ ۶ ۷ ۳۰ ۳۱

توان	مانتیس	S
------	--------	---

در یک سیستم دیگر به صورت زیر نمایش داده می شود.

۰ ۷ ۸ ۳۰ ۳۱

توان	مانتیس	S
------	--------	---

کدام یک از جملات زیر درباره محدوده اعداد حقیقی و دقت مدل دوم نسبت به مدل صحیح می باشد؟

(مهندسی IT – سراسری ۸۳)

الف) هر دو محدوده و دقت کاهش می یابد.

ب) هر دو محدوده و دقت افزایش می یابد.

ج) محدوده افزایش و دقت کاهش می یابد.

د) محدوده کاهش و دقت افزایش می یابد.

۷- در اعداد ممیز شناور کوچکترین نما را به عدد صفر اختصاص می دهیم برای

اینکه:

(سراسری ۸۳ – مهندسی IT)

الف) به نمایش نرمال شده دست پیدا کنیم.

ب) خطای محاسبات حداقل شود.

ج) با مقدار افزودن نما حاصل صفر تولید شود.

د) تشخیص صفر توسط مدارات داخل پردازنده ساده تر صورت گیرد.

حل تست‌های فصل ۱۰

۱- گزینه‌ی ب صحیح است.

تشخیص overflow از طریق OR انحصاری دو رقم نقلی آخر امکان‌پذیر است. همچنین برای تشخیص صفر بودن باید گیتی بکار ببریم که تنها در صورتی ۱ شود که همه ورودی‌ها ۰ باشد.

۲- گزینه‌ی ج صحیح است.

به دلیل اینکه صفر و ۱ بودن بیت‌های Multiplier در الگوریتم ضرب، جمع شدن و یا عدم جمع شدن حاصلضرب جزئی را باعث می‌شوند. سعی ما بر این است که Multiplier، مثبت باشد. پس اگر مثبت بود ضرب را انجام می‌دهیم و اگر منفی بود، دو راه داریم:

یا هر دو Multiplier و Multiplicand را تغییر علامت می‌دهیم و ضرب را انجام می‌دهیم.

یا Multiplier را مثبت کرده و بعد از انجام عمل ضرب، حاصلضرب را تغییر علامت می‌دهیم.

۳- گزینه‌ی ج صحیح است.

با توجه به الگوریتم و سرعت عملیات ضرب و تعداد عملیات جمع، وابسته به تعداد زوج بیت‌های ۰۰، ۰۱ و ۱ در عدد است.

۴- گزینه‌ی د صحیح است.

مواردی که باعث می‌شوند در عمل تقسیم، سرریز رخ دهد عبارت‌اند از:

- زمانی که مقسوم علیه ۰ باشد.

- زمانی که نیمه بالای مقسوم از مقسوم علیه بزرگتر یا مساوی باشد.

لفظ «فقط» در گزینه‌ی الف، باعث غلط بودن این گزینه است.

۵- گزینه‌ی ب صحیح است.

مطابق با توضیحات حل مسئله‌ی ۲۲-۱۰ فصل «الگوریتم‌های حسابی و معماری کامپیوتر» سرریز مانتیس بعد از عمل ضرب ممکن نیست.

۶- گزینه‌ی ج صحیح است.

در اعداد ممیز شناور، که به فرم $0/100 \times 2^e \times (-1)^s$ هستند، بیت‌های بیشتر برای مانتیس دقت عدد را افزایش می‌دهد زیرا باعث می‌شود تعداد بیشتری رقم اعشاری داشته باشیم. از طرفی بیت‌های بیشتر برای نما، باز ۰ نمایش را زیاد می‌کند چون در قسمت توان عدد، واقع شده است.

۷- گزینه‌ی ب صحیح است.

دلیل استفاده از نمای بایاس شده کاهش خطای محاسبات و همچنین مقایسه راحت‌تر بین اعداد ممیز شناور است.

تست‌های فصل ۱۱ «سازمان ورودی – خروجی»

۱- عمل Direct Memory Access (DMA) در میکروپروسورها به چه منظور

است؟

(مهندسی کامپیوتر – سراسری (۷۱)

الف) انتقال اطلاعات بین دستگاه‌های جانبی

ب) انتقال اطلاعات بین دستگاه جانبی و حافظه از طریق میکروپروسور

ج) انتقال مستقیم اطلاعات از دستگاه جانبی به حافظه

د) هیچکدام

۲- برای انتقال داده‌های هشت بیتی روی دو سیم از یک بیت شروع (start)، یک

بیت ختم (STOP) و یک بیت توازن زوج (Even Parity) استفاده می‌شود. سرعت

انتقال (Baud rate) برابر با ۱۲۰۰ بیت بر ثانیه می‌باشد. برای انتقال پیاپی ۱k بایت

اطلاعات، چند ثانیه وقت مورد نیاز است؟

(مهندسی کامپیوتر – سراسری (۸۱)

الف) ۹/۳۹ ب) ۸/۵۳ ج) ۷/۶۸ د) ۶/۸۳

۳- سیکل ربایی (cycle stealing) مربوط به کدام مفهوم زیر است؟

(آزاد ۷۷)

الف) NMI ب) HALT ج) DMA د) T_{wait}

۴- دادن حق تقدم به وقفه ایجاد شده توسط دستگاه‌های جانبی به کدام طریق

انجام می‌گیرد؟

(آزاد ۸۴)

الف) وقفه زنجیری ب) وقفه تقدم موازی

ج) هر دو گزینه ۱ و ۲ د) هیچکدام

۵ - کدام گزینه در مورد روش‌های I/O صحیح نیست؟

(مهندسی IT – سراسری ۸۵)

- الف) روش programmed I/O پردازنده اصلی را درگیر عملیات I/O می‌کند.
- ب) روش Interrupted I/O پردازنده اصلی را درگیر عملیات I/O می‌کند.
- ج) روش DMA I/O پردازنده اصلی را درگیر عملیات I/O می‌کند. (در حین I/O)
- د) روش I/O با استفاده از هم پردازنده خاص I/O امکان نوشتن برنامه‌های مختلط از دستورات پردازنده اصلی و هم پردازنده I/O را می‌دهد.

حل تست‌های فصل ۱۱

۱- گزینه‌ی ج صحیح است.

روش DMA، از درگیر کردن CPU در جزئیات عمل انتقال اطلاعات بین دستگاه جانبی و حافظه جلوگیری می‌کند.

۲- گزینه‌ی الف صحیح است.

با توجه به بیت‌های شروع، ختم و توازن، هر بایت اطلاعات به صورت یک بسته‌ی ۱۱ بیتی منتقل می‌شود. پس:

$$1024 \times 11 = 11264 \text{ bit} \Rightarrow \frac{11264}{1200} = 9,386$$

۳- گزینه‌ی ج صحیح است.

کنترل‌کننده DMA به چند روش عمل انتقال اطلاعات بین حافظه و I/O را انجام می‌دهد که یکی از روش‌ها سیکل ربایی است. در این روش بعد از تبادل ۱ کلمه اطلاعات، گذرگاه دوباره در اختیار CPU قرار می‌گیرد.

۴- گزینه‌ی ج صحیح است.

دادن حق تقدم به وقفه ایجاد شده به سه روش زیر قابل انجام است.

- سرکشی (polling)

- زنجیره‌ای (Daisy - chain)

- موازی

۵- گزینه‌ی ج صحیح است.

در روش Interrupted I/O، صرفاً از بررسی‌های پشت سرهم توسط CPU جلوگیری به عمل می‌آید، اما اجرای درخواست را بر عهده خواهد داشت.

تست‌های فصل ۱۲ «سازمان حافظه»

۱- یک پردازنده دارای n خط آدرس J خط داده است. فقط یک تراشه‌ی حافظه با m خط آدرس و J خط داده را مستقیماً به پردازنده وصل کرده‌ایم ($n > m$)، هر کلمه J بیتی از این تراشه با چند آدرس قابل دسترسی است؟

(مهندسی کامپیوتر - سراسری (۸۱)

الف) 2^{ϕ} ب) 2^{n-m}

ج) 2^{n-m-1} د) قابل محاسبه نیست.

۲- سیستمی را که دارای دو سطح حافظه است، در نظر بگیرید. M_1 حافظه پنهان (cache) سیستم است و M_2 حافظه اصلی. زمان دسترسی به cache برابر 20ns و ضریب hit برابر ۰/۹۵ است. زمان دسترسی به حافظه M_2 باید چقدر باشد تا زمان مؤثر دسترسی به اطلاعات به صورت کلی 40ns باشد؟

(علوم کامپیوتر - ۷۹)

الف) 400ns ب) 42ns ج) 210ns د) 21ns

۳- در یک پردازنده سرعت 5 mhz است و باس داده‌ای آن ۳۲ بیتی می‌باشد. در طراحی جدید این پردازنده، سرعت را به 20mhz رسانده‌اند ولی باس داده‌ای آن را ۸ بیتی کرده‌اند. ماکزیمم speed - up به دست آمده در طراحی جدید نسبت به قبل به چه صورت است؟

(علوم کامپیوتر - ۸۰)

الف) ۲ برابر شده است. ب) speed - up ثابت مانده است.

ج) ۴ برابر شده است. د) ۱۴ برابر شده است.

۴- کدام یک از موارد زیر از مزایای استراتژی write through است؟

(مهندسی IT - آزاد (۸۴)

- الف) اطلاعات را می‌توان با سرعت cache نوشت.
- ب) از پهنای باند انتقالی حداکثر استفاده به عمل می‌آید.
- ج) برخورد با حالت Miss در حافظه cache راحت‌تر و ساده‌تر می‌باشد.
- د) موارد ۲ و ۳

۵- در صورتی که دسترسی به حافظه نهان (cache) از طریق نگاشت مستقیم (Direct Mapping) به روش مقابل باشد، مشخص کنید به ترتیب حجم حافظه اصلی، حافظه نهان و محل استقرار آدرس $(000B3A4F)_H$ چیست؟

(مهندسی کامپیوتر - سراسری ۸۴)

۴ بیت	۷ بیت	۲۱ بیت
offset	Block	tag

- الف) $Block(164)_{10}, 4GB, 2KB$ ب) $Block(36)_{10}, 2GB, 4KB$
- ج) $Block(160)_{10}, 4GB, 2KB$ د) $Block(36)_{10}, 4GB, 2KB$
- ۶- تفاوت اصلی بین (SRAM) Static Ram و (DRAM) Dynamic Ram چیست؟
- (علوم کامپیوتر - ۸۴)

- الف) SRAM سریع‌تر و احتیاج به refresh دارد و DRAM کندتر و احتیاج به Refresh ندارد.
- ب) SRAM کندتر و احتیاج به refresh دارد و DRAM سریع‌تر و احتیاج به refresh ندارد.
- ج) SRAM کندتر و احتیاج به refresh ندارد و DRAM سریع‌تر و احتیاج به refresh دارد.
- د) SRAM سریع‌تر و احتیاج به refresh ندارد و DRAM کندتر و احتیاج به refresh دارد.

حل تست‌های فصل ۱۲

۱- گزینه‌ی ب صحیح است.

$$\frac{2^n}{2^m} = 2^{n-m} \quad 2^{n-m} \text{ پس هر خانه با آدرس قابل دسترسی است.}$$

۲- گزینه‌ی الف صحیح است.

$$t = h.t_c + (1-h)(t_c + t_m) \text{ میانگین (مؤثر)}$$

در رابطه بالا داریم :

t_c = زمان دسترسی به حافظه نهان

t_m = زمان دسترسی به حافظه

$$\Rightarrow 40 = \frac{95}{100} \times 20 + \frac{5}{100} (20 + t_m) \Rightarrow 40 = 19 + 1 + \frac{5t_m}{100}$$

۳- گزینه‌ی ب صحیح است.

طبق تعریف، پهنای باند حافظه با عرض گذرگاه (باس) و همچنین فرکانس پردازنده رابطه مستقیم دارد.

$$\frac{BW_2}{BW_1} = \frac{8}{32} \times \frac{20}{5} = 1$$

پس تغییر در پهنای باند حافظه و در نتیجه آن، سرعت انتقال پدید نمی‌آید.

۴- گزینه‌ی ج صحیح است.

دو استراتژی در نوشتن در حافظه نهان وجود دارد:

– write – through = کلمه تغییر یافته در حافظه نهان، در حافظه نیز نوشته می‌شود.

(به هنگام می‌شود).

– write – back = تا زمانی که کلمه داخل حافظه نهان است، در حافظه اصلی به روز

نمی‌شود و در صورت حذف کلمه از حافظه نهان، حافظه اصلی به روز می‌شود.

استراتژی اول، اگرچه سرعت نوشتن را پایین می‌آورد اما از یکسان بودن محتوای

حافظه نهان و حافظه اصلی اطمینان داریم.

۵ - گزینه‌ی د صحیح است.

تعداد کل بیت‌ها ۳۲ بیت است پس حجم حافظه اصلی:

$$2^{32} = 4\text{GB}$$

حجم تعداد بیت‌های block و offset = ۱۱ = ۷+۴ پس حجم حافظه نهان (کش):

$$2^{11} = 2\text{KB}$$

برای تشخیص بلاک مورد اشاره داریم:

$$(B3A4F)_H = (101100111010\ 0100\ 1111)_2$$

Block offset

$$= (36)_{10}$$

۶ - گزینه‌ی د صحیح است.

حافظه Dynamic Ram به دلیل نیاز به Refresh کندتر است.

منابع

- 1- M.Morris Mano , Computer System Architecture : Solution Manual
- 2- M.Morris Mano , Computer System Architecture, 3rd Edition,