



۱. (آ) برای هر دو حالت بررسی می‌کنیم:

● **اتصال نقطه به نقطه:** در این روش برای ورودی هر ثبات نیاز به ۳۲ مالتی پلکسر داریم و به دلیل اینکه قطعات ما تک‌بیتی هستند پس برای هر بیت نیز یک مالتی پلکسر می‌خواهیم و در کل تعداد مالتی پلکسرهای ما برابر با  $1024 = 32 \times 32$  است. برای مالتی پلکسرهای هر ثبات نیاز به پنج سیگنال select داریم که در کل برابر با  $160 = 32 \times 5$  است و برای هر ثبات هم به یک سیگنال load نیاز داریم و در نهایت تعداد سیگنال‌های کنترلی برابر با ۱۹۲ می‌شود. (یک فرض دیگر این است که در هر کلاک صرفاً می‌توانیم در یک ثبات بنویسیم که در این حالت بهتر است از یک کدگشای<sup>۱</sup> ۵ به ۳۲ استفاده کنیم تا سیگنال‌های load را بسازیم که در این حالت به ۵ بیت سیگنال کنترلی و یک بیت هم enable کدگشا نیاز داریم و تعداد کل سیگنال‌ها برابر با ۱۶۵ می‌شود)

● **ارتباط common bus:** در این روش از یک bus ۳۲ بیتی استفاده می‌کنیم تا داده‌ها را میان ثبات‌ها منتقل کند. برای کنترل کردن ورودی bus می‌توانیم از  $1024 = 32 \times 32$  عدد tri-state buffer و یا ۳۲ مالتی پلکسر ۳۲ به ۱ استفاده کنیم.

در صورتی که از tri-state buffer استفاده کرده باشیم، برای سیگنال‌های کنترلی آن‌ها به یک کدگشای ۵ به ۳۲ داریم که باعث می‌شود سیگنال‌های کنترلی برای کنترل ورودی‌های bus برابر با ۵ بیت باشند. در صورتی که از مالتی پلکسر استفاده کرده باشیم، نیازی به کدگشا نداشته و صرفاً به ۵ بیت سیگنال کنترلی برای select مالتی پلکسر نیاز داریم.

برای کنترل ورودی‌های ثبات‌ها به سیگنال input enable نیاز داریم که از دو روش قابل تأمین است. در روش اول فرض بر این است که در هر کلاک می‌توانیم فقط در یک ثبات بنویسیم که در این صورت بهتر است از یک کدگشای ۵ به ۳۲ استفاده کنیم که در این حالت سیگنال کنترلی برابر با ۵ بیت می‌شود. اما اگر بخواهیم در هر کلاک در بیشتر از یک ثبات بنویسیم، باید برای هر ثبات یک سیگنال کنترلی جدا در نظر بگیریم که برابر با ۳۲ سیگنال می‌شود.

(ب) برای هر دو حالت بررسی می‌کنیم:

● **اتصال نقطه به نقطه:** در این حالت تعداد مالتی پلکسرهای برابر با  $2048 = 32 \times 64$  می‌شود و سیگنال‌های select برای مالتی پلکسرهای هر ثبات برابر با ۶ بیت و در کل برای همه ثبات‌ها برابر با  $384 = 64 \times 6$  می‌شود. اگر برای load ثبات‌ها از کدگشا استفاده کنیم تعداد کل سیگنال‌ها برابر با ۳۹۰ و اگر از کدگشا استفاده نکنیم برابر با ۴۴۸ می‌شود.

● **ارتباط common bus:** در صورتی که از tri-state buffer استفاده کرده باشیم، تعداد آن‌ها برابر با  $2048 = 32 \times 64$  می‌شود و این باعث می‌شود تا از کدگشای ۶ به ۶۴ استفاده کنیم و سیگنال‌های کنترلی برای ورودی bus برابر با ۶ بیت شود.

<sup>1</sup>Decoder

اگر از مالتی پلکسر استفاده کنیم، باید از ۳۲ مالتی پلکسر ۶۴ به ۱ استفاده کنیم که در این حالت نیز به ۶ بیت سیگنال کنترل نیاز داریم.

برای سیگنال ورودی ثبات‌ها نیز اگر از کدگشا استفاده کنیم، به سیگنال کنترل ۶ بیتی نیاز داریم و اگر از کدگشا استفاده نکنیم، به ۶۴ بیت سیگنال کنترل نیاز داریم.

(ج) برای هر دو حالت بررسی می‌کنیم:

● **اتصال نقطه به نقطه:** در این حالت تعداد مالتی پلکسرهایی که برای هر ثبات استفاده می‌شود ۳۲ عدد افزایش پیدا می‌کند و در کل تعداد مالتی پلکسرها ۱۰۲۸ عدد افزایش پیدا می‌کند و به ۲۰۴۸ می‌رسد و تغییر دیگری نخواهیم داشت.

● **ارتباط common bus:** در این حالت به bus ۶۴ بیتی نیاز داریم و تعداد tri-state bufferها برای هر ثبات ۳۲ عدد افزایش پیدا کرده و در کل تعداد آن‌ها ۱۰۲۸ عدد افزایش یافته و به ۲۰۴۸ می‌رسد. اگر از مالتی پلکسر استفاده کرده باشیم، باید تعداد آن‌ها را ۳۲ عدد افزایش دهیم و به ۶۴ مالتی پلکسر ۳۲ به ۱ نیاز داریم.

به همه حالات بیان شده در پاسخنامه نمره کامل تعلق خواهد گرفت. همچنین بیان نکردن سیگنال‌های enable برای کدگشاها نیز مشکلی ندارد.

۲. (آ) توصیف مدار به صورت RTL به شکل زیر است:

$$reset: R1 \leftarrow 0, R2 \leftarrow 0$$

$$\overline{C}: E \leftarrow R1$$

$$C: E \leftarrow R2 + 2$$

$$CLK.B.\overline{reset}: R1 \leftarrow A$$

$$CLK.\overline{reset}: R2 \leftarrow E$$

در صورتی که در شرط، کلاک را ننوشته باشید یا هر پاسخ دیگری که نتیجه درست بدهد را ننوشته باشید، نمره کامل را دریافت خواهید کرد.

(ب) initial values

$$R1 = 0, E = 2, R2 = 0$$

$$: t = 0$$

$$R1 = 0, E = 4, R2 = 2$$

$$: t = 10$$

$$R1 = 5, E = 6 \ (E = 5 \text{ in } 10 < t < 20), R2 = 4$$

$$: t = 20$$

$$R1 = 19, E = 19 \ (E = 7 \text{ in } 20 < t < 30), R2 = 5$$

$$: t = 30$$

$$R1 = 19, E = 9 \ (E = 19 \text{ in } 30 < t < 40), R2 = 7$$

$$: t = 40$$

$$R1 = 11, E = 11 \ (E = 21 \text{ in } 40 < t < 50), R2 = 19$$

$$: t = 50$$

$$R1 = 11, E = 23, R2 = 21$$

۳. ناچاریم برای ثبت حالتی که در آن هستیم از یک ثابت اضافه به نام S استفاده کنیم که مقادیر 0, 1, 2 در آن به ترتیب از کوچک به بزرگ مربوط به A, B, C هستند.

(آ)

$$(S == 0) : R0 \leftarrow R0 + 1$$

$$(S == 0)(R0 == 8) : S \leftarrow S + 1$$

$$(S == 1) : R0 \leftarrow R0 + 1, R2 \leftarrow R0 \oplus R2$$

$$(S == 1)(R0 == R2) : S \leftarrow S + 1$$

$$(S == 2) : R2 \leftarrow R2 + R1 - 1, R1 \leftarrow R1 - 1$$

$$(S == 2)(R1 == 0) : S \leftarrow 1$$

$$(S == 2)(R2 == 0) : S \leftarrow 1$$

البته ممکن است حالات دیگری توسط دانشجویان پیشنهاد شود که قابل قبول باشند.

(ب) حالات متعددی برای این مدار هست. هر یک که شرایط سوال را برآورده کند قابل قبول خواهد بود. به طور مثال مدار برای RTL زیر به نمایش گذاشته شده است.

$$(A = 1) : R0 \leftarrow R0 + 1$$

$$(A = 1)(R0 = 0) : A \leftarrow 0, B \leftarrow 1$$

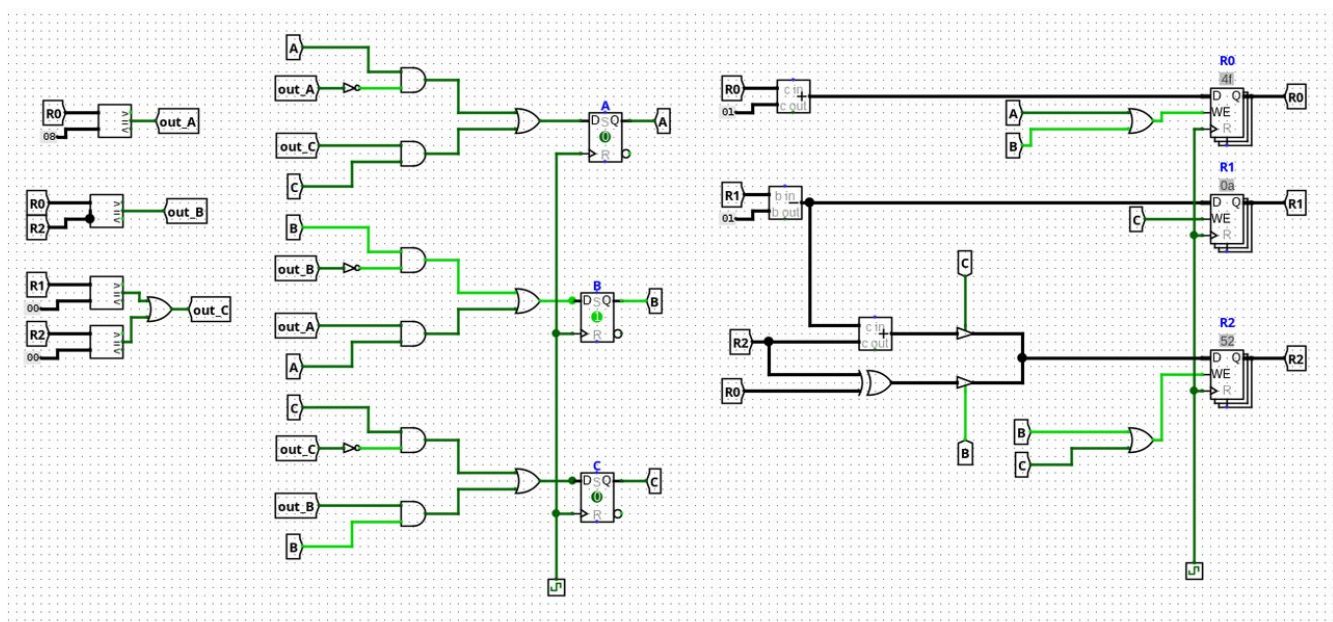
$$(B = 1) : R2 \leftarrow R0 \oplus R2, R0 \leftarrow 1$$

$$(B = 1)(R0 = R2) : B \leftarrow 0, C \leftarrow 1$$

$$(C = 1) : R1 \leftarrow R1 - 1, R2 \leftarrow R2 + R1 - 1$$

$$(C = 1)(R1 = 0) : C \leftarrow 0, A \leftarrow 1$$

$$(C = 1)(R2 = 0) : C \leftarrow 0, A \leftarrow 1$$



۴. ۱.  $R_2 = 1$  و  $R_1 = 0$ ،  $R_0 = 2$

۲. ۲۳ کلاک با احتساب کلاک اتمام طول می‌کشد.

۳. ابتدا به ترتیب شروط ۱ تا ۵ برقرار است سپس سه بار دیگر نیز همین اتفاق می‌افتد و سپس شرط ۱، ۲ و ۶ به ترتیب برقرار می‌شوند.

1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 6

۴. می‌توان دید که مقدار ۰ و ۱، ۵ بار مقدار ۲ شش بار و مقدار ۳ و ۴، ۴ بار دیده شدند. (البته اگر صفر به خاطر در نظر نگرفتن حالت اولیه ۴ هم در نظر گرفته شده باشد درست است. همچنین اگر برای ۲ حالت ترمینیت در نظر گرفته نشده باشد و مقدار ۵ ذکر شده باشد.)

- شرط اول: ۵
- شرط دوم: ۵
- شرط سوم: ۴
- شرط چهارم: ۴
- شرط پنجم: ۴
- شرط ششم: ۱

۵. با در نظر داشتن الگوریتم اقلیدسی و همچنین روش تفریق‌های متوالی برای انجام عمل تقسیم می‌توان پاسخ زیر را ارائه داد.  
(تمام پاسخ‌هایی که از نظر منطقی صحیح باشند مورد قبول هستند.)

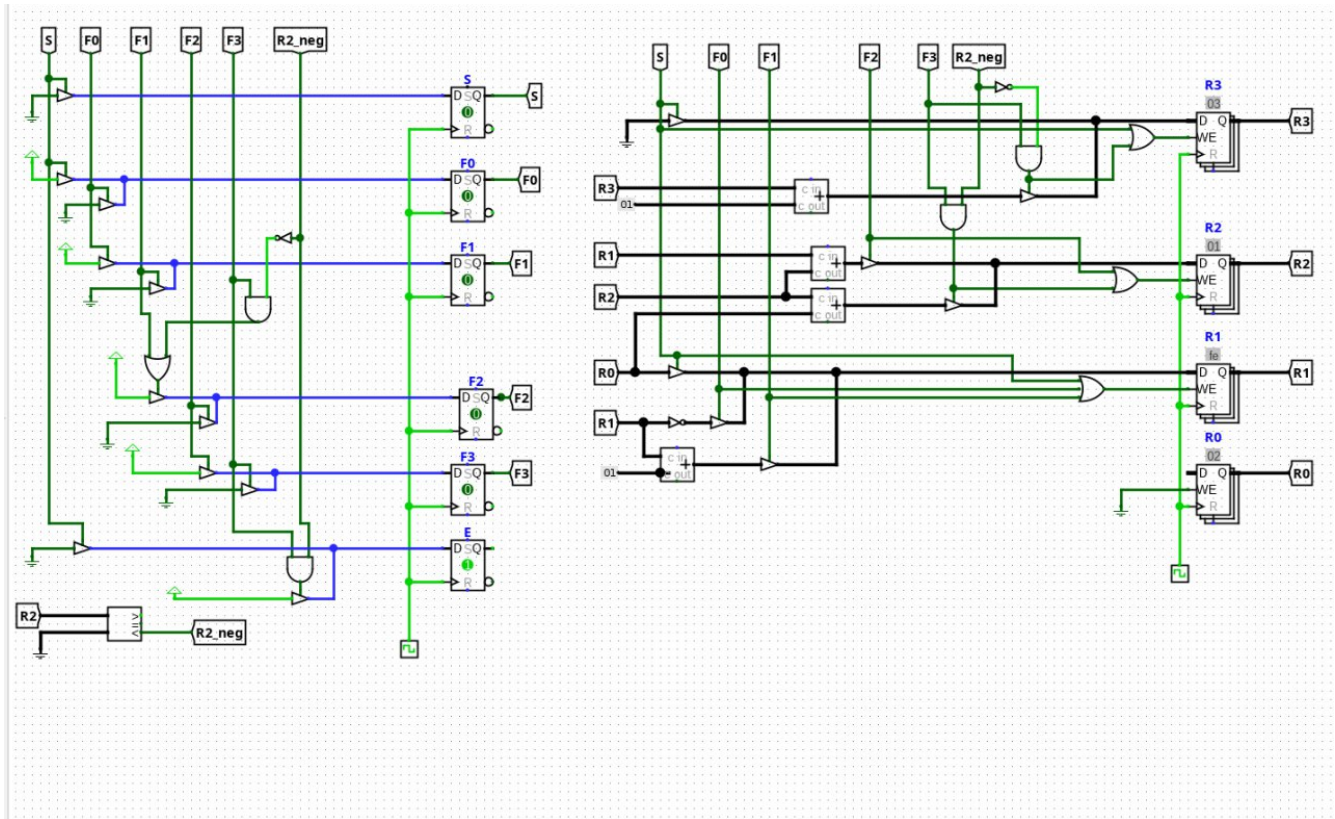
$$(r1 == 0) : r3 \leftarrow r2$$

$$(r1 \leq r2) : r2 \leftarrow r2 - r1$$

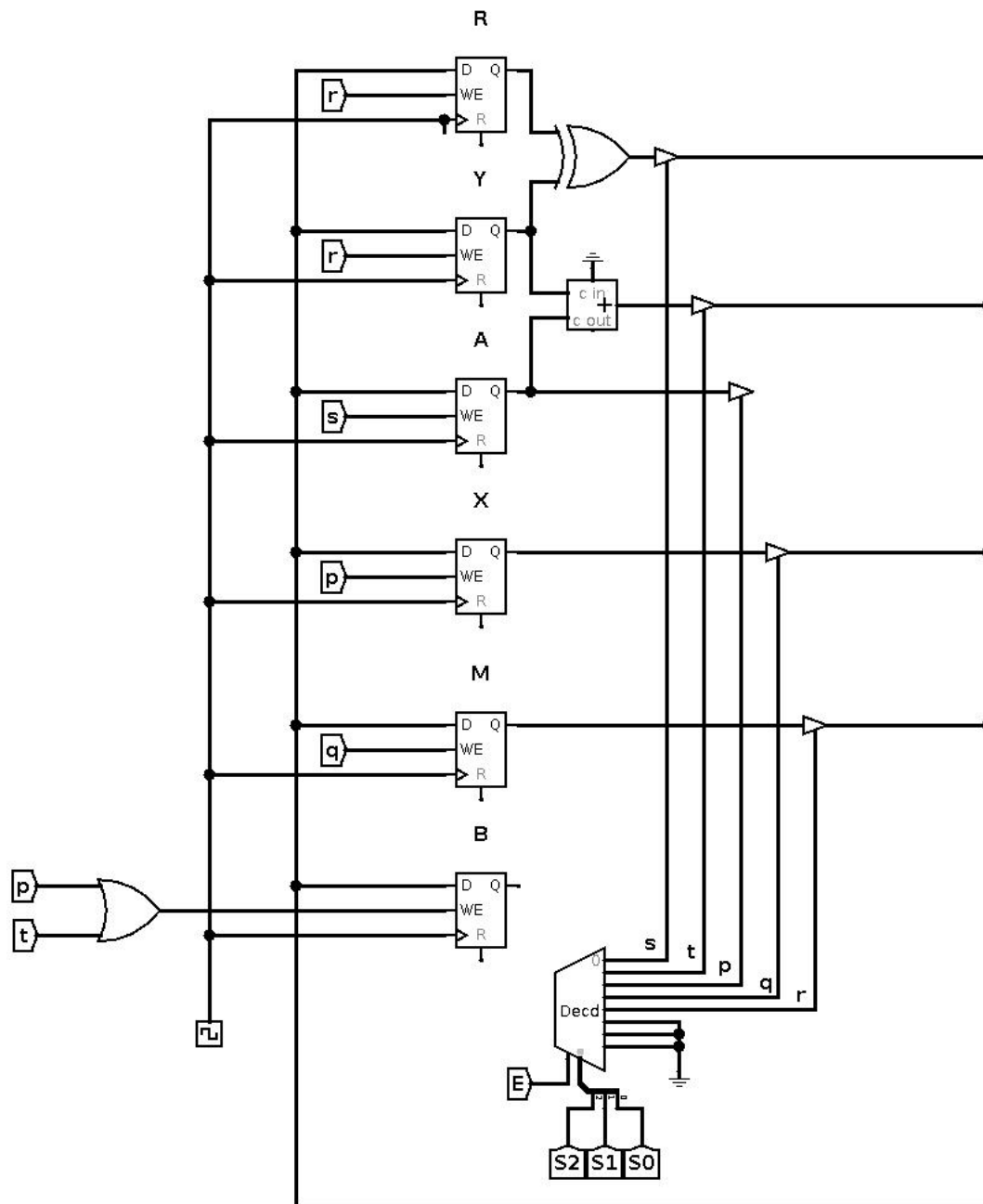
$$(r1 > r2) : r1 \leftarrow r2, r2 \leftarrow r1$$

۶. عملکرد این RTL بدین شکل است که مقدار رجیستر  $R_2$  را بر  $R_0$  تقسیم می‌کند و مقدار باقی‌مانده را در  $R_2$  و مقدار خارج قسمت را در  $R_3$  می‌ریزد.

یک نمونه از مدار صحیح این RTL در شکل زیر به نمایش گذاشته شده است.



۷. پاسخ به صورت زیر است:





۸. (آ) دستورات ۲ آدرس به صورت زیر است:

$$\begin{aligned} R1 &\leftarrow M[a] \\ R1 &\leftarrow R1 + M[b] \\ R1 &\leftarrow R1 + M[c] \\ R2 &\leftarrow M[d] \\ R2 &\leftarrow R2 \times M[e] \\ R2 &\leftarrow R2 - M[f] \\ R1 &\leftarrow R1 \times R2 \\ R3 &\leftarrow M[h] \\ R3 &\leftarrow R3 \times M[i] \\ R3 &\leftarrow R3 + M[g] \\ R1 &\leftarrow R1/R3 \\ M[O] &\leftarrow R1 \end{aligned}$$

(ب) دستورات تک آدرس به صورت زیر است:

$$\begin{aligned} ACC &\leftarrow M[a] \\ ACC &\leftarrow ACC + M[b] \\ ACC &\leftarrow ACC + M[c] \\ M[x] &\leftarrow ACC \\ ACC &\leftarrow M[d] \\ ACC &\leftarrow ACC \times M[e] \\ ACC &\leftarrow ACC - M[f] \\ ACC &\leftarrow ACC \times M[x] \\ M[x] &\leftarrow ACC \\ ACC &\leftarrow M[h] \\ ACC &\leftarrow ACC \times M[i] \\ ACC &\leftarrow ACC + M[g] \\ ACC &\leftarrow M[x]/ACC \\ M[O] &\leftarrow ACC \end{aligned}$$

(ج) دستورات را خط به خط اجرا و در هر مرحله مقدار ثبات ها را نیز مشخص می کنیم:

$$\begin{aligned} R1 &= (F2)_H = (11110010)_2, R2 = (FF)_H = (11111111)_2 \\ R3 &= (B9)_H = (10111001)_2, R4 = (EA)_H = (11101010)_2 \end{aligned}$$

(i)

$$\begin{aligned} R1 &= (F2)_H + (FF)_H = (F1)_H, R2 = (FF)_H \\ R4 &= \text{shl}(EA)_H = (D4)_H, R3 = (B9)_H + (D4)_H = (8D)_H \end{aligned}$$

(ii)

$$\begin{aligned} R3 &= (8D)_H \wedge (D4) = (84)_H, R4 = (D4)_H \\ R1 &= (F1)_H, R2 = (FF)_H + (01)_H = (00)_H \end{aligned}$$

(iii)

$$\begin{aligned} R1 &= (F1)_H - (84)_H = (6D)_H, R2 = (00)_H \\ R3 &= (84)_H, R4 = (D4)_H \end{aligned}$$

(iv)

$$R3 = \text{sar}(84)_H = (C2)_H, R4 = \text{cir}(D4)_H = (6A)_H$$

$$R1 = (06)_H \oplus (6A)_H = (A8)_H, R2 = (00)_H$$