

به نام خدا

گزارش کار پنجم آزمایشگاه مدارهای منطقی



دانشکده مهندسی کامپیووتر

آزمایش پنجم: واحد محاسبات و منطق (ALU)

علی محمدزاده شبستری

۴۰۱۱۰۶۴۸۲

تابستان ۱۴۰۲

فهرست

۳.	مقدمه.....
۳.	آشنایی با تراشه ۷۴۱۸۱.....
۱۱	ساخت مدار داخلی ALU
۱۸	جمع بندی
۱۸	منابع.....

مقدمه

هدف این آزمایش بررسی طرز کار و اتصالات و ساخت یک واحد محاسبات و منطق (ALU) است. این آزمایش دو بخش دارد:

در بخش اول: یک رشته چهاربیتی دیتا و یک رشته سه بیتی کنترل ورودی می‌گیریم و با استفاده از گیت‌های منطقی مناسب واحد کنترل را طبق خواستهٔ داخل گزارش کار می‌سازیم. واحد کنترل به ازای هشت حالت سه بیت کنترل برای ما مشخص می‌کند که چه عملیاتی انجام شود (هشت عملیات). ما در این بخش از قطعه ۷۴۱۸۱ (ALU)، دو قطعه ۷۴۱۷۵ (MUX) و دو قطعه ۷۴۱۵۷ (register) استفاده می‌کنیم.

در بخش دوم: مدار و اتصالات داخلی یک ALU را پیاده‌سازی می‌کنیم. با استفاده از چهار بیت سلکت و یک بیت کری نقلی ورودی، چهارده عملیات روی دو رشته چهار بیتی را پیاده‌سازی می‌کنیم.

آشنایی با تراشه ۷۴۱۸۱

:: بخش تئوری ::

بررسی تراشه ۷۴۱۸۱ (ALU): ورودی‌های آن به صورت زیر هستند

A_0, A_1, A_2, A_3 : رشته اول ورودی دیتا •

B_0, B_1, B_2, B_3 : رشته دوم ورودی دیتا •

C_n : کری ورودی (متصل به power) •

M : مد، صفر یعنی ریاضی (arithmetic) و یک یعنی منطقی (logical) •

S_0, S_1, S_2, S_3 : رشته سلکت (قابلیت ایجاد ۱۶ حالت مختلف) •

اکنون خروجی‌های آن را بررسی می‌کنیم:

F_0, F_1, F_2, F_3 : خروجی حاصل محاسبات ALU •

چهار خروجی دیگر هم دارد که از آن‌ها استفاده‌ای نمی‌کنیم. •

بررسی تراشه ۷۴۱۵۷ (MUX): عملکرد آن این‌گونه است که دو رشته چهار بیتی ورودی می‌گیرد، و یکی از آن‌ها را خروجی می‌دهد. تصمیم این که کدام رشته خارج شود به عهده بیت سلکت است که روی تراشه با نماد \bar{A}/B نشان داده شده‌است. اگر صفر باشد A و اگر یک باشد B را خروجی می‌دهد. ورودی enable هر کدام را هم به ground متصل می‌کنیم.

بررسی تراشه **register** ۷۴۱۷۵: این ثبات چهار بیت را ورودی می‌گیرد و با استفاده از لبه بالارونده ورودی کلاک، آن را در خود ذخیره می‌کند. ورودی **Master Reset (MR)** هم دارد که همه فلیپ‌فلاپ‌های داخل این تراشه را مستقل از کلاک و ورودی‌های دیتا ریست می‌کند.

طبق جدولی که در گزارش کار وجود دارد برای هر یک از هشت حالت M_i ‌ها یک عملیات طبق خواسته خودمان انجام می‌دهیم.

M_2	M_1	M_0	Operation
0	0	0	$A \leftarrow D_3 - D_0$
0	0	1	$B \leftarrow D_3 - D_0$
0	1	0	$A \leftarrow A$
0	1	1	$A \leftarrow B$
1	0	0	clear (A)
1	0	1	$ A \leftarrow \text{not}(A)$
1	1	0	$A \leftarrow \text{and}(A, B)$
1	1	1	$A \leftarrow \text{add}(A, B)$

۱- عملیات صورت گرفته در مدار بر حسب ورودی‌های $M_0 - M_2$

برای ساخت بخش کنترل باید از معنای هر کدام از ۱۶ حالت ورودی‌های سلکت **ALU** اطلاع داشته باشیم. جدول زیر که در دیتابیت آن موجود است، مشخص کرده است که به ازای هر حالت، چه عملیاتی انجام می‌شود.

Selection				Active-low inputs & outputs		Active-high inputs & outputs	
S3	S2	S1	S0	Logic (M = 1)	Arithmetic (M = 0) (Cn = 0)	Logic (M = 1)	Arithmetic (M = 0) (Cn = 1)
0	0	0	0	\bar{A}	$A \text{ minus } 1$	\bar{A}	A
0	0	0	1	\bar{AB}	$AB \text{ minus } 1$	$\bar{A} + B$	$A + B$
0	0	1	0	$\bar{A} + B$	$A\bar{B} \text{ minus } 1$	$\bar{A}\bar{B}$	$A + \bar{B}$
0	0	1	1	Logical 1	-1	Logical 0	-1
0	1	0	0	$\bar{A} + \bar{B}$	$A \text{ plus } (A + \bar{B})$	\bar{AB}	$A \text{ plus } (\bar{A}\bar{B})$
0	1	0	1	\bar{B}	$AB \text{ plus } (A + \bar{B})$	\bar{B}	$(A + B) \text{ plus } (A\bar{B})$
0	1	1	0	$\bar{A} \oplus \bar{B}$	$A \text{ minus } B \text{ minus } 1$	$A \oplus B$	$A \text{ minus } B$
0	1	1	1	$A + \bar{B}$	$A + \bar{B}$	\bar{AB}	$\bar{A}\bar{B} \text{ minus } 1$
1	0	0	0	\bar{AB}	$A \text{ plus } (A + B)$	$\bar{A} + B$	$A \text{ plus } AB$
1	0	0	1	$A \oplus B$	$A \text{ plus } B$	$\bar{A} \oplus \bar{B}$	$A \text{ plus } B \text{ plus } 1$
1	0	1	0	B	$A\bar{B} \text{ plus } (A + B)$	B	$(A + \bar{B}) \text{ plus } AB$
1	0	1	1	$A + B$	$A + B$	AB	$AB \text{ minus } 1$
1	1	0	0	Logical 0	$A \text{ plus } A$	Logical 1	$A \text{ plus } A$
1	1	0	1	\bar{AB}	$AB \text{ plus } A$	$A + \bar{B}$	$(A + B) \text{ plus } A$
1	1	1	0	AB	$A\bar{B} \text{ plus } A$	$A + B$	$(A + \bar{B}) \text{ plus } A$
1	1	1	1	A	A	A	$A \text{ minus } 1$

۲- جدول Mode Select -- Function

اکنون در هر کدام از هشت حالت جدول، حالت مطلوب و متناظر در جدول دیتابیت را پیدا می‌کنیم، و یک جدول درستی تشکیل می‌دهیم. تابع M و S_A, S_B را به دست بیاوریم.

Input			Output						
M ₂	M ₁	M ₀	S ₃	S ₂	S ₁	S ₀	S _A	S _B	M
.	.	.	x	x	x	x	.	1	x
.	.	1	1	1	1	1	1	.	x
.	1	.	1	1	1	1	1	1	1
.	1	1	1	.	1	.	1	1	1
1	1	1	1	1	1
1	.	1	1	1	1
1	1	.	1	.	1	1	1	1	1
1	1	1	1	.	.	1	1	1	.

3- جدول حالت و درستی $S_0 - S_3, S_A, S_B, M$ از روی $M_0 - M_2$

همانطور که در جدول فوق مشاهده می‌شود، لازم است توضیحاتی درمورد حالات منطقی S_A و S_B ارائه کنیم.

$S_A = 0$: یعنی بیت‌های دیتا وارد MUX اول بشوند.

$S_A = 1$: یعنی خروجی‌های ALU وارد MUX اول بشوند.

$S_B = 0$: یعنی بیت‌های دیتا وارد MUX دوم بشوند.

$S_B = 1$: یعنی محتویات رجیستر B وارد MUX دوم بشوند.

اکنون که جدول حالات را رسم کردایم، باید توابع هر کدام از S_i ها را به دست آوریم.

M ₂	M ₁ M ₀	..	+1	11	10
.	x	1	+	1	1
1	1	+	1	1	1

4- جدول کارنو

M ₂	M ₁ M ₀	..	+1	11	10
.	x	1	1	1	1
1	1	+	+	+	1

5- جدول کارنو

$$S_0 = M_2 M_1 + M'_0 + M'_2 M'_1 = M'_0 + (M_2 \oplus M_1)'$$

$$S_1 = M'_0 + M'_2 = (M_0 M_2)'$$

M ₂	M ₁ M ₀	..	+1	11	10
.	x	1	+	1	1
1	+	+	+	+	+

6- جدول کارنو

$$S_2 = M'_2 M'_1 + M'_2 M'_0 = (M_2 + M_1 M_0)'$$

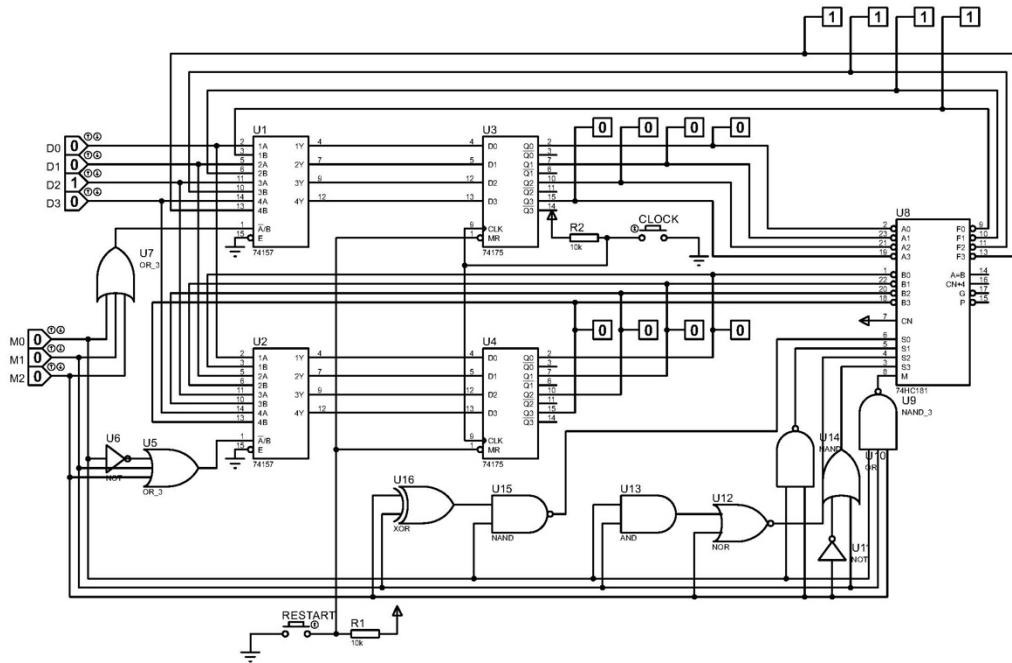
M ₂	M ₁ M ₀	..	+1	11	10
.	x	1	1	1	1
1	+	+	+	1	1

7- جدول کارنو

$$S_3 = M_1 + M_2'$$

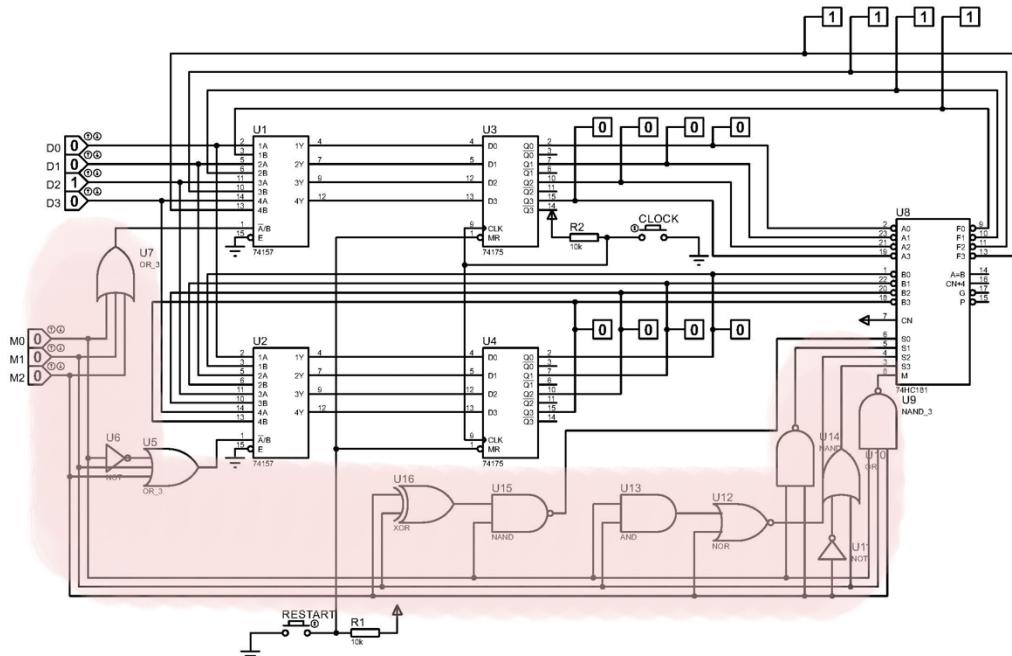
:: بخش عملی ::

مدار زیر در برنامه پروتئوس طراحی شده و در قسمت Export Graphics خروجی تصویری گرفته شده است.



-شکل نهایی مدار

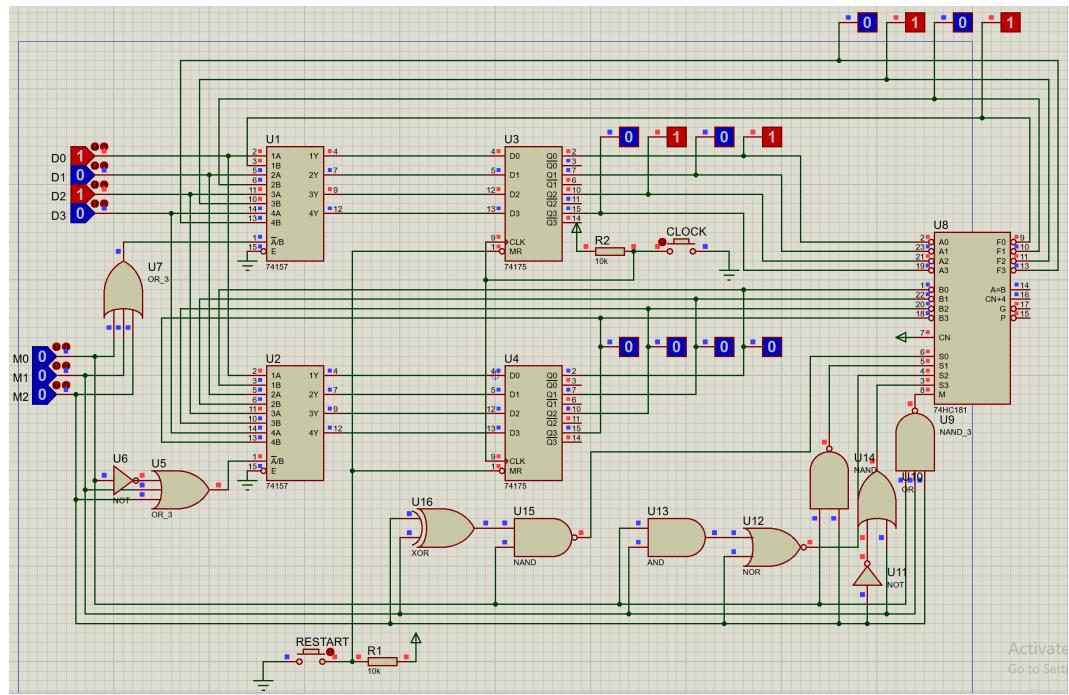
در مدار فوق ما برای ساختن بخش کنترل از توابع و جداول قبل استفاده کردہ‌ایم. جهت مشخص‌تر شدن آن به شکل زیر توجه کنید.



-مشخص کردن بخش کنترل در مدار

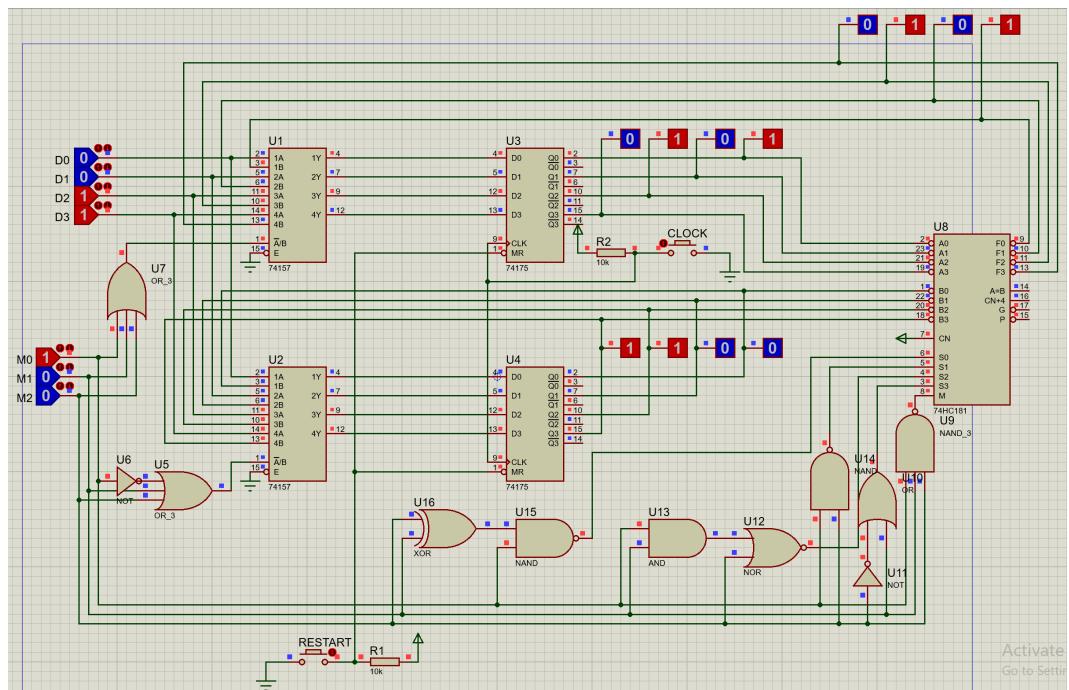
تست مدار: یک بار هم مدارمان را امتحان می‌کنیم.

$A \leftarrow D_0 - D_3 | M_2 M_1 M_0 = 000$ حالت •



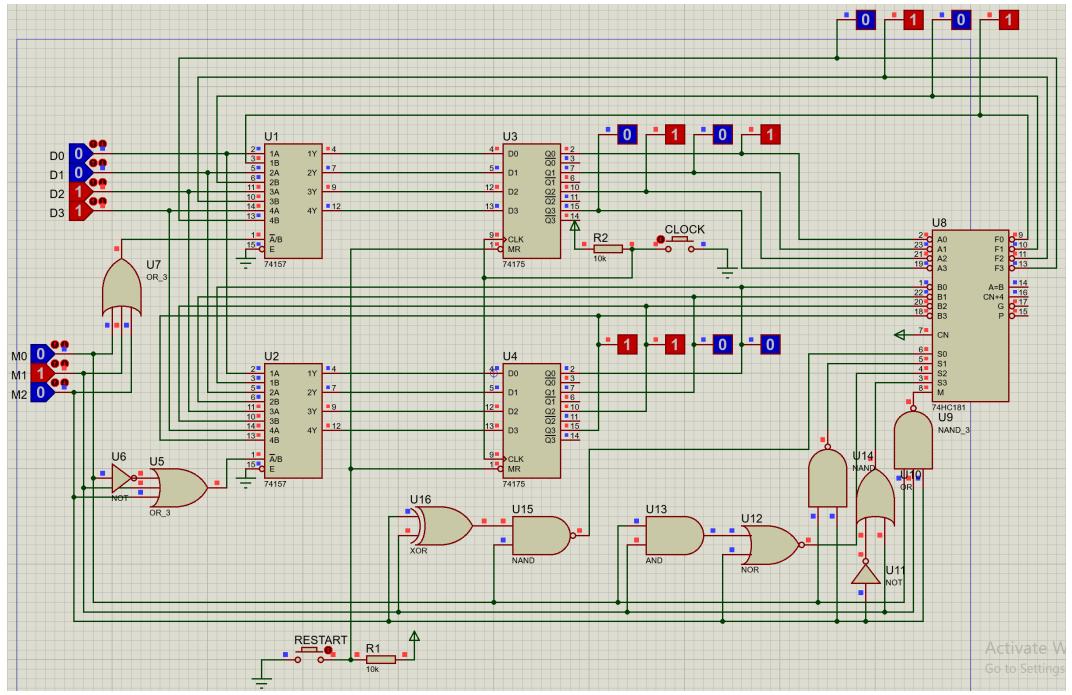
$M_2 M_1 M_0 = 000$ تست -۱۰

$B \leftarrow D_0 - D_3 | M_2 M_1 M_0 = 001$ حالت •



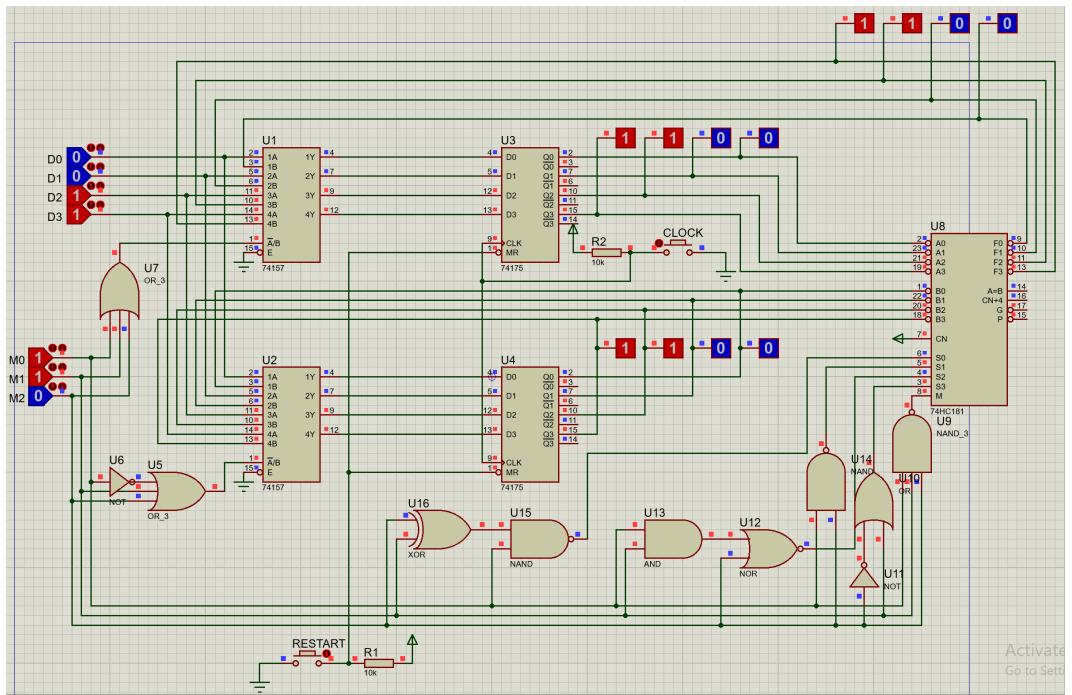
$M_2 M_1 M_0 = 001$ تست -۱۱

$A \leftarrow A | M_2 M_1 M_0 = 010$ حالت •



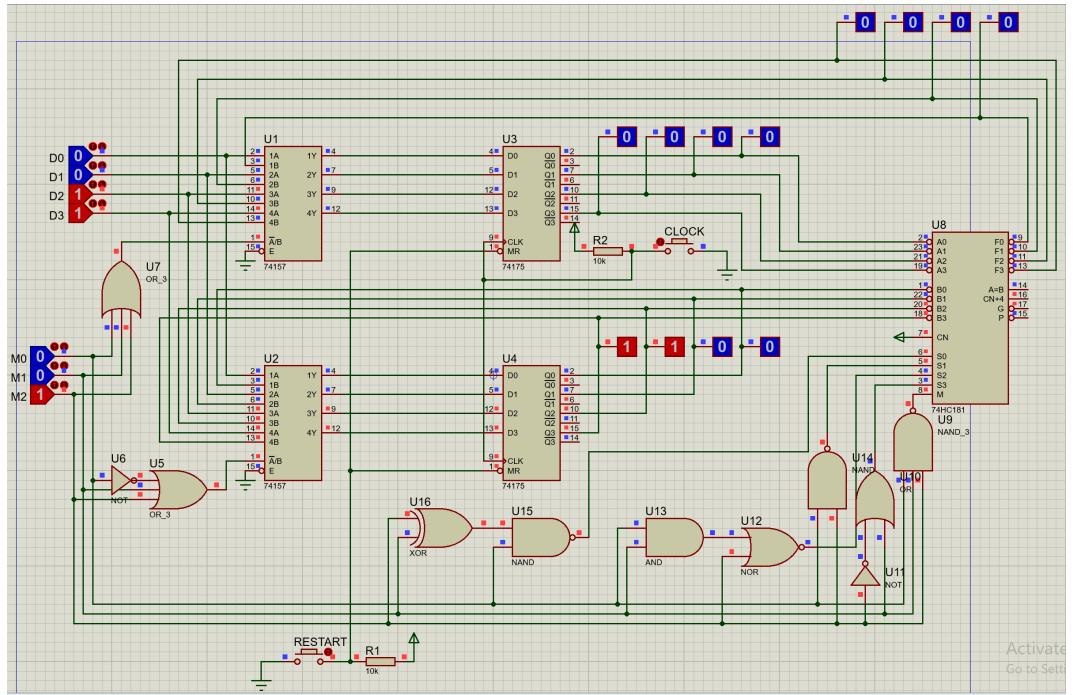
$M_2 M_1 M_0 = 010$ تست -۱۲

$A \leftarrow B | M_2 M_1 M_0 = 011$ حالت •



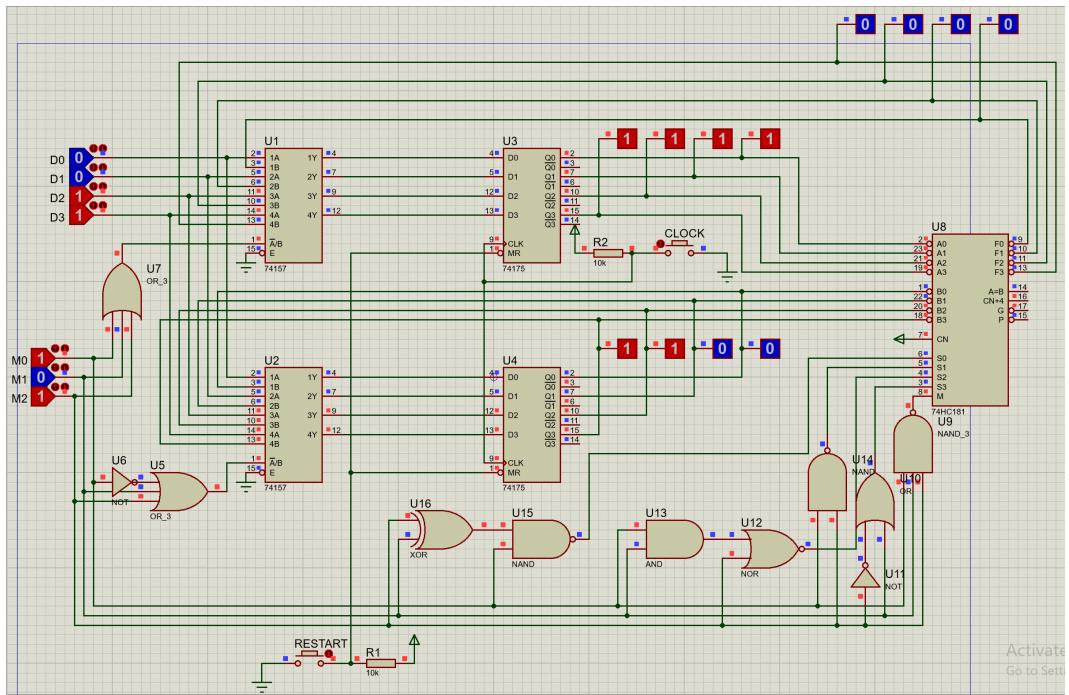
$M_2 M_1 M_0 = 011$ تست -۱۳

حالت A | $M_2 M_1 M_0 = 100$ •



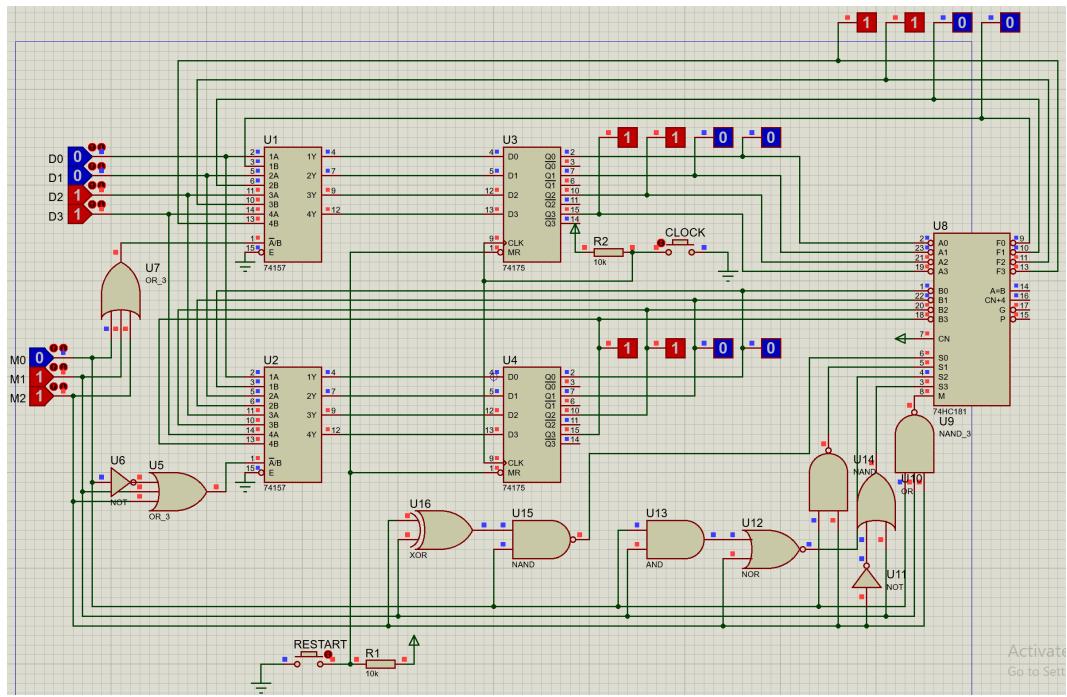
$M_2 M_1 M_0 = 100$ تست -۱۴

حالت $A \leftarrow \text{not}(A)$ | $M_2 M_1 M_0 = 101$ •



$M_2 M_1 M_0 = 101$ تست -۱۵

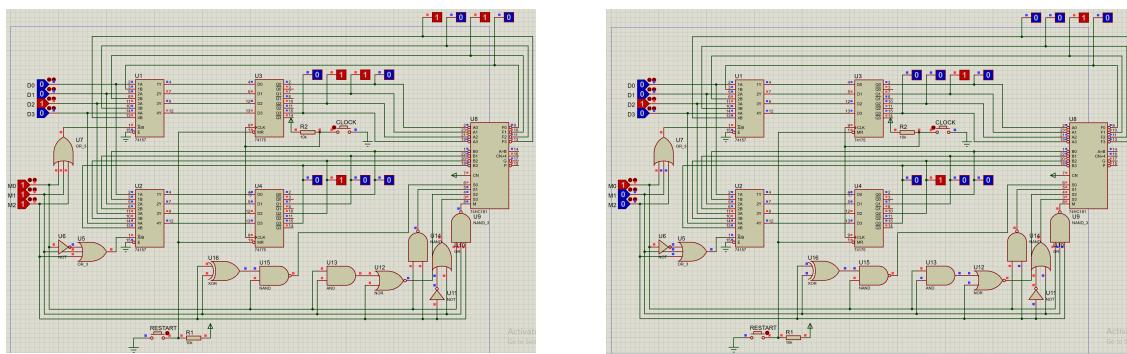
$$A \leftarrow \text{and}(A, B) \mid M_2 M_1 M_0 = 110 \quad \bullet$$



$$M_2 M_1 M_0 = 110 - \text{تست ۱۶}$$

$$A \leftarrow add(A, B) \mid M_2 M_1 M_0 = 111 \quad \bullet$$

ابتدا به A مقدار ۲ و به B مقدار ۴ را می‌دهیم تا بتوانیم حاصل جمعشان (عدد ۶) را به درستی داشته باشیم. اگر حاصل جمع دو عدد بیشتر از ۱۵ شود، یک کری خروجی (به عنوان عدد ۱۶) خواهد داشت که در ساز و کار مدار قابل مشاهده نیست ولی خودمان آن را در نظر می‌گیریم و در واقع یک بیت پنجم کنار چهار بیت که قابل مشاهده هستند در نظر می‌گیریم.



$$M_2 M_1 M_0 = 111 - \text{تست ۱۶}$$

$$M_2 M_1 M_0 = 111 - \text{تست ۱۷}$$

همانطور که مشاهده شد، همه حالت‌های مدار بررسی شدند و نتایج کاملاً درستی را نشان دادند.

ساخت مدار داخلی ALU

:: بخش تئوری ::

در این بخش قصد داریم که مدار داخلی یک واحد محاسبات ریاضی و منطقی یک ALU را بسازیم. در جدول زیر عملکرد مدار بر اساس شانزده حالت بیت‌های کنترل آمده است.

Operation Select					Operation	Function
S ₃	S ₂	S ₁	S ₀	C _{in}		
.	F = A	Transfer A
.	.	.	.	1	F = A + 1	Increment A
.	.	.	1	.	F = A + B	Addition
.	.	.	1	1	F = A + B + 1	Add with carry
.	.	1	.	.	F = A + B'	Subtract with borrow
.	.	1	.	1	F = A + B' + 1	Subtraction
.	.	1	1	.	F = A - 1	Decrement A
.	.	1	1	1	F = A	Transfer A
.	1	.	.	x	F = A \wedge B	AND
.	1	.	1	x	F = A \vee B	OR
.	1	1	.	x	F = A \oplus B	XOR
.	1	1	1	x	F = A'	Complement A
1	.	x	x	x	F = shr A	Shift right A to F
1	1	x	x	x	F = shl A	Shift left A to F

۱۹- جدول عملکرد ALU بر اساس حالات بیت‌های کنترلی

در ابتدا به کمک گیت‌های منطقی یک مدار ترکیبی با ۱۴ خروجی می‌سازیم. خروجی هر بخش زمانی که بیت‌های select و بیت C_{in} مطابق آن حالت تنظیم شده باشند یک می‌شود. در ادامه‌ی آن، خط مربوط به هر حالت را با نتیجه متناظر آن AND می‌کنیم.

1) Transfer A:

$$\text{control} = S_0'S_1'S_2'S_3'C' = \text{NOR}(S_0, S_1, S_2, S_3, C)$$

2) Increment A:

$$\text{control} = S_0'S_1'S_2'S_3'C = \text{NOR}(S_0, S_1, S_2, S_3, C')$$

3) Addition:

$$\text{control} = S_0S_1'S_2'S_3'C' = \text{NOR}(S_0', S_1, S_2, S_3, C)$$

4) Add with carry:

$$\text{control} = S_0 S_1' S_2' S_3' C = \text{NOR}(S_0', S_1, S_2, S_3, C')$$

5) Subtract with borrow:

$$\text{control} = S_0' S_1 S_2' S_3' C' = \text{NOR}(S_0, S_1', S_2, S_3, C)$$

6) Subtraction:

$$\text{control} = S_0' S_1 S_2' S_3' C = \text{NOR}(S_0, S_1', S_2, S_3, C')$$

7) Decrement A:

$$\text{control} = S_0 S_1 S_2' S_3' C' = \text{NOR}(S_0', S_1', S_2, S_3, C)$$

8) Transfer A:

$$\text{control} = S_0 S_1 S_2' S_3' C = \text{NOR}(S_0', S_1', S_2, S_3, C')$$

9) AND:

$$\text{control} = S_0' S_1' S_2 S_3' = \text{NOR}(S_0, S_1, S_2', S_3)$$

10) OR:

$$\text{control} = S_0 S_1' S_2 S_3' = \text{NOR}(S_0', S_1, S_2', S_3)$$

11) XOR:

$$\text{control} = S_0' S_1 S_2 S_3' = \text{NOR}(S_0, S_1', S_2, S_3)$$

12) Complement A:

$$\text{control} = S_0 S_1 S_2 S_3' = \text{NOR}(S_0', S_1', S_2', S_3)$$

13) Shift right A into F:

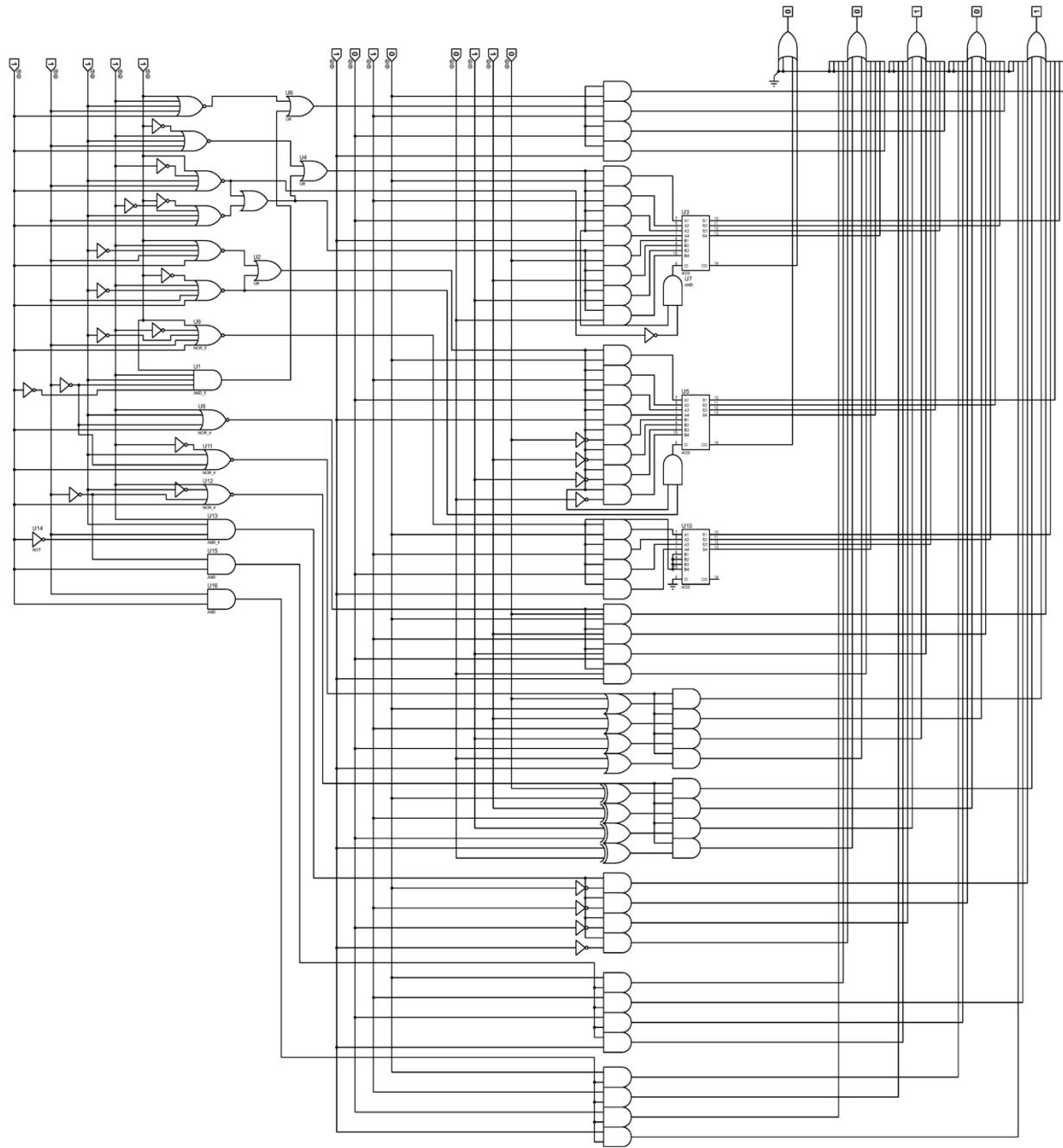
$$\text{control} = S_2' S_3 = \text{NOR}(S_2, S_3')$$

14) Shift left A into F:

$$\text{control} = S_2 S_3 = \text{NOR}(S_2', S_3')$$

:: بخش عملی ::

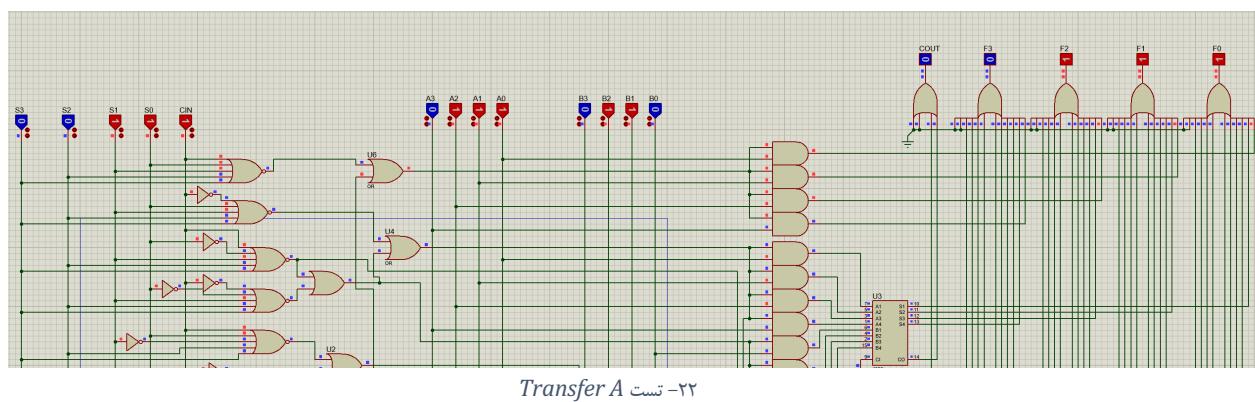
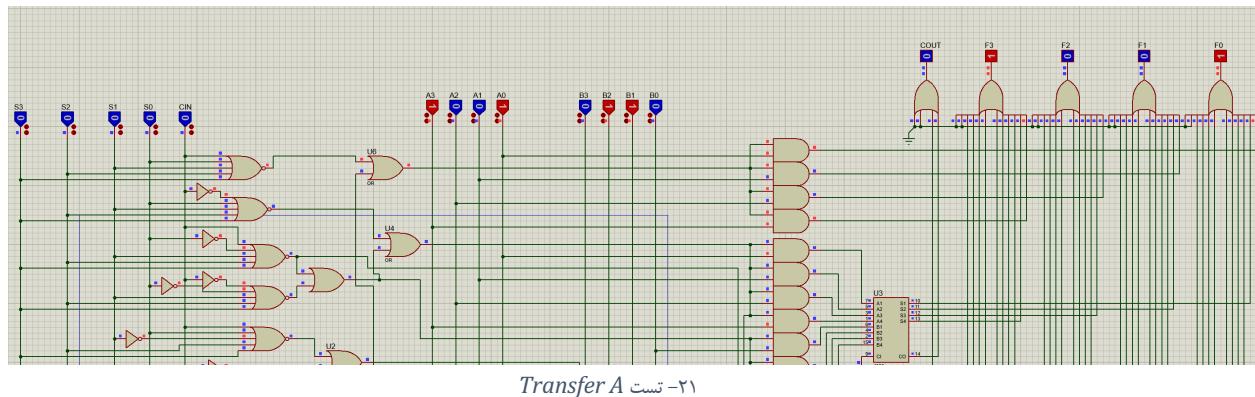
با توجه به توضیحات کاملی که داده شد، و توابعی که برای هر کدام از خطاهای کنترلی قابل مشاهده هستند، مدار زیر در برنامه پروتئوس طراحی شده و در قسمت Export Graphics خروجی تصویری گرفته شده است.



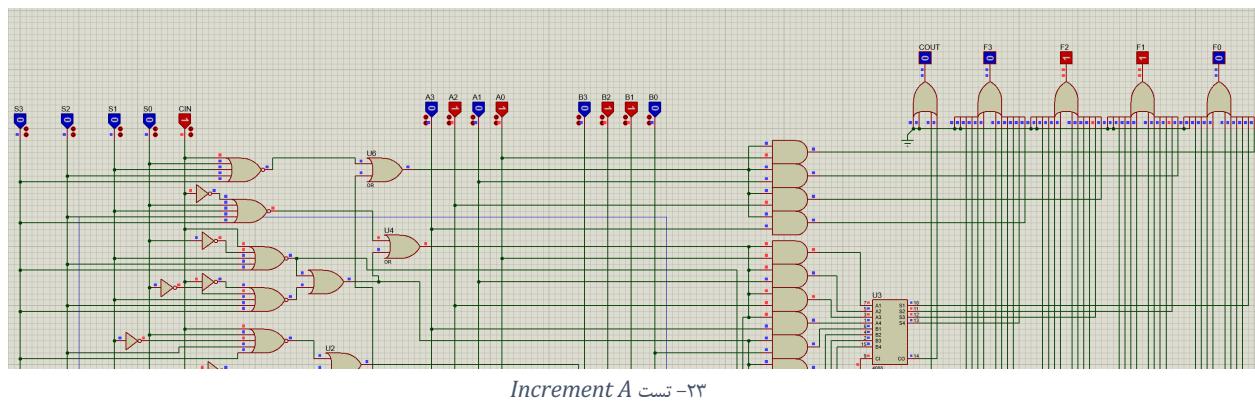
-۲۰- مدار داخلی ALU

تست مدار: یک بار هم مدارمان را برای همهی حالات امتحان می کنیم.

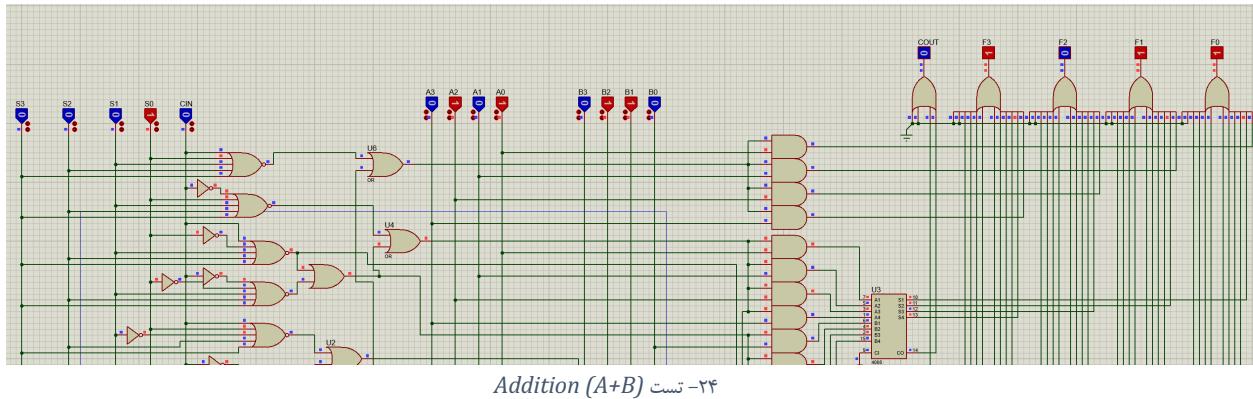
Transfer A | $C_{in} = 1$ و $S_3S_2S_1S_0 = 0011$ | $C_{in} = 0$ و $S_3S_2S_1S_0 = 0000$ •



Increment A | $C_{in} = 1$ و $S_3S_2S_1S_0 = 0000$ •

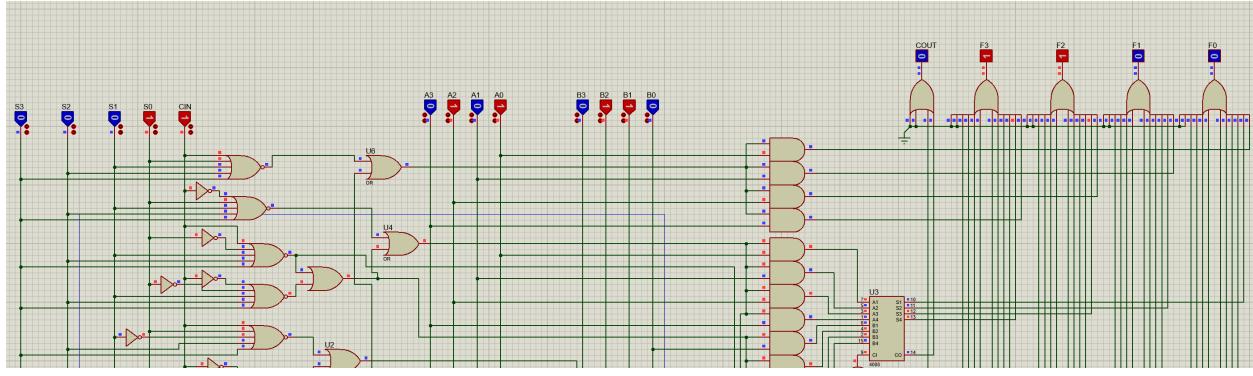


Addition | $C_{in} = 0$, $S_3S_2S_1S_0 = 0001$ حالت •



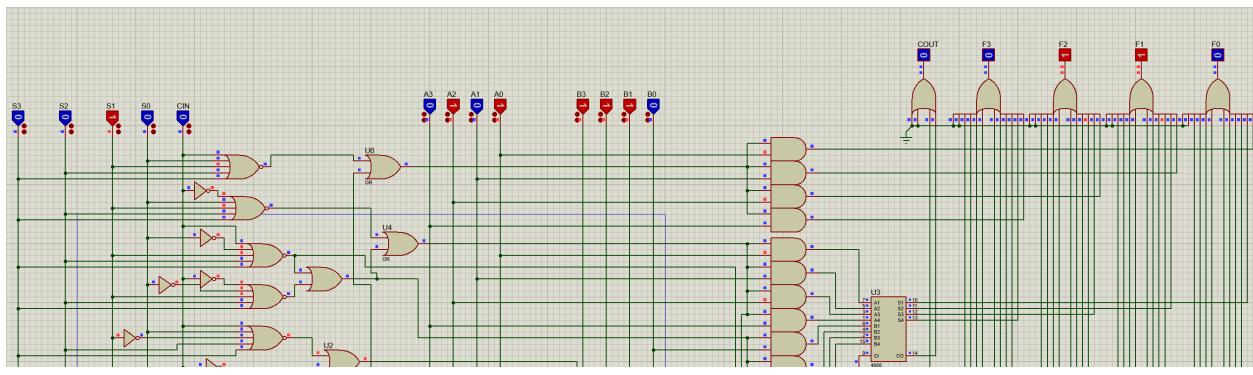
Addition (A+B) تست -٢٤

Addition with Carry | $C_{in} = 1$, $S_3S_2S_1S_0 = 0001$ حالت •



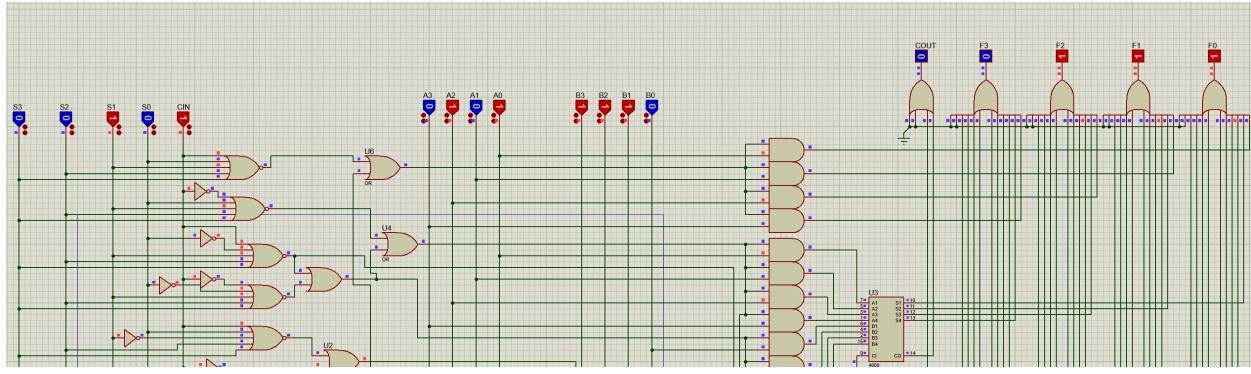
Addition with carry (A+B+1) تست -٢٥

Subtract with Borrow | $C_{in} = 0$, $S_3S_2S_1S_0 = 0010$ حالت •

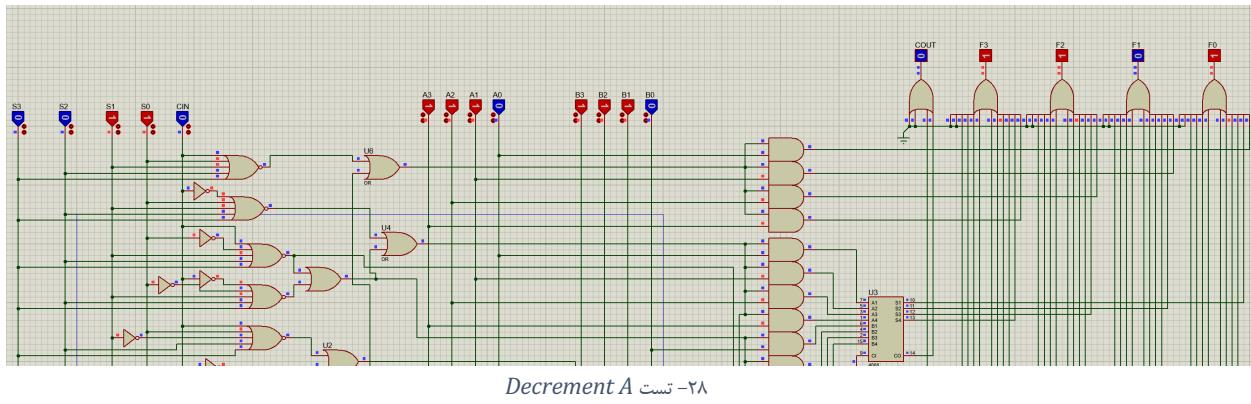


Subtract with borrow (A - B') تست -٢٦

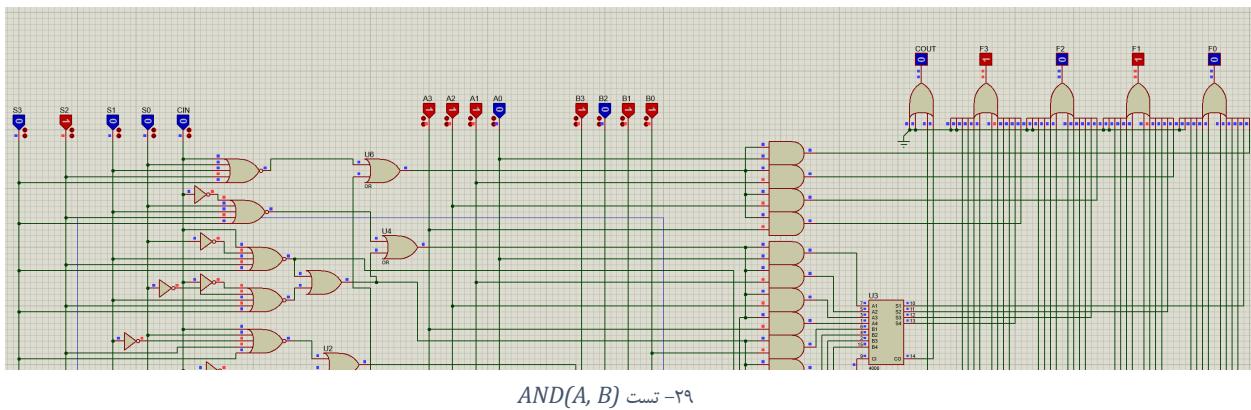
Subtraction | $C_{in} = 1$, $S_3S_2S_1S_0 = 0010$ حالت •



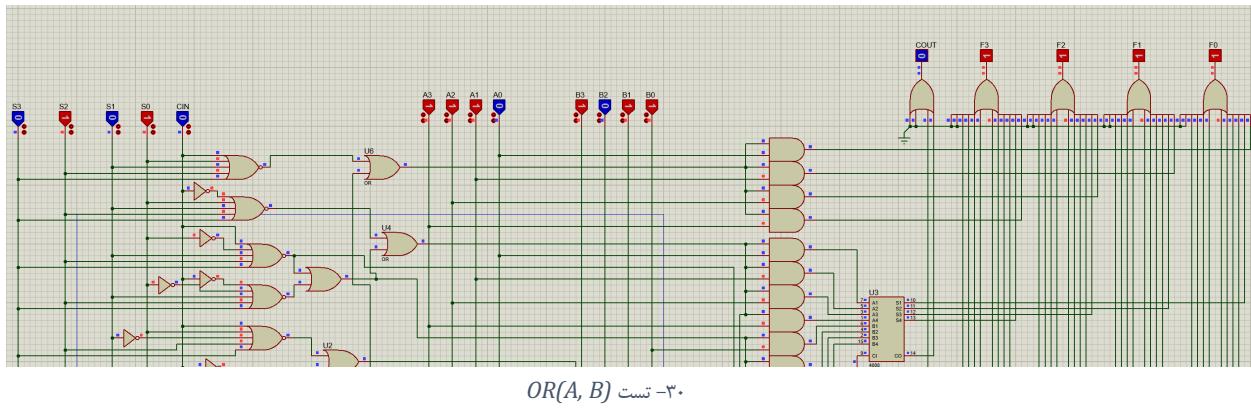
Decrement A | $C_{in} = 0$, $S_3S_2S_1S_0 = 0011$ حالت •



AND | $C_{in} = \times$, $S_3S_2S_1S_0 = 0100$ حالت •

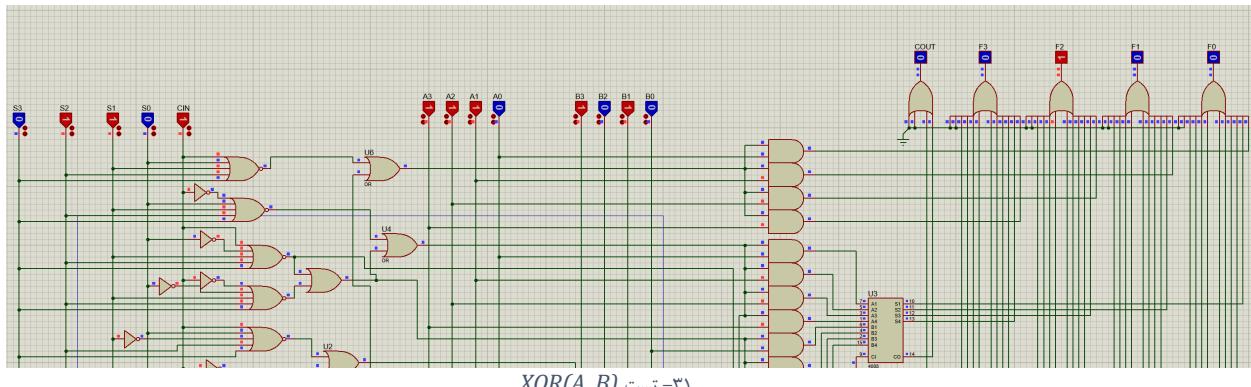


حالت OR | $C_{in} = \times$ ، $S_3S_2S_1S_0 = 0101$ •



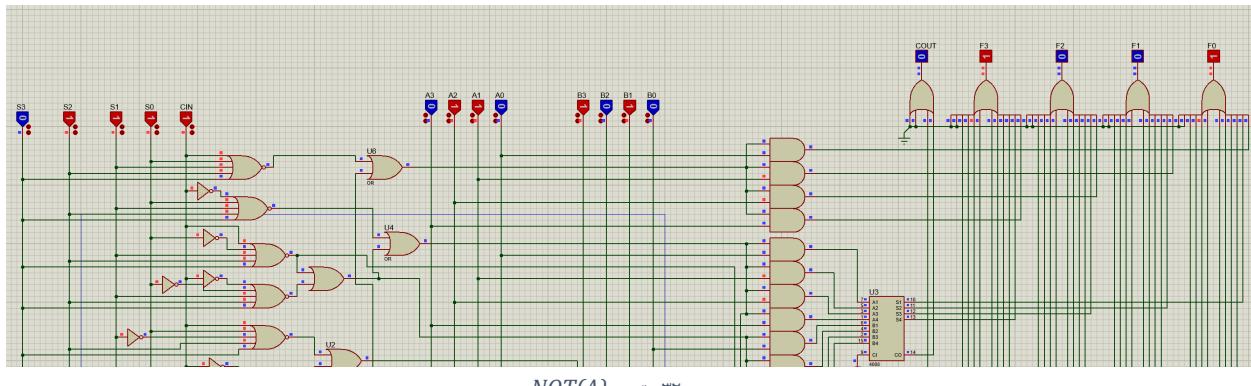
OR(A, B) تست -٣٠

حالت XOR | $C_{in} = \times$ ، $S_3S_2S_1S_0 = 0110$ •



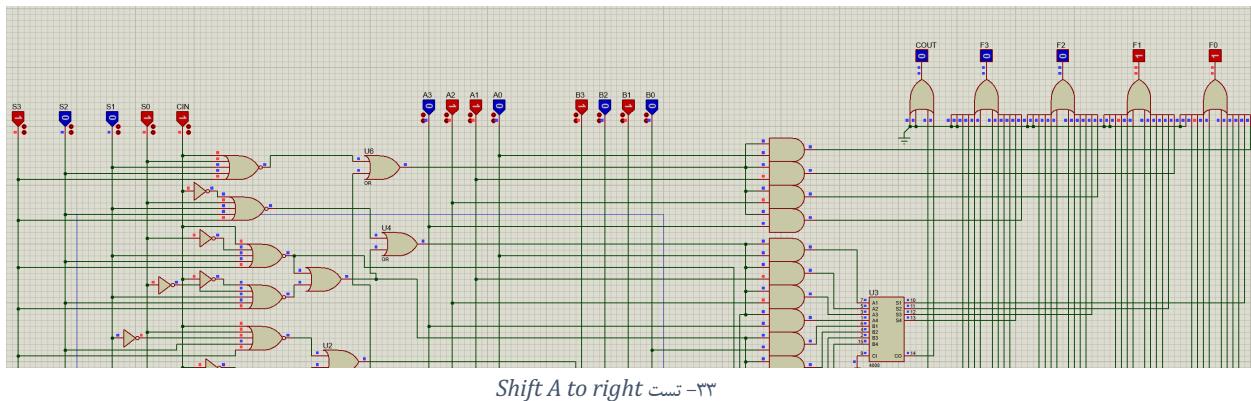
XOR(A, B) تست -٣١

حالت Complement A | $C_{in} = \times$ ، $S_3S_2S_1S_0 = 0111$ •

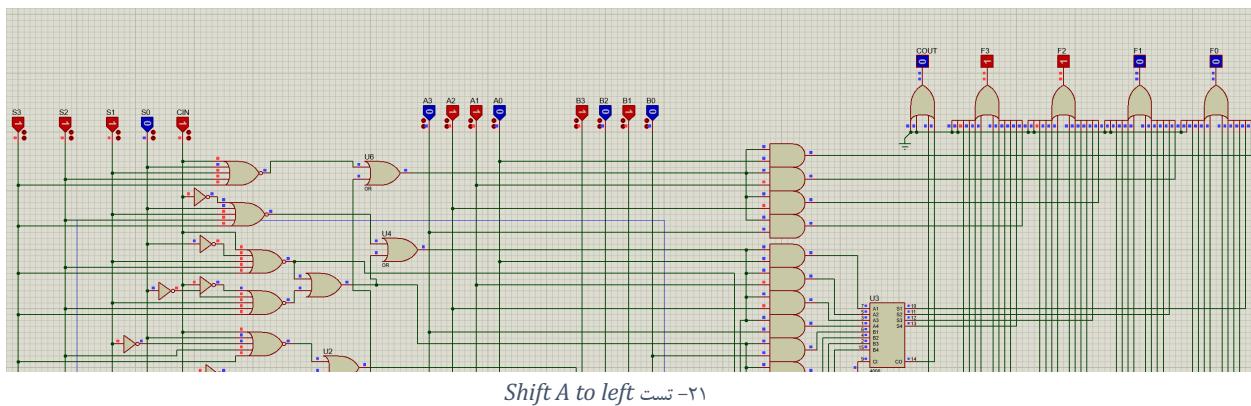


NOT(A) تست -٣٢

Shift A to right | $C_{in} = \times$ و $S_3S_2S_1S_0 = 10 \times \times$ •



Shift A to left | $C_{in} = \times$ و $S_3S_2S_1S_0 = 11 \times \times$ •



جمع‌بندی

در آزمایش پنجم درس آزمایشگاه مدارهای منطقی به آشنایی با استفاده از ALU پرداختیم. ابتدا یک مدار ساختیم که با استفاده یک تراشه 74181 (ALU)، تعدادی مالتیپلکسر و رجیستر می‌توانست هشت عملکرد مختلف که با سه بیت کنترلی (طبق دستور کار آزمایشگاه) تعیین می‌شدند را انجام دهد.

در بخش دوم، ما به درون ALU سفر کردیم و مدار داخلی آن را از صفر تا صد پیاده‌سازی کردیم. با توجه به حضور چهار بیت سلکت، در مجموع ۱۶ حالت ممکن بود که با در نظر گرفتن حالات کری ورودی، این حالات می‌توانست افزایش بیابد. اما از جایی به بعد در جدول مخصوص آن، تعداد زیادی don't care داشتیم که کار را راحت‌تر کرد.

ما در هر دو بخش این آزمایش، بخشی در مدار داشتیم که می‌توانیم از آن به عنوان بخش کنترلی یاد کنیم که شامل گیت‌های منطقی برای ایجاد خط‌های کنترل و تعیین عملکرد مدار است.

منابع

<https://datasheetspdf.com/pdf/493318/Philips/74181/1>