باسمه تعالی

آزمایشگاه معماری کامپیوتر

A blue and black logo

Description automatically generated

دانشکده مهندسی کامپیوتر دانشگاه صنعتی شریف

تابستان 1403

**استاد:**

دکتر حمید سربازی آزاد

مهندس عطیه غیبی فطرت

**اعضای گروه:**

سعید فراتی کاشانی – 401106299

امیرحسین صوری – 401106182

زهرا آذر – 99109744

فهرست عناوین

[موضوع آزمایش: 3](#_Toc171160957)

[شرح کلی آزمایش: 3](#_Toc171160958)

[پیاده‌سازی مدار اولیه در پروتئوس: 3](#_Toc171160959)

[طراحی **Full Adder** 3](#_Toc171160960)

[طراحی 4-**Bit Adder** 4](#_Toc171160961)

[طراحی **BCD Adder** 4](#_Toc171160962)

[طراحی 3-**Digit BCD Adder** 6](#_Toc171160963)

[نمایش خروجی 3-**Digit BCD Adder** 7](#_Toc171160964)

[پیاده‌سازی مدار به صورت فیزیکی: 8](#_Toc171160965)

[تراشه‌های مورد استفاده 8](#_Toc171160966)

[توضیحات مدار 9](#_Toc171160967)

موضوع آزمایش:

جمع‌کننده‌ی سه رقمی BCD

شرح کلی آزمایش:

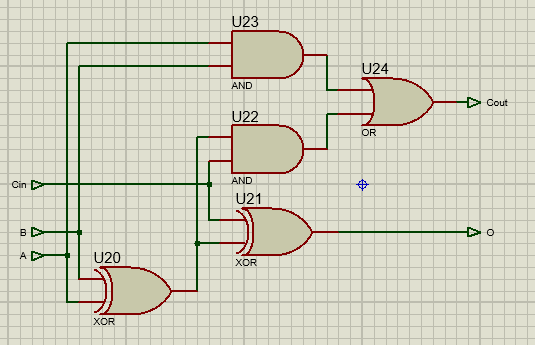
در این آزمایش می‌خواهیم دو عدد سه رقمی در مبنای 10 که به صورت BCD هستند را با یکدیگر جمع بزنیم و حاصل را برروی 7 Segment نمایش دهیم.

پیاده‌سازی مدار اولیه در پروتئوس:

برای این آزمایش در چند سطح مختلف، قطعات را طراحی کردیم و در ادامه هر بخش را توضیح می‌دهیم.

طراحی **Full Adder**

در گام اول یک تمام جمع‌کننده با استفاده از گیت‌های اولیه ساختیم تا بتوانیم در مراحل بعدی از آن استفاده کنیم.



شکل 1

طراحی 4-**Bit Adder**

در گام بعدی، با توجه به این‌که هر عدد BCD شامل 4 بیت است، 4 تمام جمع‌کننده را به طور سری به یکدیگر متصل کردیم تا بتوانیم دو عدد 4 بیتی را با یکدیگر جمع بزنیم.

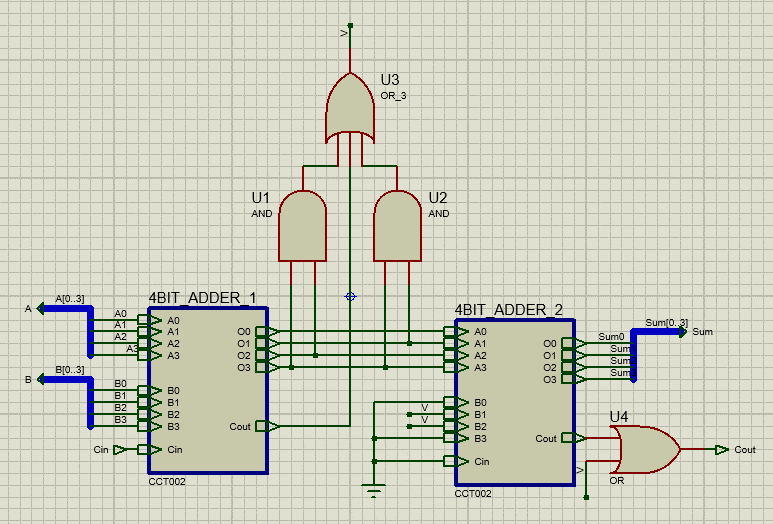
A diagram of a graph

Description automatically generated

شکل 2

طراحی **BCD Adder**

در گام بعدی، با استفاده از جمع‌کننده‌ی 4 بیتی که در مرحله قبل ساختیم، یک جمع کننده BCD را پیاده‌سازی کردیم. تفاوت این جمع‌کننده با مرحله‌ی قبلی در آن است که هر رقم خروجی ما حداکثر 9 می‌تواند باشد و ورودی‌هایمان نیز بیشتر از 9 نخواهند بود. لذا با مجموعی بزرگ‌تر از 18 روبه‌رو نخواهیم شد. مدار پیاده‌سازی شده به صورت زیر است:



شکل 3

همان‌طور که مشاهده می‌کنید، ابتدا C in و دو رقم BCD را با استفاده از Busهای A و B به جمع‌کننده‌ی 4 بیتی می‌دهیم. یک سیگنال تحت عنوان V تعریف کردیم که با استفاده از آن بتوانیم تشخیص دهیم که حاصل جمع، بیشتر از 9 شده‌است یا نه. برای این کار، یک بار بیت دوم و چهارم، و باری دیگر بیت سوم و چهارم را AND کردیم. زیرا در صورتی که حاصل بخواهد بالای 9 باشد، حاصل حداقل یکی از این دو گیت AND، 1 خواهد بود. لذا خروجی گیت‌های گفته شده را به همراه C out، از گیت OR عبور دادیم تا سیگنال V که مشابه overflow است آماده گردد.

حال، برای استفاده از این سیگنال V باید بدین صورت عمل کنیم که اگر V رخ داده بود، عدد حاصل را با 6 جمع کنیم و در صورتی که V رخ نداده بود، عدد حاصر بدون تغییر بماند و یا به عبارتی دیگر با 0 جمع شود. دلیل جمع کردن با 6 نیز، انتقال عدد از محدوده‌ی غیر مجاز BCD به محدوده‌ی مجاز آن است. پس به همین دلیل، بیت‌های دوم و سوم عدد B در جمع‌کننده‌ی دوم را برابر با V قرار می‌دهیم که بسته به نیاز، 0 یا 6 تشکیل شود. در نهایت نیز اگر V رخ داد، C out نیز داریم.

طراحی 3-**Digit BCD Adder**

با استفاده از BCD Adder که پیاده‌سازی کردیم، سه تا از آن‌ها را به صورت متوالی به یکدیگر متصل می‌کنیم تا یک جمع‌کننده‌ی سه رقمی ده‌دهی تشکیل دهیم.

A diagram of a computer

Description automatically generated

شکل 4

نمایش خروجی 3-**Digit BCD Adder**

در گام نهایی، با استفاده از قطعه‌ای که در مرحله قبل ساختیم و تعدادی 7 Segment، صحت عملکرد مدار جمع‌کننده‌مان را نمایش دادیم.

A computer screen shot of a circuit board

Description automatically generated

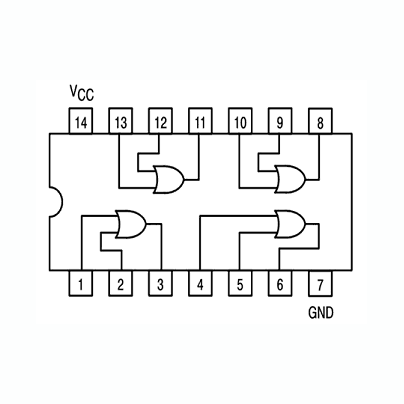
شکل 5

پیاده‌سازی مدار به صورت فیزیکی:

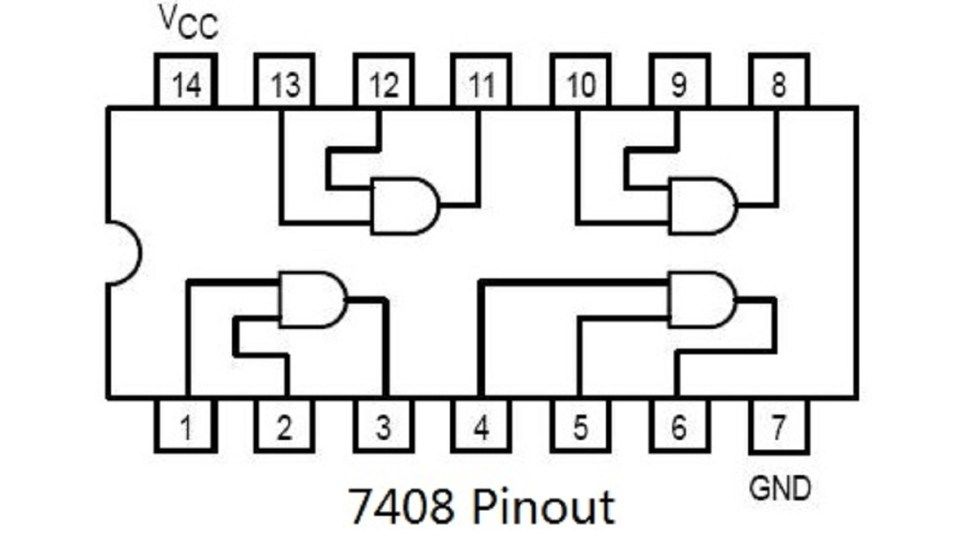
مداری که شبیه‌ساز آن را در پروتئوس پیاده‌سازی کردیم را بر روی بردبورد نیز به صورت فیزیکی آماده کردیم.

تراشه‌های مورد استفاده

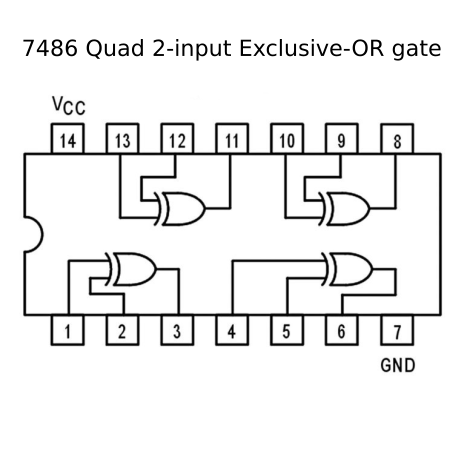
* 2-input OR: 7432



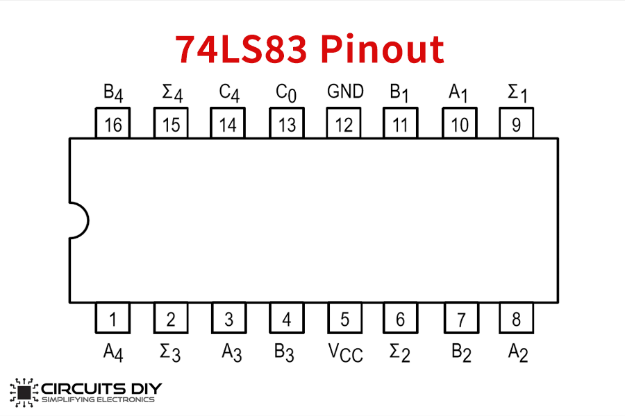
* 2-input AND: 7408



* 2-input XOR: 7486



* 4-bit full adder: 7483



توضیحات مدار

تصویر مدار پیاده‌سازی شده به صورت زیر است:

A circuit board with wires and lights

Description automatically generated

شکل 6

جهت سهولت در پیاده‌سازی مدار فیزیکی، به‌جای جمع زدن سه رقم BCD، صرفاً یک رقم را جمع زدیم. همچنین از آن‌جا که برای جمع زدن هر رقم BCD نیاز به دو تمام جمع‌کننده‌ی 4 بیتی داریم، یکی از آن‌ها را خودمان با استفاده از گیت‌های اولیه ساختیم و برای دیگری از تراشه آماده‌ی جمع‌کننده استفاده کردیم.

در بردبورد سمت چپ، چهار تا تمام‌جمع‌کننده‌ی یک بیتی ساختیم. از آن‌جا که برای هر یک تمام جمع‌کننده، نیاز به 2 گیت XOR، 2 گیت AND و 1 گیت OR داریم پس چینش تراشه‌ها را به صورت بالا انجام دادیم تا استفاده بهینه از تمامی 4 گیت موجود در هر چهار تراشه صورت بگیرد.

در بردبورد سمت راست نیز نیمه‌ی راست مدار پیاده‌سازی شده در پروتئوس را آماده کردیم. 4 بیت خروجی جمع را با استفاده از 4 عدد LED و همچنین بیت C out که معادل با 10 است را نیز با LED دیگری همان‌طور که در تصویر مشخص شده‌است، نمایش دادیم.