

# آزمایشگاه معماری کامپیوتر

دانشکده مهندسی کامپیوتر

دکتر سربازی  
تابستان ۱۴۰۳

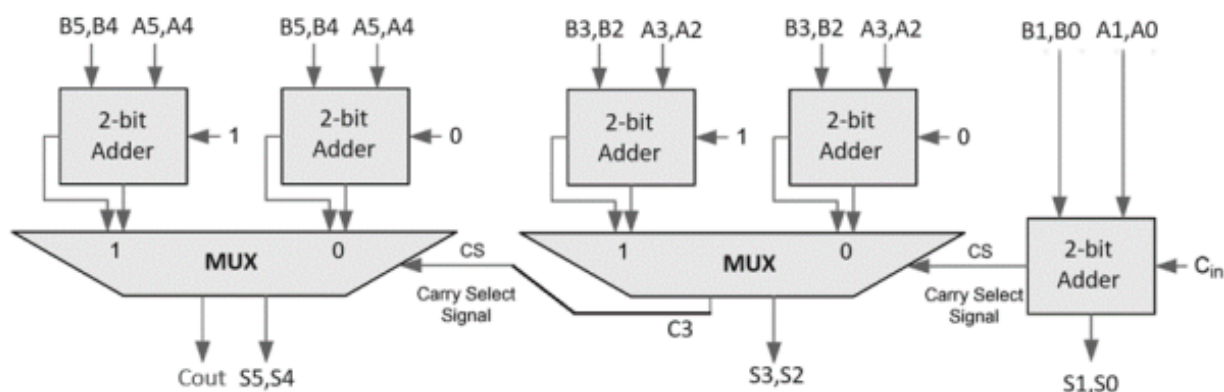
رادرین چراغی، ۴۰۱۱۰۵۸۱۵  
مبین پورعابدینی، ۴۰۱۱۱۰۵۵۶  
آرین نوری، ۴۰۱۱۰۶۶۶۳



## گزارش آزمایش دوم

### هدف

در این آزمایش سعی داریم به منظور بهبود سرعت عمل جمع، یک جمع کننده ۶ بیتی با انتخاب رقم نقلی (Carry Select Adder) بسازیم. طبق شکل ۲ ورودی‌ها شامل  $A[0,5]$ ،  $B[0,5]$  و  $C_{in}$  و همچنین خروجی‌ها شامل  $S[0,5]$  و  $C_{out}$  می‌باشند.



شکل ۲. ضرب کننده ۶ بیتی با انتخاب رقم نقلی (Carry Select Adder)

Carry Select Adder میتواند با تاخیر بسیار کمتری نسبت به Ripple Carry Adder جمع دو عدد را حساب کند. دلیل این اختلاف سرعت در نحوه عملکرد Carry Select Adder نهفته است.

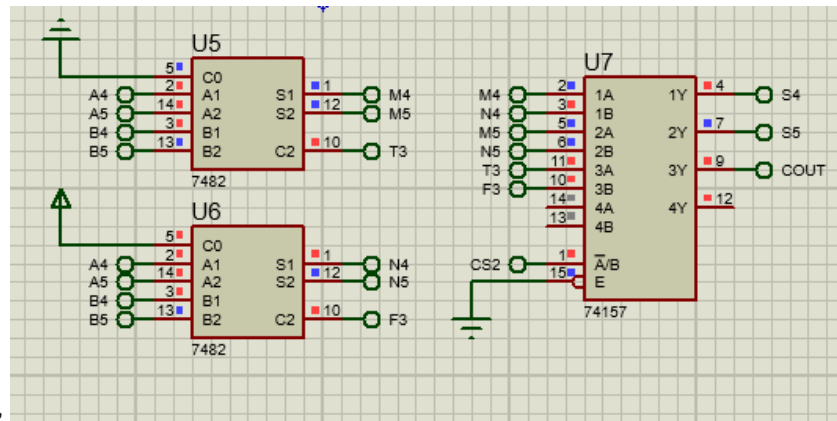
### ۱ نحوه عملکرد:

در این نوع جمع کننده تعداد بیت های ورودی به تعدادی بازه افزا می شوند مثلا در یک جمع کننده Carry Select Adder ۱۶ بیتی ممکن است بیت ها به ۴ قسمت ۴ بیتی تقسیم شوند برای ۴ بیت اول مقدار Carry مشخص است و میتوانیم آن را وارد به Adder ۴ بیتی کنیم اما برای ۴ بیتی های دوم، سوم و چهارم مقدار Carry نامشخص است

ایده اصلی این معماری در این قسمت است که برای ۳ قسمت ۴ بیتی دیگر به ازای  $C_i = 0$ ،  $C_i = 1$  جواب آن قسمت را حساب میکنیم و بعد از مشخص شدن جواب مرحله قبلی با استفاده از یک Multiplexer بین دو حالت  $C_i = 0$ ،  $C_i = 1$  مورد درست را انتخاب میکنیم و به همین ترتیب تاخیر ای که به ازای هر قسمت اضافه میشود صرفا برابر مقدار تاخیر Multiplexer خواهد بود.

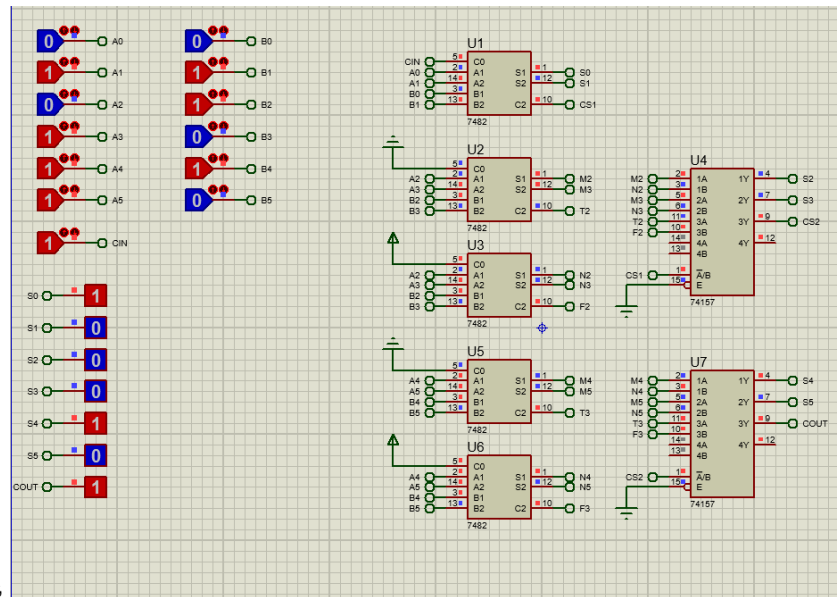


همچنین خروجی  $CS_2$  Multiplexer قسمت بالا را به عنوان ورودی  $Select$  Multiplexer بعدی وصل میکنیم



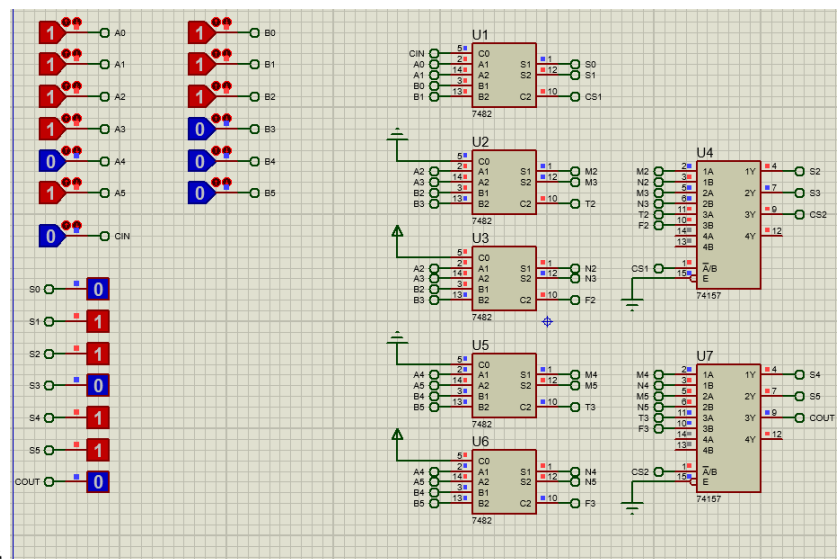
شکل (۴.۲.۲)

## ۳.۲ نتایج پیاده سازی



شکل (۱.۳.۲)

در این تست اعداد  $A = ۵۸$  و  $B = ۲۲$  با  $CIN = ۱$  جمع میشوند و مقدار خروجی آنها برابر ۸۱ میشود و جمع کننده مان به درستی عمل میکند

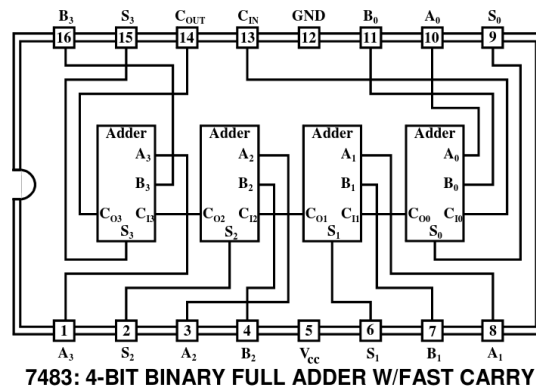


شکل (۲.۳.۲)

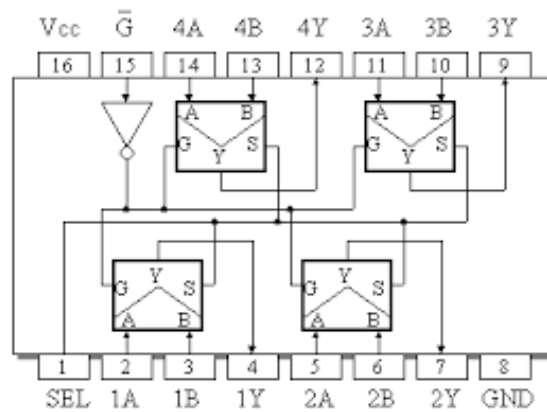
در این تست اعداد  $A = ۴۷$  و  $B = ۷$  با  $CIN = ۰$  جمع میشوند و مقدار خروجی آنها برابر ۵۴ میشود و جمع کننده مان به درستی عمل میکند

### ۳ پیاده سازی در آزمایشگاه

در این قسمت نیز همانند قبل ابتدا یک ۴-bit Adder را روی برد متصل کرده و با LED آن را چک میکنیم. همچنین برای جمع کردن ۲ بیت بعدی از ۷۴۸۳ استفاده میکنیم (به تعداد کافی ۷۴۸۲ موجود نبود) همچنین یک Multiplexer قرار میدهیم تا از بین  $C_1 = 1, C_1 = 0$  بر اساس خروجی Adder قسمت قبل خروجی مناسب را انتخاب کند به این منظور ابتدا Datasheet قطعه های ۷۴۸۳، ۷۴۱۵۷ را بررسی میکنیم.

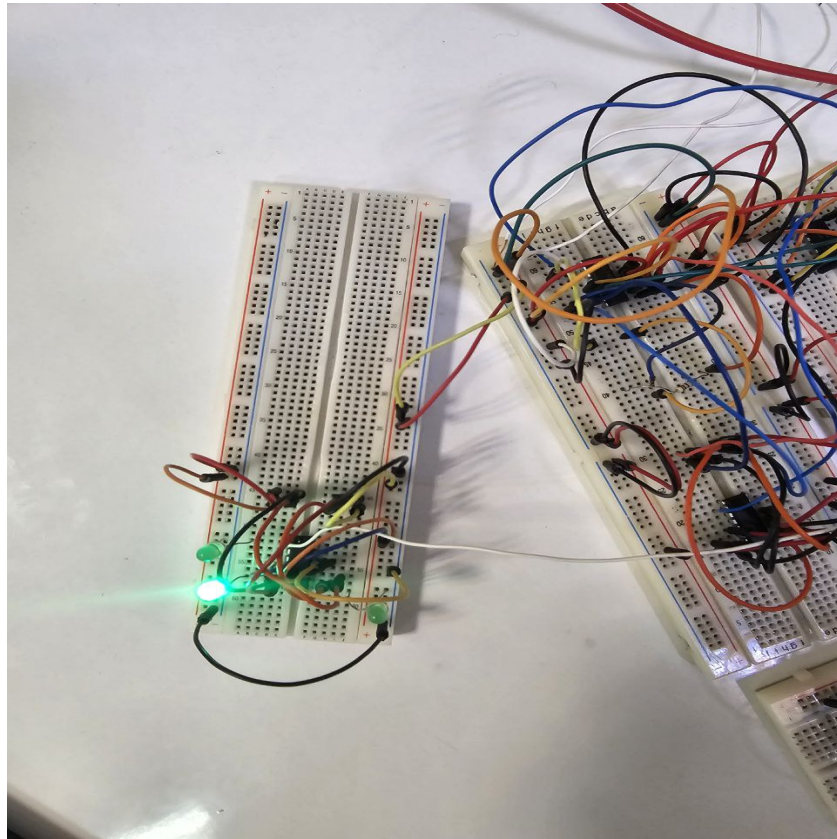


شکل (۱.۳)



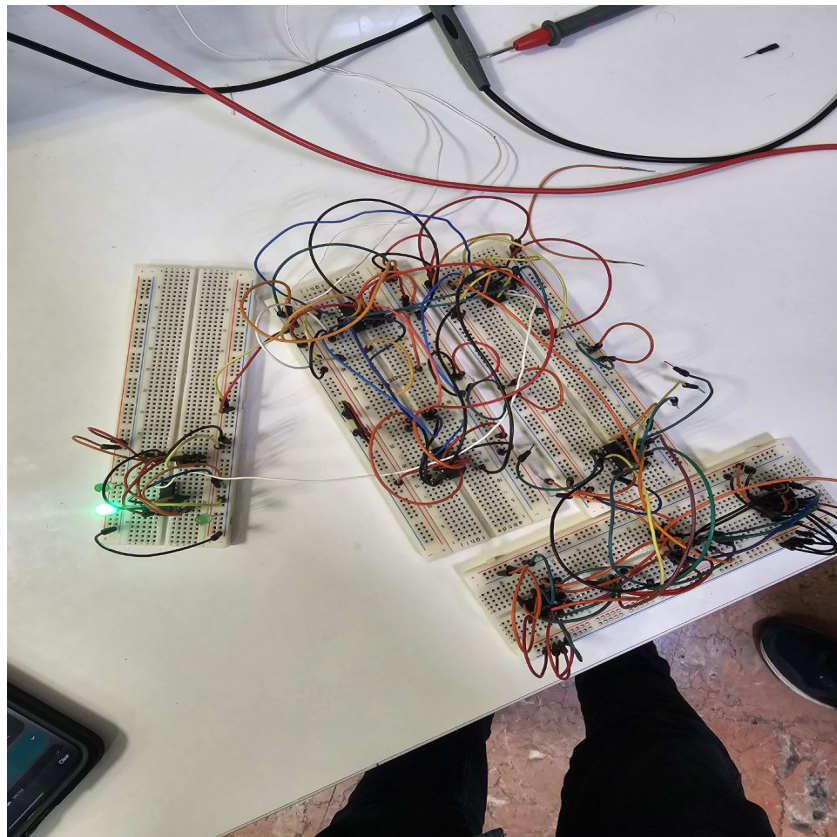
شکل (۲.۳)

حال با توجه به Datasheet ها و با استفاده از طراحی پروتئوس مان مدار مد نظر را روی برد پیاده سازی میکنیم



شکل (۳.۳)

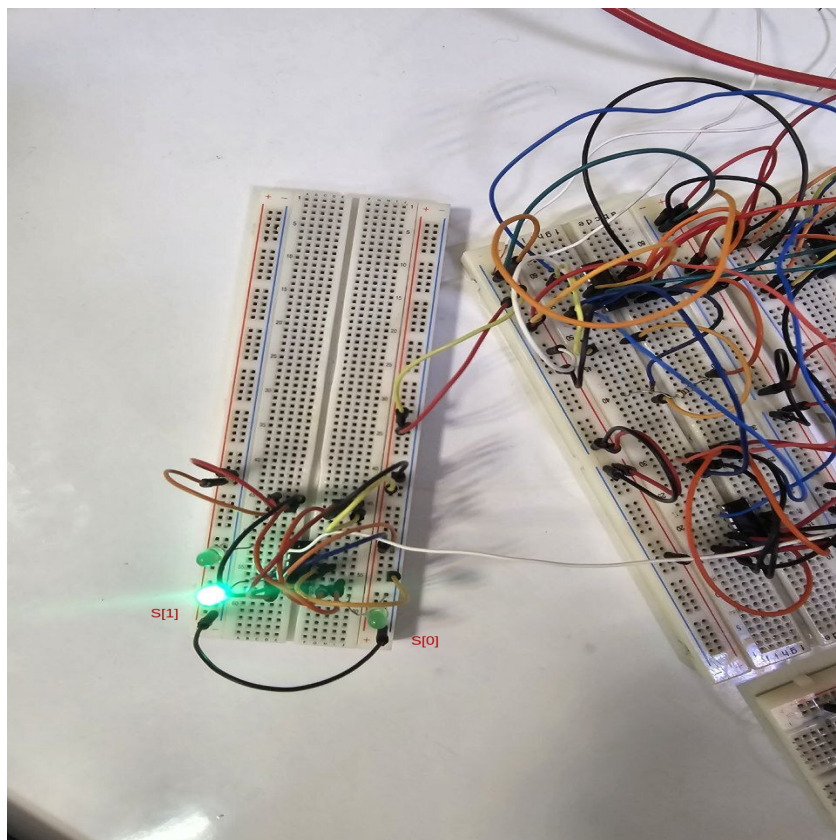
همچنین برای جمع کردن ۲ بیت بعدی نیز از ۲ قطعه ۷۴۸۳ دیگر استفاده میکنیم و مشابه بالا با یک Multiplexer از بین  $C_2 = 1$ ,  $C_2 = 0$  بر اساس خروجی Adder دوم خروجی مناسب را انتخاب کند



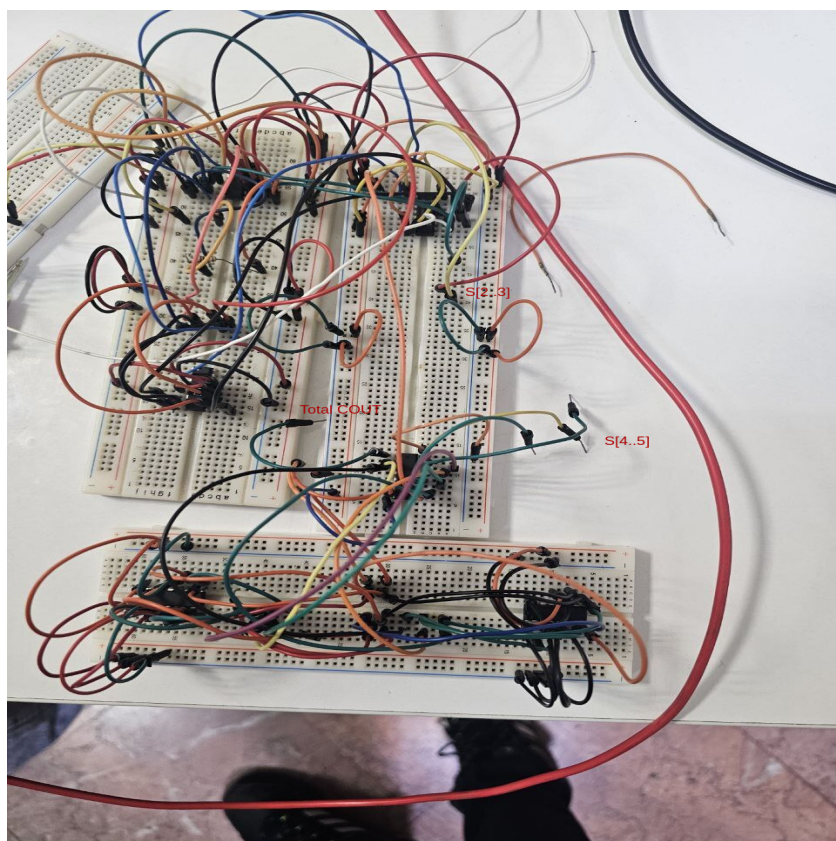
شکل (۴.۳)



برای بررسی صحت عملکرد پیاده سازی انجام شده سیم های خروجی هر قسمت را با ولت سنج بررسی میکنیم و چک میکنیم که با توجه به ورودی داده شده مقادیر خروجی درست باشند ، ورودی های مدار را نیز با استفاده از خطوط مثبت و منفی برد مورد مقدار دهی میکنیم.  
در زیر خروجی های مدار را مشخص کرده ایم:



شکل (۵.۳)



شکل (۶.۳)