دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

عنوان:

Carry Select Adder

نام درس

آزمایشگاه معماری کامپیوتر

نیم‌سال ۱۴۰۴۳

استاد

دکتر سربازی آزاد

گروه ۲

امیرشایان سلیمانی‌نیا - ۴۰۳۱۰۶۱۰۸
محمدمهدی مرادی - ۴۰۳۱۰۶۶۸۱

فهرست مطالب

۱ **مقدمه و هدف آزمایش** ۲

۲ **شرح روش آزمایش** ۲

۳ **شرح ساختار مدار** ۲
۱.۳ بخش ۱ مدار ۲
۲.۳ بخش ۲ مدار ۳
۳.۳ بخش ۳ مدار ۳
۴.۳ عکس کامل مدار ۴

۴ **نکات جزئی** ۵

۵ **تست مدار** ۵
۱.۵ تست ۱ . . . . . ۵
۲.۵ تست ۲ . . . . . ۶
۳.۵ تست ۳ . . . . . ۶
۴.۵ تست ۴ . . . . . ۷

۶ **نتیجه‌گیری** ۷

۱ مقدمه و هدف آزمایش

هدف از انجام این آزمایش، طراحی، پیاده‌سازی و تست یک مدار جمع‌کننده ۶ بیتی بر اساس معماری Carry-Select است. در این آزمایش قصد داریم به صورت عملی بررسی کنیم که چگونه می‌توان وابستگی‌های متوالی در تولید رقم نقلی را که عامل اصلی کندی در جمع‌کننده‌های معمولی است، برطرف کرد. در طول این گزارش خواهیم دید که این معماری چگونه با بهره‌گیری از پردازش موازی و استفاده از مالتی‌پلکسرها، زمان تاخیر کل مدار را کاهش می‌دهد.

۲ شرح روش آزمایش

در روش Carry-Ripple خروجی هر طبقه از مدار وابسته به این است که خروجی بخش قبل تولید شده باشد؛ یا به بیانی دیگر، برای کری ورودی هر بخش نیاز به آماده بودن کری خروجی بخش قبل داریم. به همین دلیل این وابستگی باعث می‌شود تاخیر مدار افزایش پیدا کند.

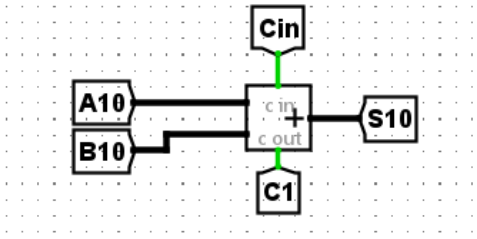
برای کاهش این تاخیر و عدم وابستگی به طبقه قبلی، از روش Carry-Select استفاده می‌کنیم. در این روش در هر طبقه ما هر دو خروجی را تولید می‌کنیم (هم با فرض $Carry = 0$ و هم با فرض $Carry = 1$). به بیانی دیگر، بدون نیاز به دانستن کری مرحله قبلی، خروجی تولید می‌شود و آماده نگه داشته می‌شود. حالا با استفاده از ۲ عدد مالتی‌پلکسر برای هر طبقه، که بیت سلکت آن کری مرحله قبل است، خروجی هر طبقه را پیدا می‌کنیم. یکی از مالتی‌پلکسرها برای انتخاب کری و یکی هم برای انتخاب حاصل جمع ۲ بیتی در هر مرحله استفاده می‌شود.

۳ شرح ساختار مدار

به طور کلی در مدار ۳ طبقه یا بخش داریم که هر طبقه یک جمع ۲ بیتی را انجام می‌دهد.

۱.۳ بخش ۱ مدار

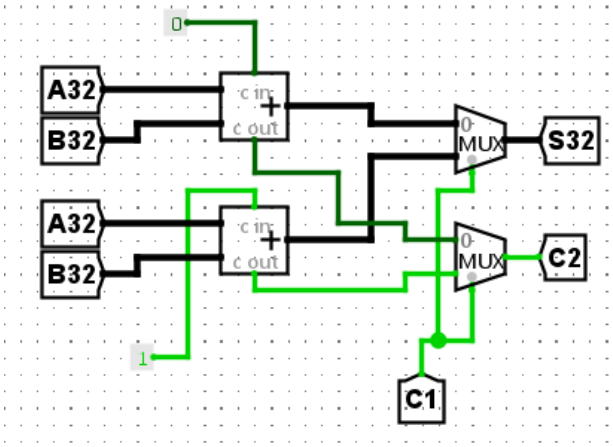
طبقه اول با انجام دادن جمع ۲ بیتی بین بیت‌های کم‌ارزش A و B، ۲ بیت کم‌ارزش خروجی نهایی (S_0, S_1) را همراه با یک بیت کری (C_1) تولید می‌کند؛ به طوری که بیت کری تولید شده در این بخش، در طبقه بعدی به عنوان سلکت مالتی‌پلکسر استفاده می‌شود.



شکل ۱: بخش اول مدار

۲.۳ بخش ۲ مدار

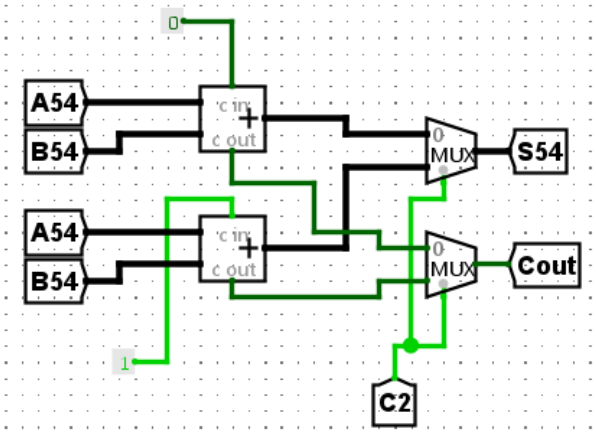
طبقه دوم، جمع ۲ بیتی دیگری را ($A_3A_2 + B_3B_2$) یک‌بار با کری ورودی ۱ و بار دیگر با کری ۰ انجام می‌دهد که خروجی این ۲ جمع‌کننده به عنوان ورودی‌های مالتی‌پلکسری استفاده می‌شود که بیت سلکت آن C_1 است. به طوری که اگر $C_1 = 1$ باشد، آن مقداری که توسط جمع‌کننده با کری ورودی ۱ تولید شده بود، انتخاب می‌شود و در نهایت خروجی این بخش، ۲ بیت وسط نهایی (S_2, S_3) را تولید می‌کند. بیت کری خروجی این بخش نیز با استفاده از مالتی‌پلکسر انتخاب می‌شود؛ به بیانی دیگر مانند حاصل جمع، خروجی کری هر ۲ حالت را (یعنی یک‌بار با کری ۱ و یک‌بار با کری ۰) به مالتی‌پلکسر می‌دهیم و C_1 را به عنوان بیت سلکت مالتی‌پلکسر قرار می‌دهیم تا کری خروجی را انتخاب کند، به همین ترتیب C_2 تولید می‌شود.



شکل ۲: بخش دوم مدار

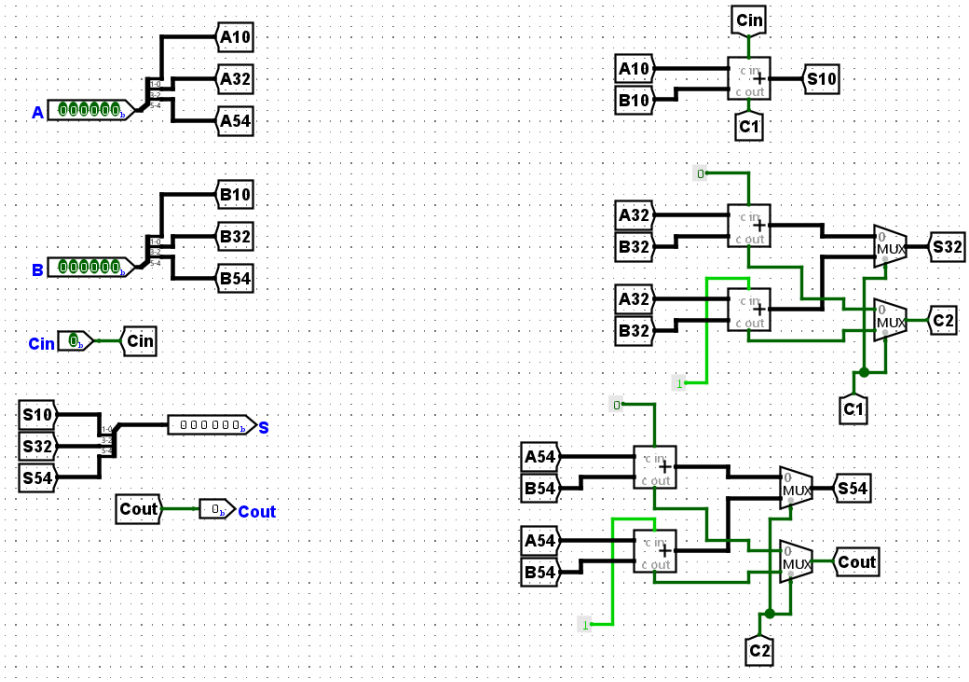
۳.۳ بخش ۳ مدار

طبقه سوم هم مانند طبقه دوم می‌باشد، به طوری که جمع $A_5A_4 + B_5B_4$ را حساب می‌کند (باز هم یک‌بار با کری ۱ و یک‌بار با کری ۰). به همین ترتیب خروجی را با استفاده از ۲ مالتی‌پلکسر که بیت سلکت هر دوی آن‌ها رقم کری بخش قبلی یعنی C_2 است، حساب می‌کند. اگر $C_2 = 1$ باشد، خروجی هر دو مالتی‌پلکسر از ورودی ۱ آن‌ها می‌آید؛ یعنی خروجی‌های جمع‌کننده‌ای که با کری ورودی ۱ محاسبات را انجام داده بود، به عنوان خروجی انتخاب می‌شوند. در نهایت خروجی این بخش، ۲ بیت ارزشمند خروجی اصلی مدار را تشکیل می‌دهد (S_4, S_5). بیت کری خروجی نیز، کری خروجی نهایی را تشکیل می‌دهد (C_{out}).



شکل ۳: بخش سوم مدار

۴.۳ عکس کامل مدار



شکل ۴: عکس کل مدار

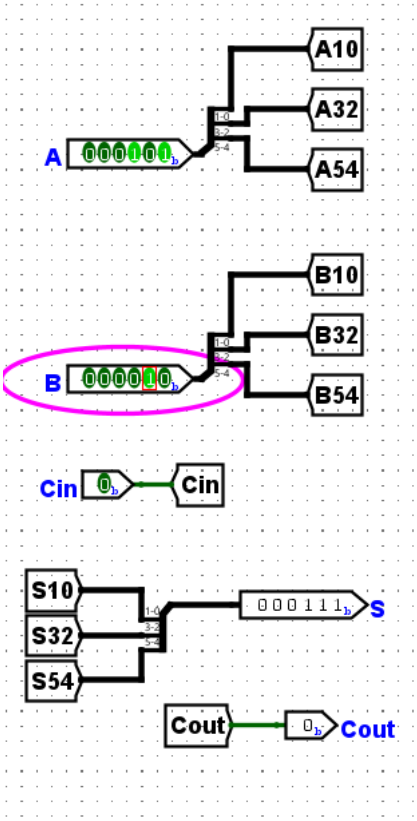
۴ نکات جزئی

- برای ورودی از ۲ عدد ورودی ۶ بیتی استفاده می‌کنیم که با استفاده از اسپلیتر آن را به ۳ بخش ۲ بیتی تقسیم می‌کنیم تا هر ۲ بیت به طور مستقل برای ورودی یک طبقه مورد استفاده قرار بگیرد.
- در نهایت ۳ عدد خروجی ۲ بیتی داریم که برای این بخش نیز با یک اسپلیتر با جهت برعکس، این ۳ بخش به صورت یک خروجی کامل ۶ بیتی استخراج می‌شوند.
- در شکل صورت سوال، کری و خروجی هر جمع‌کننده به صورت یکپارچه با یکدیگر به عنوان ۳ بیت در نظر گرفته می‌شود و به مالتی‌پلکسر ارسال می‌شود تا برای انتخاب استفاده بشود؛ اما در این مدار، به جای این کار، برای هر بخش ۲ عدد مالتی‌پلکسر استفاده می‌کنیم، به این معنی که یکی برای کری و یکی برای حاصل جمع استفاده می‌شود و هر کدام از آن‌ها به طور مستقل به مرحله بعد می‌روند.

۵ تست مدار

۱.۵ تست ۱

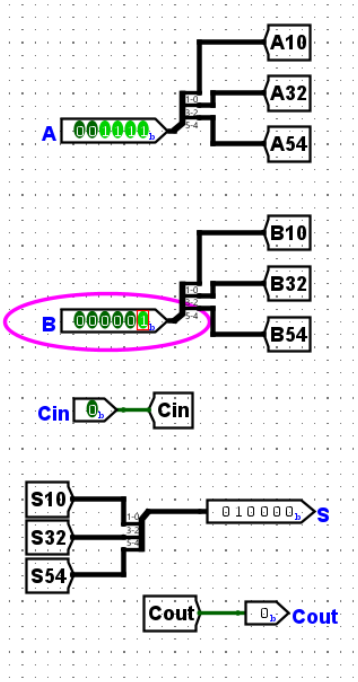
$$C_{in} = 0, A = 000101, B = 000010, S = 000111, C_{out} = 0$$



شکل ۵: تست اول

۲.۵ تست ۲

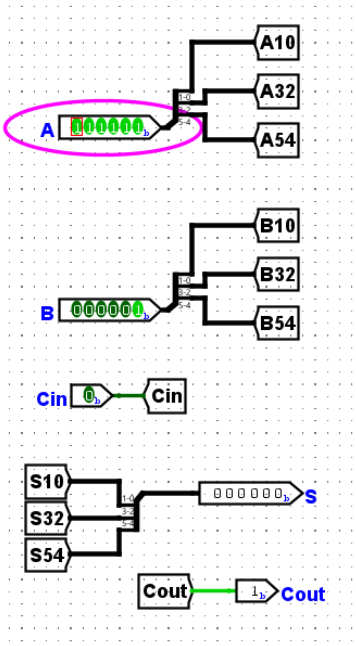
$$C_{in} = 0, A = 001111, B = 000001, S = 010000, C_{out} = 0$$



شکل ۶: تست دوم مدار

۳.۵ تست ۳

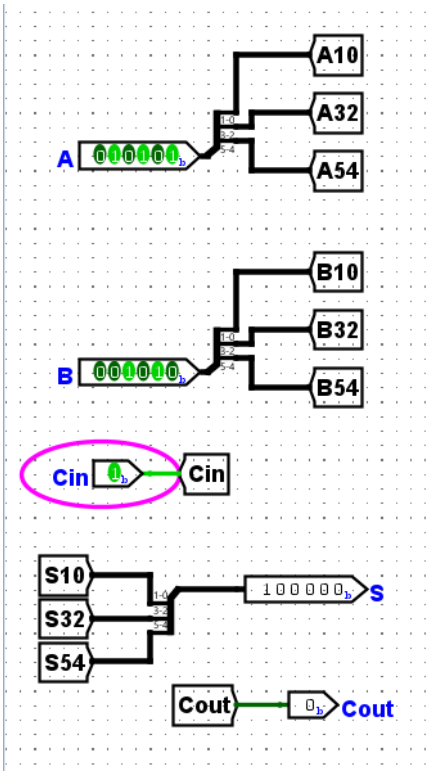
$$C_{in} = 0, A = 111111, B = 000001, S = 000000, C_{out} = 1$$



شکل ۷: تست سوم مدار

۴.۵ تست ۴

$$C_{in} = 1, A = 010101, B = 001010, S = 100000, C_{out} = 0$$



شکل ۸: تست چهارم مدار

۶ نتیجه‌گیری

در این آزمایش متوجه شدیم که می‌شود با استفاده از سخت‌افزار بیشتر، تاخیر را کاهش داد. در اینجا به جای استفاده از یک جمع‌کننده در هر بخش، از ۲ جمع‌کننده استفاده کردیم تا هر ۲ حاصل قابل رویت را حساب کنیم؛ اما در ازای آن دیگر نیازی به منتظر ماندن برای محاسبه کری مرحله قبل نبود و همین باعث شد تاخیر کاهش پیدا کند. به بیانی دیگر، در یک جمع‌کننده معمولی ۶ بیتی، سیگنال کری باید یکی‌یکی از هر ۶ بلوک عبور کند؛ این یعنی تاخیری به اندازه ۶ عمل جمع متوالی. اما در مداری که ما طراحی کردیم، به دلیل انجام محاسبات موازی، تاخیر کل مدار فقط برابر با زمان جمع طبقه اول (دو بیت اول) به اضافه‌ی زمان سوئیچ کردن دو عدد مالتی‌پلکسر می‌شود.