

به نام خدا



درس معماری کامپیوتر  
نیم سال دوم ۰۲-۰۳  
استاد: دکتر حسین اسدی

دانشکده مهندسی کامپیوتر

---

تمرین سری سوم

---

- پرسش‌های خود را در صفحه quera مربوط به تمرین مطرح نمایید.
- سوالات نظری را حتماً به صورت انفرادی و سوالات عملی را می‌توانید در گروه‌های دو نفر تحویل دهید.
- پاسخ‌ها را به صورت تاپی بنویسید.
- اسکرین‌شات‌ها، عکس‌ها و فایل‌های مربوط به سوال عملی را در فایل فشرده مربوطه در cw و quera قرار دهید. هر گونه عدم تطابق بین دو تمرین آپلود شده در دو سایت منجر به از دست رفتن نمره تمرین مربوطه می‌شود.
- پی دی اف قسمت تئوری را در سامانه cw و quera بارگذاری کنید.
- هر دانشجو می‌تواند حداکثر سه تمرین را با دو روز تأخیر بدون کاهش نمره ارسال نماید.

## تمارین تئوری

۱. برای جمع کردن دو عدد ۱۰ بیتی از یک جمع‌کننده Carry Select چند مرحله‌ای استفاده می‌کنیم که در مرحله اول ۲ بیت، در مرحله دوم ۳ بیت و در مرحله سوم ۵ بیت را به کمک Ripple Adder جمع می‌کند. هزینه و تأخیر هر گیت را در این طراحی به ترتیب  $C_g$  و  $D_g$  در نظر بگیرید و هزینه و تأخیر هر MUX را در این طراحی به ترتیب  $C_m = 3C_g$  و  $D_m = 2D_g$  در نظر بگیرید. فرض شده همه تمام جمع‌کننده‌ها<sup>۱</sup> با استفاده از توابع زیر پیاده‌سازی شده‌اند:

$$S = XOR(A, B, C_{in})$$

$$C_{out} = A.B + A.C_{in} + B.C_{in}$$

دقت شود که پهنای ورودی تمامی گیت‌ها و MUX ها یک بیتی هستند و همچنین امکان دسترسی به OR و XOR با سه ورودی وجود دارد که تأخیر و هزینه‌اشان با سایر گیت‌ها یکسان است. (توجه: استفاده از سایر گیت‌های دارای چند ورودی یا MUX به جز ۱: ۲ ممکن نیست).

Adder بالا را ترسیم کنید. هزینه و تأخیر طراحی بالا را بیان کنید و با یک Ripple Carry Adder معادل مقایسه کنید.

۲. فرض کنید، قصد طراحی یک Carry Select Adder با اندازه ۱۲۸ بیتی را داریم. برای این که کمترین میزان تأخیر را داشته باشیم، سائز بلاک‌ها باید در چه اندازه‌ای باشد. فرض کنید که تأخیر تمام جمع‌کننده و MUX یکسان باشد. توجه شود که لزومی ندارد بلوک‌ها ابعاد یکسانی داشته باشند و می‌توانند در ابعاد متفاوت باشند.

۳. در مورد Carry Save Adder تحقیق کنید و نحوه کارکرد آن و تفاوت آن با سایر جمع‌کننده‌هایی که در درس با آن آشنا شدید را توضیح دهید. سپس یک Carry Save Adder با ۸ عدد تمام جمع‌کننده طراحی کنید که بتواند اعداد ۴ بیتی را جمع کند. برای جمع زدن Carry و Sum در مرحله اول، می‌توانید از روش Ripple Carry Adder استفاده کنید.

۴. در کامپیوترهای امروزی مدارهایی به منظور ضرب یک ثبات<sup>۲</sup> در یک عدد ثابت وجود دارد. به عنوان مثال در کامپیوترهای امروزی برای ضرب کردن یک ثبات در توان‌های عدد دو از Barrel Shifter استفاده می‌شود. همان طور که می‌دانید صرفاً انتقال عدد به سمت چپ به اندازه‌ی  $n$  عملاً همان ضرب عدد در  $2^n$  هست. اما یک عملیات دیگر که بسیار مورد استفاده است ضرب یک عدد دودویی در ۱۰ است. از این رو فرض کنید که قرار است یک مدار طراحی کنید که عدد بدون علامت ۳۲ بیتی را به صورت دودویی بگیرد و آن را در عدد ۱۰ ضرب کند. برای این منظور سریع‌ترین مدار ممکن را طراحی نمایید.

۵. اگر بخواهیم ضرب علامت‌دار دو عدد ۰۰۰۰۱۱ و ۰۱۱۱۰۱ را با الگوریتم booth انجام دهیم، با فرض اینکه هر عمل جمع ۱۰ns و هر عمل انتقال ۲ns و هر مکمل‌گیری ۵ns طول بکشد، زمان ضرب با این الگوریتم و حاصل ضرب چه مقدار خواهد بود (مراحل ضرب به روش Booth نوشته شود)؟

۶. با توجه به الگوریتم Booth به سوالات زیر پاسخ دهید.

آ) حداکثر تعداد جمع و تفریق در ضرب booth را برای چهار حالت یعنی ۱) اعداد علامت‌دار  $n$  بیتی و تعداد بیت‌های اعداد ورودی زوج ۲) اعداد علامت‌دار  $n$  بیتی و تعداد بیت‌های اعداد ورودی فرد ۳) اعداد بدون علامت  $n$  بیتی و تعداد بیت‌های اعداد ورودی زوج ۲) اعداد بدون علامت  $n$  بیتی و تعداد بیت‌های اعداد ورودی فرد را به صورت پارامتری محاسبه کنید.

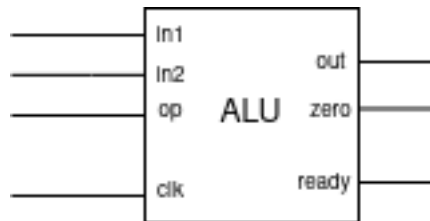
ب) حاصل ضرب  $-13 * -9$  را به روش ضرب booth بدست آورید و تعداد جمع و تفریق‌ها را محاسبه نمایید. ( $n=5$ )

<sup>1</sup>Full Adder

<sup>2</sup>Register

## تمارین عملی

۱. در ادامه تمرینات عملی قصد داریم به طراحی پردازنده MIPS بپردازیم. در این تمرین می‌خواهیم قسمت ALU این پردازنده را طراحی کنیم. این واحد پردازشی دو عدد ۸ بیتی ( $in1$ ,  $in2$ ) و یک کد عملیات ۳ را می‌گیرد. اما خروجی بر اساس نوع عملیات متفاوت است. خروجی یک عملیات ممکن است یک عدد ۸ بیتی به اسم  $out$  باشد یا یک سیگنال یک بیتی به اسم  $zero$  باشد که برای مقایسه‌ها استفاده می‌شود. علاوه بر این یک سیگنال به اسم  $ready$  نیز داریم، این سیگنال هنگامی که جواب آماده شد فعال می‌شود. در شکل زیر می‌توانید ساختار کلی ALU را مشاهده کنید.



حال در ادامه عملیات‌هایی که بر روی دو عدد ۸ بیتی انجام می‌شود را بررسی می‌کنیم.

آ) AND :

$$op = 0000$$

$$out = in1 \text{ AND } in2$$

ب) OR :

$$op = 0001$$

$$out = in1 \text{ OR } in2$$

ج) XOR :

$$op = 0010$$

$$out = in1 \text{ XOR } in2$$

د) Add : Carry Select Adder آنها را جمع می‌کند و خروجی ۸ بیتی را به ما می‌دهد.

$$op = 0011$$

$$out = in1 + in2$$

ه) عملیات تفریق:

$$op = 0100$$

$$out = in1 - in2$$

و) عملیات ضرب: از الگوریتم booth برای ضرب با علامت استفاده می‌کند.

$$\begin{aligned} op &= 0101 \\ out &= in1 \times in2 \end{aligned}$$

ن) Equal :

$$\begin{aligned} op &= 0110 \\ zero &= (in1 == in2) \end{aligned}$$

ح) N-Equal :

$$\begin{aligned} op &= 0111 \\ zero &= (in1 \neq in2) \end{aligned}$$

حتما طراحی خود را به صورت کامل و دقیق انجام دهید چرا که تمارین بعدی به این قسمت وابستگی دارند. همچنین در گزارش خود نیز برای هر حالت حداقل ۳ مثال بزنید و جواب آن‌ها را در گزارش خود بیاورید. توجه کنید که اعداد علامت‌دار هستند.