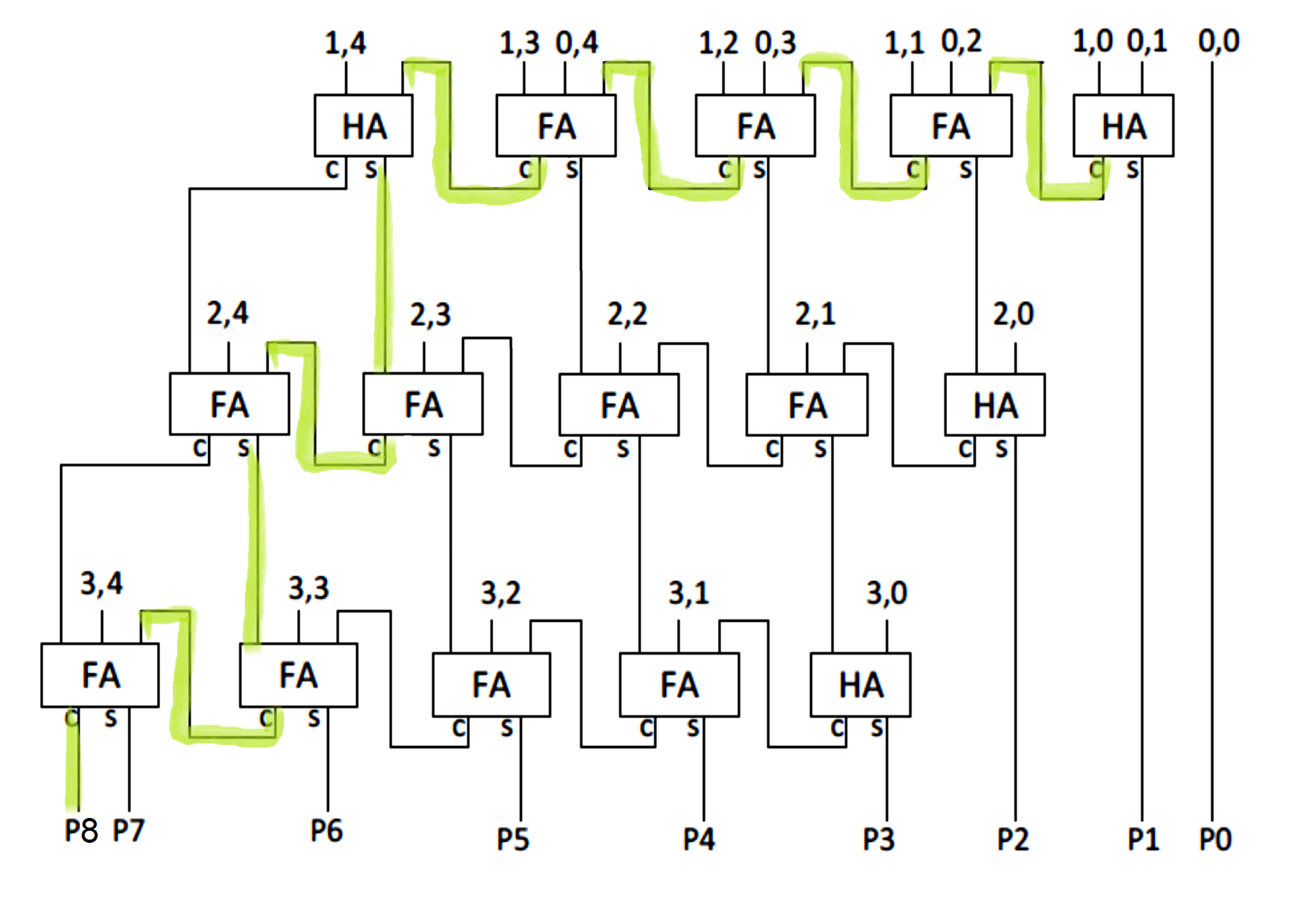
**گزارش دستورکار هفتم آزمایشگاه معماری کامپیوتر**

نگار موقتیان، 9831062

**ماژول simple\_multiplier**

در این قسمت از آزمایش می­خواهیم یک ضرب کنندۀ معمولی 4 در 5 بیتی طراحی کنیم. ابتدا بررسی می­کنیم پاسخ خروجی چند بیتی خواهد بود. این محاسبات برای تمام ضرب کننده­ها در آزمایش­های بعدی نیز معتبر است و تعداد بیت­های خروجی همین مقدار خواهد بود. برای این کار بزرگ­ترین عدد 4 بیتی را در بزرگترین عدد 5 بیتی ضرب می­کنیم تا حداکثر حاصل ضرب را بیابیم.

بنابراین حاصل حداکثر 9 بیتی خواهد بود. مدار این ضرب کنندۀ 4 در 5 به صورت زیر می­باشد.



همانطور که در این شکل دیده می­شود نیاز به 4 نیم جمع کننده و 11 تمام جمع کننده داریم که در سه طبقه (1 - 4) قرار دارند. در این ضرب کننده مانند ضرب دهدهی هر بیت از عدد 4 بیتی A را در بیت­های عدد 5 بیتی B ضرب کرده و با یک شیفت با حاصل مرحلۀ بعد جمع می­زنیم.

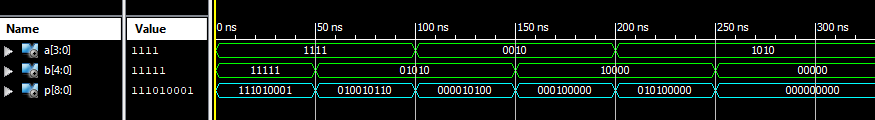
بنابر توضیحات گفته شده هزینۀ ساخت این ضرب کننده برابر است با:

Cost = 20 AND + 4 HA + 11 FA = 20g + 4(2g) + 11(5g) = 83g

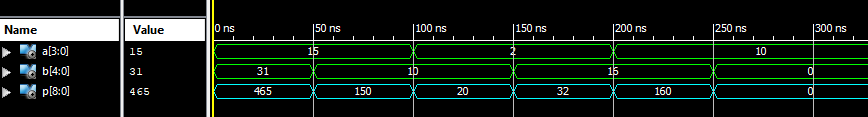
همچنین تاخیر این مدار، تاخیر مسیر بحرانی آن است که در شکل با رنگ سبز مشخص شده است و برابر است با:

Delay = delay AND + delay(carry) HA + 6 delay(carry) FA + delay(sum) HA + delay(sum) FA = d + d + 6(2d) + d + d = 16d

پس از پیاده سازی مدار شبیه سازی آن توسط test bench نوشته شده و به ازای مقادیر مختلف A و B انجام شد، تا از درستی رفتار مدار ساخته شده اطمینان حاصل شود.



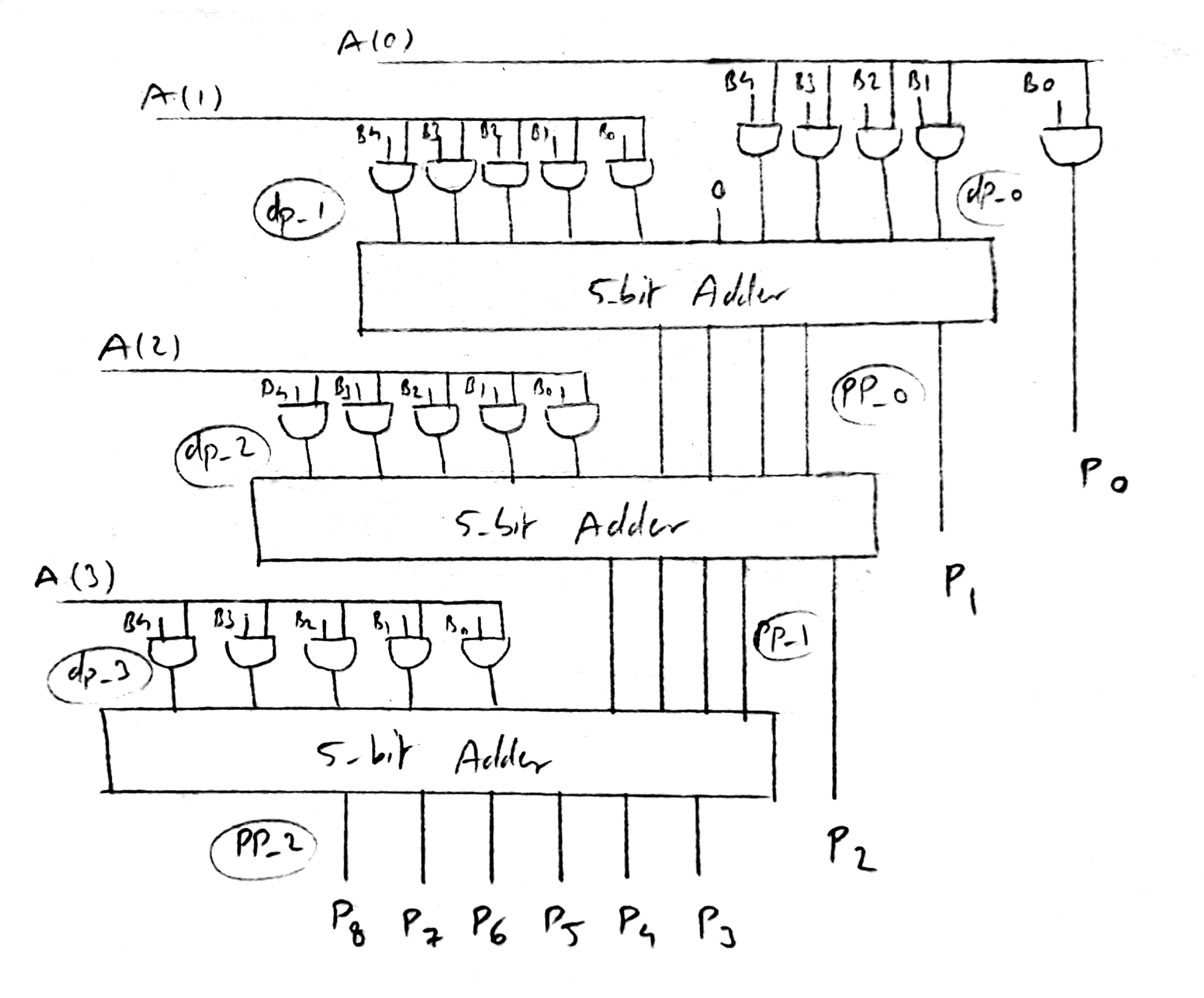
برای سهولت بیش­تر در بررسی سیگنال­های ورودی و خروجی تمام اعداد چند بیتی را با فرض بی­علامت بودن به مبنای 10 می­بریم. نتیجۀ تنظیمات گفته شده به صورت زیر است.



همانطور که مشاهده می­شود تمامی حاصل ضرب­ها به درستی انجام شده­اند.

**ماژول array\_multiplier**

حال در این قسمت از آزمایش می­خواهیم یک ضرب کنندۀ آرایه­ای 4 در 5 بیتی طراحی کنیم. همانطور که در قسمت قبل توضیح دادیم خروجی این ضرب کننده باید 9 بیتی باشد. مدار این ضرب کنندۀ 4 در 5 به صورت زیر می­باشد.



این ضرب کننده درست مانند ضرب کنندۀ قبلی عمل می­کند، با این تفاوت که به جای استفاده از تمام جمع­کننده­ها و نیم جمع کننده­ها برای جمع هر مرحله با مرحلۀ قبل از یک جمع کنندۀ 5 بیتی مانند یک جمع کنندۀ آبشاری استفاده می­کند.

هزینۀ ساخت این جمع کننده به سادگی برابر است با:

Cost = 20 AND + 3 RA:5-bit = 20g + 3(25g) = 85g

همچنین هزینۀ این مدار را می­توان به دو صورت تفسیر کرد. زمانی که هر جمع کنندۀ آبشاری را به صورت یک Black Box بینیم داریم:

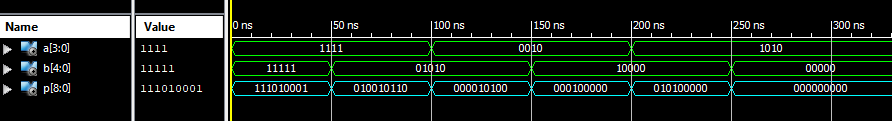
Delay = delay AND + 3(delay (carry) RA) = d + 3(10d) = 31d

اما در عمل چنین نیست و محاسبات بیت­های کم ارزش­تر می­توانند سریع­تر حاضر شوند. اگر نگاه جزئی­تری داشته باشیم و هر جمع کنندۀ آبشاری را به صورت مجموعه­ای از تمام جمع کننده­ها ببینیم داریم:

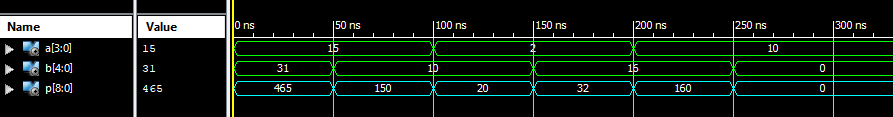
Delay = delay AND + 7 delay(carry) FA + 2 delay(sum) HA = d + 7(2d) + 2d = 17d

لذا هزینه و تاخیر این جمع کننده از جمع کنندۀ معمولی کمی بیش­تر است (به دلیل اینکه به جای HA های موجود در جمع کنندۀ معمولی از FA استفاده می­شود)، اما به دلیل اینکه جمع­ کننده­ها به صورت آماده موجود هستند پیاده­سازی این مدار از مدار قبلی ساده­تر است.

پس از پیاده سازی مدار شبیه سازی آن توسط test bench نوشته شده و به ازای مقادیر مختلف A و B انجام شد، تا از درستی رفتار مدار ساخته شده اطمینان حاصل شود.



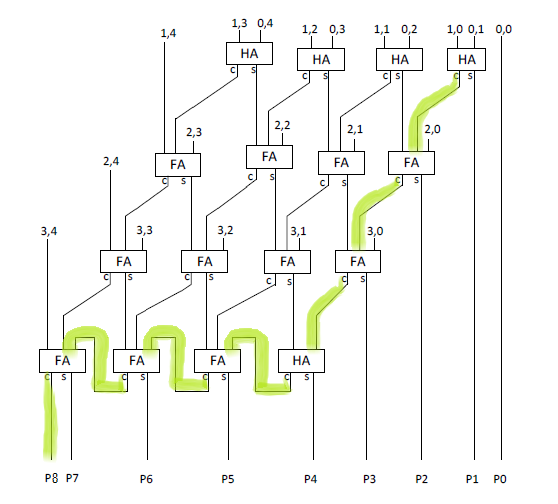
برای سهولت بیش­تر در بررسی سیگنال­های ورودی و خروجی تمام اعداد چند بیتی را با فرض بی­علامت بودن به مبنای 10 می­بریم. نتیجۀ تنظیمات گفته شده به صورت زیر است.



همانطور که مشاهده می­شود تمامی حاصل ضرب­ها به درستی انجام شده­اند.

**ماژول save\_adder\_multiplier**

در نهایت می­خواهیم یک ضرب کنندۀ 4 در 5 بیتی با استفاده از جمع کنندۀ ذخیره­گر رقم نقلی طراحی کنیم. همانطور که در قسمت قبل توضیح دادیم خروجی این ضرب کننده باید 9 بیتی باشد. مدار این ضرب کنندۀ 4 در 5 به صورت زیر می­باشد.



روش کار این ضرب کننده کمی متفاوت با ضرب کننده­های قبلی و مبتنی بر روش ذخیرۀ رقم نقلی است.

هزینۀ ساخت این جمع کننده به سادگی برابر است با:

Cost = 20 AND + 4 HA + 11 FA = 20g + 4(2g) + 11(5g) = 83g

که با هزینۀ ضرب کنندۀ اول برابر است.

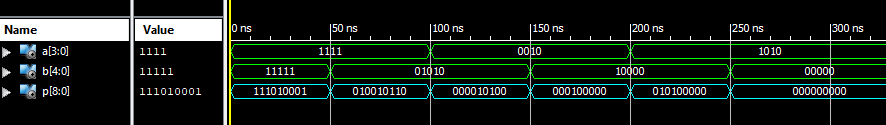
همچنین تاخیر این مدار، تاخیر مسیر بحرانی آن است که در شکل با رنگ سبز مشخص شده است و برابر است با:

Delay = delay AND + 2 delay(carry) HA + 5 delay(carry) FA = d + 2d + 5(2d) = 13d

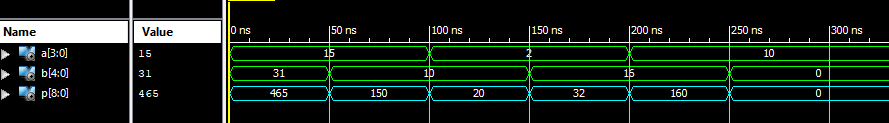
که در مقایسه با ضرب کننده­های قبلی تاخیر بسیار خوبی است.

در کل نیز به نظر می­رسد این ضرب کننده بهترین performance per cost میان این چند ضرب کننده را داراست، زیرا کم­ترین هزینه و کم­ترین تاخیر را دارد.

پس از پیاده سازی مدار شبیه سازی آن توسط test bench نوشته شده و به ازای مقادیر مختلف A و B انجام شد، تا از درستی رفتار مدار ساخته شده اطمینان حاصل شود.



برای سهولت بیش­تر در بررسی سیگنال­های ورودی و خروجی تمام اعداد چند بیتی را با فرض بی­علامت بودن به مبنای 10 می­بریم. نتیجۀ تنظیمات گفته شده به صورت زیر است.



همانطور که مشاهده می­شود تمامی حاصل ضرب­ها به درستی انجام شده­اند.

در نهایت برای جمع بندی می­توان جدول زیر را ارائه داد.

|  |  |  |  |
| --- | --- | --- | --- |
| ضرب کنندۀ ذخیره­گر نقلی | ضرب کنندۀ آرایه­ای | ضرب کنندۀ عادی |  |
| 83g | 85g | 83g | هزینه |
| 13d | 17d | 16d | تاخیر |
| مانند جمع­کنندۀ ذخیره­گر نقلی و بر خلاف جمع کنندۀ آبشاری، رقم نقلی هر مرحله را نگه داشته و مانند یک عدد آن را با اعداد اصلی جمع  می­زند | همانند ضرب کنندۀ عادی | مانند ضرب دهدهی هر رقم عدد اول در ارقام عدد دوم ضرب شده و در هر مرحله با یک شیفت با مرحلۀ بعد جمع می­شود | روش کار |
| سریع­ترین و کم­ هزینه­ترین ضرب کننده میان این سه است | به دلیل استفاده از جمع کننده­های آماده پیاده سازی نسبتاً ساده­ای دارد | هزینه و تاخیر نسبتاً خوب | مزایا |
| نسبت به ضرب کنندۀ آرایه­ای پیاده سازی دشوارتری دارد | نسبت به دیگر ضرب کننده­ها کند است و هزینۀ بیش­تری دارد | نسبت به ضرب کنندۀ آرایه­ای پیاده سازی دشوارتری دارد و نسبت به ضرب کنندۀ ذخیره گر نقلی کندتر است | معایب |