با وجود این که سرعت و مساحت کماکان هدف های اصلی طراحی VLSI امروزی است توان مصرفی به یک دغدغه ی بسیار مهم در طراحی VLSI امروزی بدل گشته است . نگاه به کاربرد های مختلف نشان می دهد که هر کدام از این کاربرد ها ، نیاز ها و نگرانی های خاص خود را در زمینه ی توان مصرفی دارند و نگاهی به آینده نشان می دهد که این نیاز ها و نگرانی ها نه تنها در حال رفع شدن نیست بلکه همه چیز نشانگر این است که در آینده پررنگ تر نیز خواهند شد. برای یک ایده ی بسیار کلی درباره ی این موضوع ، می توان در نظر گرفت که تمام انرژی موجود در کهکشان راه شیری 1059 ژول است از طرفی کران بالا برای تعداد عملیات پردازشی که در یک سال می تواند انجام بگیرد 1078 \* 6 است که در نظر می گیریم که توسط یک بیلیون کامپیوتر MOPS100 انجام میگیرد با فرض این که طبق قانون مور نیاز های پردازشی بشر هر سال دوبرابر می شود با این روند تمام انرژی کهکشان راه شیری باید تا 180 سال دیگر مصرف شود.

مساله ای که در سال های اخیر مطرح شده است ، نیاز به الکترونیک سبز است به این معنا که فرایند طراحی ، تولید و استفاده از و {دفع؟disposal} کامپیوتر ها ، سرور ها و وسایل حانبی آن ها همانند مونیتور ، پرینتر ادوات ذخیره سازی اطلاعات و ابزار های شبکه و انتقال اطلاعات به گونه ای باشد که کمترین تاثیر را بر روی محیط زندگی داشته باشند .

درحالی که تا مدتی پیش اتلاف توان قسمت های الکترونیکی، تنها بخش کوچکی از بودجه ی برق مصرفی را تشکیل می داد ، این وضع ، در دهه های اخیر تغییر کرده است .

استفاده ی فراگیر از کامپیوتر های رومیزی و لپ تاپ ها در انواع محیط های خانگی و اداری بسیار مرسوم شده است . توان مصرفی زمان بیکاری (standby) قطعات الکترونیکی مصرفی و setup box به سرعت در حال افزایش است به طوری که در حال حاضر توان مصرف آن ها معادل یخچالی با حجم معادل است . قطعات الکترونیکی در حال تبدیل شدن به یکی از بخش های عمده ی مصرف توان در یک اتومبیل مدرن می شوند .

به گفته ی Luis Barosso از گوگل (شرکتی که یکی از فعال ترین پیش برندگان مفهوم پردازش از راه دور است ) هزینه ی یک Datacenter به تنهایی توسط صورتحساب ­ماهیانه ی برق مصرفی آن مشخص می شود ( نه هزینه های سخت افزاری و یا نگه داری ) ، این انرژی صرف توان تلف شده در قطعات الکترونیکی و نیز توان لازم برای تجهیزات خنک کننده یعنی سیستم های تهویه می شود ، این دلیل این حقیقت را توضیح می دهد که datacenter ها به دقت در مکان هایی انتخاب می شوند که انرژی الکتریکی و تکنیک های کارآمد سرمایشی به راحتی در دسترس است (مثلا در نزدیکی رودخانه های بزرگ ).

در زمینه ی مسئله ی دفع حرارت ناهمگونی مصرف توان در مکان های گوناگون یک چیپ خود موضوعی است که نیاز به توجّه دارد زیرا این باعث می شود که تراشه در نقاطی که تراکم مصرف توان وجود دارد دمای بسیار بالایی داشته ونتواند به صورت کار آمدی خنک شود . این مشکل در SOC ها که در آن ها مؤلفه های مختلفی با توان های متفاوت حضور دارند حاد تر نیز می باشد.در آن ها برای غلبه بر این مشکل باید یک بسته بندی پیچیده ساخته شود که گرما را در سطح بزرگ تری پخش کند و به این صورت دفع حرارت را بهبود بخشد. این باعث می شود که در قطعات با کارکرد بالا هزینه ِی { {Packaging قسمت اصلی یا حداقل سهم مهمی از هزینه ی کل را تشکیل میدهد.

\*\*\*\*

در یک وسیله ی الکترونیکی که با باطری کار می کند کل انرژی قابل استفاده ،مقدار ثابتی است و این اندازه ی توان مصرفی است که عمر شارژ باطری را مشخص می کند . ابعاد ،حجم و وزن باطری را نمی توان به صورت دلخواه در نظر گرفت و معمولا وسیله ی مورد نظر این پارامتر هارا تعیین خواهد کرد مثلا مقدار عمر باطری ای که از یک گوشی موبایل مورد توقع است چند روز زمان انتظار و 5- 4 ساعت زمان مکالمه است بنابراین تکنولوژی های امروزی ساخت باطری حکم می کنند که توان مصرفی یک گوشی موبایل نباید از 3W بیشتر باشد این توان باید جوابگوی انتظارات امروزه ی استفاده کنندگان از قبیل پخش موزیک ، ارتباط های ،3G و WiFi باشد.

\*\*\*

برای بالا بردن عمر باطری ها تلاش های بسیاری انجام گرفته و تکنولوژی های جایگزین مانند سلول های سوختی که از اکسید شدن سوخت ها (مانند متانول ، اتانول و بنزین ) یا اکسیدان ها (مانند هیدروژن با روی ) استفاده می کنند .بسیار مورد توجه هستند اما این تکنولوژی ها به بازار نیامده اند و کماکان تکنولوژی های یون لیتیوم و پلیمر لیتیم و تکنولوژی های برتر موجود در بازار اند .

انگیزه ی سوم که موجب رفتن به سمت طراحی بسیار کم مصرف می شود ،ظهور یک دسته از کاربرد های جدید به نام قطعات الکترونیکی صفر توان (نا پدید شونده) است (گره های میکرو واتی ) .

کوچک سازی پیوسته قطعات پردازشی که توسط فرایند کاهش ابعد تکنولوژی نیمه هادی محقق شده تولید حسگر های بسیار ریز بی سیم (WSN ها) را محقق ساخته که ابعاد آن ها در حد سانتی متر مکعب و کمتر از آن است این قطعات می توانند در اشیایی که به صورت روزمره با آن ها سرو کار داریم تعبیه شوند . شکل مجتمع و قیمت بسیار پایین استفاده از آن ها در تعداد بسیار وسیع را ممکن می سازد . این قطعات حسگر بی سیم از زمان پیدایش آن ها در اواخر دهه ی 1990 تحولات بسیاری را موجب شده اند .

برای موفقیت چنین قطعاتی مشکل انرژی یکی از اصلی ترین موانعی است که باید بر آن غلبه شود جایگزینی یا شارژ مجدد در چنین اصولا مطرح نیست . بنابراین چنین وسیله ای باید تا آخر عمر خود (که ممکن است ده ها سال باشد) از لحاظ انرژی خودکفا باشد بنابراین یا باید بتواند از یک بار شارژ باتری تا پایان عمرخود استفاده کند یا بتواند انرژی خود را ازمنبع دیگری مثلا محیط اطراف به دست آورد از زمان شکل گیری مفهوم WSN پیشرفت های زیادی در خصوص کاهش اندازه هزینه و مصرف چنین قطعاتی به وجود آمده است

پیشرفت های طراحی بسیار کم مصرف و کوچک سازی مدار در حد بسیار بالا می تواند باعث پیدایش کاربردهایی شود که در غیر این صورت غیر ممکن می شد مثلا شبکه ی متراکمی از که روی سطحی گسترده قرار می گیرند می توانند تشکیل یک پوست مصنوعی بدهند که به لمس ، تنش، فشار و یا فرسودن حساس باشد. کاربرد های واضح چنین شبکه ای می نواند بالهای هوشمند هواپیما رابط های کاربری بدیع و یا روبات های پیشرفته باشد .

همچنین کاربرد های دیگری همچون ابزارهای تشخیص و نظارت پزشکی و یا جراحی که به بدن فرد تزریق می شوند می توان تصور کرد. اما برای تحقق یافتن چنین کاربرد هایی گره های سانتی متر مکعبی امروزی باید به واقع تبدیل به قطعه های با ابعاد گرد و غبار (یعنی میلیمتر مکعب) شوند. یک انگیزه ی واقعی جستجوی بیشتر در انتهای مرزهای طراحی کم مصرف ایجاد می کنند .

عمل ضرب در بسیاری ازسیستم های پردازش محاسباتی یک عمل بنیادی است . ضرب کننده ها مساحت زیادی را اشغال می کنند . تاخیر زیادی دارند و توان قابل توجهی را مصرف می کنند.

توان به مقدار انرژی بر حسب انرژی برحسب ژول که در یک بازه ی زمانی معین مصرف می شود اتلاق می گردد به زبانی طراحی با توان مصرفی پایین از طراحی برای انرژی مصرفی پایین متمایز است با وجود این که این دو به هم مرتبط هستند. توان مساله ای است که عمدتا زمانی مطرح می گردد که دغدغه ی خنک کردن وجود دارد . برای طراحی سیم بندی زمین و منبع و تحلیل حاشیه ی نویز و قابلیت اطمینان اکثرا توان مصرفی ماکزیمم مورد استفاده قرار می گیرد . انرژی لازم برای یک عمل واحد مقیاس بهتری برای کارایی از لحاظ مصرف انرژی است به ویژه زمانی که هدف افزایش طول عمر باتری است.

**روش های ضرب**  
چندین نوع ضرب کننده مورد مطالعه قرار گرفته اند ضرب کننده های آرایه ای خطی ضرب کننده های آرایه ای leapfrog ضرب کننده های آرایه ای خطی ضرب کننده های آرایه ای خطی چپ به راست ضرب کننده های آرایه ای تفکیکی و ضرب کننده های درختی

**>>>>>>>>>>>>>> کتاب روش های ضرب <<<<<<<<<**

الگوریتم ضرب ترتیبی

مضروب فیه و مضروب به ترتیب با نماد های و با رشته ای از ارقام به صورت زیر نشان داده می شوند :

,

که و نشان دهنده ی رقم علامت ، فارغ از نحوه ی نمایش علامت عدد ( نمایش علامت به صورت اندازه ی علامت یا روش های مکمل )،هستند .

الگوریتم علامت ضرب به صورت ترتیبی شامل گام یا مرحله است که در گام ام بیت ازمضروب فیه بررسی شده و حاصل ضرب به حاصل ضرب های محاسبه شده ی قبلی () اضافه می گردد . عبارت مناسب برای این فرآیند بازگشتی به صورت

; (3-1)

و در گام اول است . ضرب در ، آن را یک موقعیت به سمت راست شیفت می دهد که این عمل به منظور هم تراز سازی قبل از جمع با صورت می گیرد . از آنجا که وزن دو برابر است ، این هم تراز سازی ضروری است . برای اثبات فرآیند بازگشتی فوق ، با تکرار جایگزاری در رابطه بازگشتی (3-1) خواهیم داشت :

اگر هر دو عملوند مثبت باشند ( به عبارت دیگر حاصل ضرب از رابطه ی زیر بدست می آید

(3-2)

مقدار بدست آمده حاصل ضربی است که اندازه ی آن بیت دارد، زیرا بیشینه ی مقدار به دست آمده برای هنگامی به دست می آید که دارای بیشینه ی مقدار خود باشند بنابراین

( (3-3)

ازآنجا که مقدار داخل پرانتز در رابطه ی (3-3) به ازای مثبت است ، نا معادله ی زیر بر قرار می باشد

(3-4)

لذا بیت برای نمایش اندازه ی حاصل ضرب نیاز است که با اضافه شدن بیت در مجموع به بست نیاز خواهد بود.

برای ضرب اعداد به شکل اندازه – علامت ، اندازه های دو عدد با استفاده ازالگوریتم فوق در هم ضرب شده و علامت نتیجه نتیجه جداگانه محاسبه می گردد (اگر هر دو عدد هم علامت باشند نتیجه مثبت و در غیر این صورت نتیجه منفی خواهد بود). در نمایش های مکمل یک و مکمل دو ، عمل ضرب با مضروب منفی با عمل ضرب با مضروب فیه منفی متفاوت است و تنها وقتی مضروب منفی باشد الگوریتم فوق بدون تغییر قابل استفاده است خواهد بود ،در این حالت چند برابر شده های یک عدد منفی با هم جمع می شوند و نحوه ی نمایش آن ها تاثیری در روند اجرای الگوریتم نخواهد داشت .این موضوع در مثال بعدی نشان داده شده است .

مثال 3-1

در مثال پیش رو مضروب یک عدد منفی با نمایش مکمل دو و مضروب فیه مثبت در نظر گرفته شده است . هر دوعدد 4 بیتی هستند بنابراین حاصل ضرب آن ها به همراه بیت علامت دارای 7 بیت خواهد بود. در یک واحد محاسباتی با عملوند های 4 بیتی ،تمام رجیسترها 4 بیتی هستند،لذا رجیستری با طول دو برابر برای ذخیرهتمام بیت های حاصل ضرب لازم می باشد.در جدول زیر خط عمودی دو نیمه ی پر ارزش و کم ارزش حاصل ضرب را از هم جدا می کندکه هر یک می توانند در رجیسترهای با طول واحد ذخیره شوند .

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | 1 0 1 1  Shift 1 1 0 1  1 0 1 1 |

سه بیت از مضروب فیه ، و به صورت تک به تک و با شروع از کم ارزش ترین رقم مورد بررسی قرار گرفته اند و یک عمل جمع همراه با شیفت و با یک شیفت تنها (با توجه بیت بررسی شده مضروب فیه ) انجام شده است . نتیجه ی نهایی مقدار منفی است که به شکل درستی در نمایش مکمل 2 نشان داده شده است . لازم به ذکر است که فقط 4 بیت پر ارزش حاصل ضرب جزئی که در رجیستر سمت چپ نگه داری می شود در عمل جمع شرکت می کند و رجیستر حاوی بیت های کم ارزش وارد عمل جمع نمی شوند . در پایان تنها از سه بیت رجیستر دوم (حاوی بیت های کم ارزش ) استفاده می شود و کم ارزش ترین مکان آن بدون استفاده باقی می ماند. در اینجا جواب به دست آمده قابل قبول خواهد بود و آرایش مجدد آن ها امری ضروری نیست اما برای پر کردن این فضا خالی می توان بدیت صورت عمل نمود : سه بیت موجود در رجیستر دوم در سه موقعیت سمت راست رجیستر قار گیرند و بیت علامت آن (1) همواره مقدار آن بدون توجه به علامت حاصل ضرب صفر در نظر گرفته شود و یا(2) مقدار آن برابر بیت علامت رجیستر اول (بیت علامت حاصل ضرب ) می گردد. یک روش دیگر برای آزمایش مجدد به حاصل به دست آمده شیفت به راست آن و تکرار بیت علامت می باشد .

شرایط وقتی که مضروب فیه منفی باشد متفاوت می گردد،در الگوریتم ارائه شده که هر بیت به صورت جداگانه مورد بررسی قرار می گیرد به طور یقین نمی توان برای بیت علامت (وزن بیت علامت منفی است ) روشی مشابه سایر بیت ها در پیش گرفت . اعداد مکمل دو را می توان به صورت :

(3-5)

نمایش داد که است .

اگر بیت علامت مضروب فیه نادیده گرفته شود نتیجه ی نهایی برابر خواهد بود با

(3-6)

که عبارت مقدار مطلوب است ، از این رو اگر باشد یک عمل تصحیح به شکل زیر نیاز خواهد بود :

(3-7)

به عبارت دیگر اگر باشد،می بایست مضروب از نیمه ی پر ارزش کم شود .

مثال 3-2

در این مثال هر دو عدد مضروب فیه منفی و به شکل مکمل 2 هستند

در مرحله تصحیح عمل تفریق مضروب با استفاده از جمع با مقدار مکمل 2 آن انجام می شود.

به همین ترتیب برای مکمل یک خواهیم داشت

(3-8)

در نتیجه

(3-9)

بنابراین اگر باشد،عمل تصحیح دو قسمت خواهد داشت (کم کردن مقدار ، و افزودن ) ، بنابراین ، برای اعمال قسمت دوم عبارت تصحیح با مقدار اولیه ی شروع می نماییم و در پایان عملیات قسمت اول عبارت تصحیح را اعمال می نماییم.

مثال 3-3

حاصل ضرب اعداد 5 و 3- در نمایش مکمل یک در این مثال نشان داده شده است.

مشابه مثال قبل ،تفریق در عملیات تصحیح با استفاده از جمع با مکمل (در این مثال جکع با مکمل 1 )انجام گرفته است.اما بر خلاف مثال قبل عمل مکمل 1 کردن روی هر دو رجیستر (قسمت کم ارزش و پر ارزش ) می بایست صورت گیرد (برای دلیل این موضوع به بخش 1-6 مراجعه شود) و این معنی به کار گیری یک جمع کننده دودویی با طول دو برابر برای این مرحله است.

الگوریتم ضرب ترتیبی

مضروب فیه و مضروب به ترتیب با نماد های و با رشته ای از ارقام به صورت زیر نشان داده می شوند :

,

که و نشان دهنده ی رقم علامت ، فارغ از نحوه ی نمایش علامت عدد ( نمایش علامت به صورت اندازه ی علامت یا روش های مکمل )،هستند .

الگوریتم علامت ضرب به صورت ترتیبی شامل گام یا مرحله است که در گام ام بیت ازمضروب فیه بررسی شده و حاصل ضرب به حاصل ضرب های محاسبه شده ی قبلی () اضافه می گردد . عبارت مناسب برای این فرآیند بازگشتی به صورت

; (3-1)

و در گام اول است . ضرب در ، آن را یک موقعیت به سمت راست شیفت می دهد که این عمل به منظور هم تراز سازی قبل از جمع با صورت می گیرد . از آنجا که وزن دو برابر است ، این هم تراز سازی ضروری است . برای اثبات فرآیند بازگشتی فوق ، با تکرار جایگزاری در رابطه بازگشتی (3-1) خواهیم داشت :

اگر هر دو عملوند مثبت باشند ( به عبارت دیگر حاصل ضرب از رابطه ی زیر بدست می آید

(3-2)

مقدار بدست آمده حاصل ضربی است که اندازه ی آن بیت دارد، زیرا بیشینه ی مقدار به دست آمده برای هنگامی به دست می آید که دارای بیشینه ی مقدار خود باشند بنابراین

( (3-3)

ازآنجا که مقدار داخل پرانتز در رابطه ی (3-3) به ازای مثبت است ، نا معادله ی زیر بر قرار می باشد

(3-4)

لذا بیت برای نمایش اندازه ی حاصل ضرب نیاز است که با اضافه شدن بیت در مجموع به بست نیاز خواهد بود.

برای ضرب اعداد به شکل اندازه – علامت ، اندازه های دو عدد با استفاده ازالگوریتم فوق در هم ضرب شده و علامت نتیجه نتیجه جداگانه محاسبه می گردد (اگر هر دو عدد هم علامت باشند نتیجه مثبت و در غیر این صورت نتیجه منفی خواهد بود). در نمایش های مکمل یک و مکمل دو ، عمل ضرب با مضروب منفی با عمل ضرب با مضروب فیه منفی متفاوت است و تنها وقتی مضروب منفی باشد الگوریتم فوق بدون تغییر قابل استفاده است خواهد بود ،در این حالت چند برابر شده های یک عدد منفی با هم جمع می شوند و نحوه ی نمایش آن ها تاثیری در روند اجرای الگوریتم نخواهد داشت .این موضوع در مثال بعدی نشان داده شده است .

مثال 3-1

در مثال پیش رو مضروب یک عدد منفی با نمایش مکمل دو و مضروب فیه مثبت در نظر گرفته شده است . هر دوعدد 4 بیتی هستند بنابراین حاصل ضرب آن ها به همراه بیت علامت دارای 7 بیت خواهد بود. در یک واحد محاسباتی با عملوند های 4 بیتی ،تمام رجیسترها 4 بیتی هستند،لذا رجیستری با طول دو برابر برای ذخیرهتمام بیت های حاصل ضرب لازم می باشد.در جدول زیر خط عمودی دو نیمه ی پر ارزش و کم ارزش حاصل ضرب را از هم جدا می کندکه هر یک می توانند در رجیسترهای با طول واحد ذخیره شوند .

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | 1 0 1 1  Shift 1 1 0 1  1 0 1 1 |

سه بیت از مضروب فیه ، و به صورت تک به تک و با شروع از کم ارزش ترین رقم مورد بررسی قرار گرفته اند و یک عمل جمع همراه با شیفت و با یک شیفت تنها (با توجه بیت بررسی شده مضروب فیه ) انجام شده است . نتیجه ی نهایی مقدار منفی است که به شکل درستی در نمایش مکمل 2 نشان داده شده است . لازم به ذکر است که فقط 4 بیت پر ارزش حاصل ضرب جزئی که در رجیستر سمت چپ نگه داری می شود در عمل جمع شرکت می کند و رجیستر حاوی بیت های کم ارزش وارد عمل جمع نمی شوند . در پایان تنها از سه بیت رجیستر دوم (حاوی بیت های کم ارزش ) استفاده می شود و کم ارزش ترین مکان آن بدون استفاده باقی می ماند. در اینجا جواب به دست آمده قابل قبول خواهد بود و آرایش مجدد آن ها امری ضروری نیست اما برای پر کردن این فضا خالی می توان بدیت صورت عمل نمود : سه بیت موجود در رجیستر دوم در سه موقعیت سمت راست رجیستر قار گیرند و بیت علامت آن (1) همواره مقدار آن بدون توجه به علامت حاصل ضرب صفر در نظر گرفته شود و یا(2) مقدار آن برابر بیت علامت رجیستر اول (بیت علامت حاصل ضرب ) می گردد. یک روش دیگر برای آزمایش مجدد به حاصل به دست آمده شیفت به راست آن و تکرار بیت علامت می باشد .

شرایط وقتی که مضروب فیه منفی باشد متفاوت می گردد،در الگوریتم ارائه شده که هر بیت به صورت جداگانه مورد بررسی قرار می گیرد به طور یقین نمی توان برای بیت علامت (وزن بیت علامت منفی است ) روشی مشابه سایر بیت ها در پیش گرفت . اعداد مکمل دو را می توان به صورت :

(3-5)

نمایش داد که است .

اگر بیت علامت مضروب فیه نادیده گرفته شود نتیجه ی نهایی برابر خواهد بود با

(3-6)

که عبارت مقدار مطلوب است ، از این رو اگر باشد یک عمل تصحیح به شکل زیر نیاز خواهد بود :

(3-7)

به عبارت دیگر اگر باشد،می بایست مضروب از نیمه ی پر ارزش کم شود .

مثال 3-2

در این مثال هر دو عدد مضروب فیه منفی و به شکل مکمل 2 هستند

در مرحله تصحیح عمل تفریق مضروب با استفاده از جمع با مقدار مکمل 2 آن انجام می شود.

به همین ترتیب برای مکمل یک خواهیم داشت

(3-8)

در نتیجه

(3-9)

بنابراین اگر باشد،عمل تصحیح دو قسمت خواهد داشت (کم کردن مقدار ، و افزودن ) ، بنابراین ، برای اعمال قسمت دوم عبارت تصحیح با مقدار اولیه ی شروع می نماییم و در پایان عملیات قسمت اول عبارت تصحیح را اعمال می نماییم.

مثال 3-3

حاصل ضرب اعداد 5 و 3- در نمایش مکمل یک در این مثال نشان داده شده است.

مشابه مثال قبل ،تفریق در عملیات تصحیح با استفاده از جمع با مکمل (در این مثال جکع با مکمل 1 )انجام گرفته است.اما بر خلاف مثال قبل عمل مکمل 1 کردن روی هر دو رجیستر (قسمت کم ارزش و پر ارزش ) می بایست صورت گیرد (برای دلیل این موضوع به بخش 1-6 مراجعه شود) و این معنی به کار گیری یک جمع کننده دودویی با طول دو برابر برای این مرحله است.

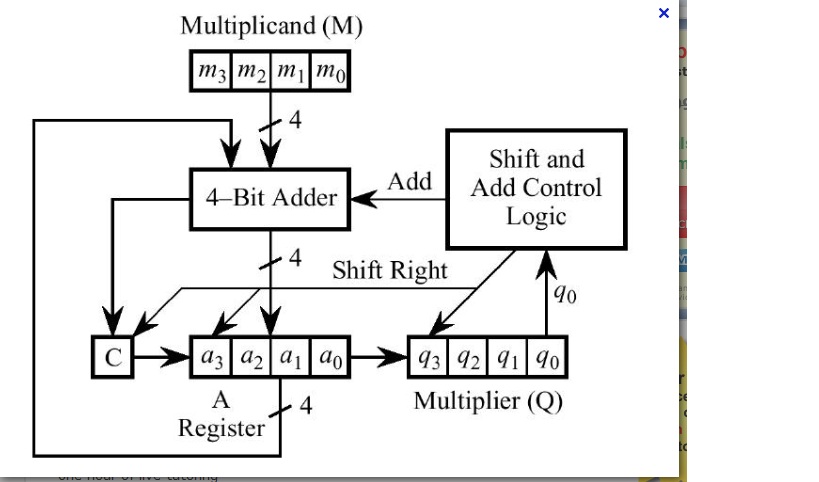
**>>>>>>>>>>>>> ضرب کننده ی با توان مصرفی پایین <<<<<<**

کار های بسیار زیادی بر روی ضرب کننده ی کم مصرف در سطوح مختلف تکنولوژی فیزیکی مداری و منطقی انجام گرفته است . این تکنیک های پایین آوردن توان مصرفی منحصر به واحد های ضرب کننده نیستند و عموما قابل اعمال به سایر قسمت ها نیز می باشند

**>>>>> کاهش توان مصرفی ضرب کننده ی شیفت و جمع <<<<<**

ضرب کننده ی سریال ساده ترین نوع ضرب کننده است . در این ضرب کننده در کلاک با استفاده از یک بیت مضروب فیه ، یک حاصلضرب جزئی تولید و با مجموع فعلی جمع می شود ، پس می توان ضرب کننده ی سزیال را متشکل از چند قسمت اصلی تصور کرد. یک جمع کننده که بتواند در هرمرحله مضروب را با حاصل جمع ضرب های جزئی قبلی جمع کند . در هرمرحله کلاک، در صورتی که بیت فعلی مضروب فیه برابر 1 باشد مضروب باید با شیفت یافته ی حاصل جمع ضرب های جزئی جمع گردد و در صورتی که این بیت صفر باشد، تنها باید این حاصل جمع شیفت پیدا کند . بنابراین یک شیفت دهنده ، نیز یک مالتی پلکسر برای انتخاب بین جمع کردن و شیفت دادن ، و یا شیفت دادن تنها ، نیز مورد نیاز است.

همچنین باید تعداد مراحل نگه داشته شود تا از پایان عملیات - هنگامی که همه ی بیت های مضروب فیه بررسی شده و در نتیجه همه ی جمع های جزئی تولید و جمع شده اند - اطلاع حاصل شود .

در یک ضرب کننده ی سریال قراردادی ، از یک جمع کننده که یکی از وزودی های آن بیت های حاصلضرب جزئی و رودی دیگر آن توسط یک مالتی پلکسر بین صفر و مضروب فیه انتخاب می گردد. دسترسی به بیت های مضروب فیه و ذخیره ی بیت های

ساختار یک ضرب کننده ی سریال قراردادی به صورت شکل زیر است .

ایده هایی برای کاهش تو ان مصرفی

در اینجا سعی می کنیم با استفاده از ایده هایی فعالیت سوئیچینگ را درجاهایی که امکان دارد کاهش دهیم یکی از کارهایی که می توان به این منظور انجام داد حذف شیفت رجیستر است . در ضرب کننده ی قراردادی برای دسترسی به بیت های مضروب فیه یکی پس از دیگری ، و نیز ذخیره ی بیت های حاصلضرب از کم ارزش به پر ارزش از shift register استفاده شده است در این جا به جای شیفت رجیستر از یک Ring Counter و انتخابگر باس استفاده می شود این انتخابگر باس بر اساس این که کدام بیت Ring Counter فعال باشد یکی از بیت های مضروب فیه را انتخاب می کند و رقم محاسبه شده ی حاصلضرب نیز در یکی از لچ ها ذخیره می شود . و هنگامی که بیت پایانی ring counter فعال می شود به معنی پایان عملیات است . به این صورت به جای شیفت رجیستر و شمارنده یک ring counter قرار داده می شود که همان طور که خواهیم دید می توان آن را با توان مصرفی پایینی طراحی کرد. در ضمن در ابتدای عملیات باید به جای شیفت دادن محتوای رجیستر مقدار مضروب فیه در آن قرار داده شود که این کار توسط یک مالتی پلکسر انجام می گیرد .

در ضرب کننده ی قرار دادی یکی از ورودی های جمع کننده ، بیت های ضرب جزئی دیگری بسته به بیت فعلی مضروب فیه بین صفر و مضروب انتخاب می شود. و نتیجه در یک رجیستر ذخیره می شود. ضمن این که