

معماری‌هایی که برای کامپیوتر در سیستم‌های دیجیتال استفاده می‌شوند به ۴ دسته می‌توان تقسیم کرد:

۱) پردازنده‌های عام منظوره (general purpose) مثل پردازنده‌هایی که در کامپیوترها و موبایل‌ها استفاده می‌شوند که این‌ها به صورت خاص فن‌نویسن (von neuman) بودند

۲) Domain specific processor: این پردازنده‌ها برای دسته Domain خاصی از این‌ها استفاده می‌شوند مثل Domain‌های DSP, multi media و ...

۳) Application specific processor: برای این‌ها به منظور طراحی می‌شوند و کارایی می‌کنند

۴) Reconfigurable processor

The von neuman computer

نقطه‌های توهم کامپیوتری داشته باشیم به ساختار ساده داشته باشد و بتواند program را اجرا کند بدون آنکه نیاز به تغییر سخت‌افزار باشد در این جهت control unit مناسب‌ترین برنامه داشته باشیم

ساختار فن‌نویسن

یک memory داده که در آن سیگنال‌های عمل‌هایی که باید اجرا شوند قرار دارند، یک control path داده که در کنترل کردن عملیات در data path (ALU) را انجام می‌دهد، یک ALU (data path) داده که در جمع و تفریق و برجستگ‌های داخلی و خارج می‌شود

Control path ساختار از memory می‌شود و به ترتیب با آن‌ها داده‌ها را از ALU می‌گیرد

لیک برنامه محو ای از دستورات است که به صورت ترتیبی احصا می شوند که دستورات نامی می :
Year: _____ Month: _____ Day: _____ Subject: _____

- Fetch (IF) : یعنی دستوردهی از حافظه نمونه شده
- (d) decode : مشخص شدن چه عملیاتی باید انجام ده
- Read operand : آدرسها که عملیات روش باید انجام شده از حافظه نمونه شده
- Execute (Ex) : عملیات با نتایج آنرا انجامی که در ALU هستند باید احصا شود
- write : نتایج در حافظه نوشته شود

این مراحل برای هر دستور باید احصا شود

ضرورت کامپیوتر شدن نمونه Flexibility : لازمی دارد یعنی می توانیم حرکات را code کنیم و آن برای ما احصا کنند

حایب کامپیوتر شدن نمونه :

- کامپیوتری سرعت : چون دستورات باید به صورت ترتیبی احصا شوند از لحاظ سرعت efficient هستند
- کامپیوتری resource : منابع رایجی که در هر کامپیوتر احصا می کنند چون در واحد ALU ما ارباب های مختلفی است که می توانیم جمع می کنند ، می ضرب می کنند و ... و چون این دستورات سی سی احصا می شوند هر لحظه متقاضی از اینها داریم و باید به یکجا هستند

دستور سی سی حافظه : حافظه ها هستند دستور سی سی به یکجا هستند و ما باید بتوانیم حرکت از عملیات سی سی داخل CPU انجام می شود که این خودش یک لایه است و هر چند در اجرای کارش را سریع انجام می ده چون باید به حافظه دسترسی می دهیم و حافظه بسیار حساس است و باید به یکجا باشد

نظا : کامپیوتری از لحاظ انرژی : انرژی زیادی مصرف می کند به هر حسب 1000 ops در هر ثانیه

و باید به سیستم به چه قدرتی به این سرعت دستور سی سی می دهند

CACTUS

- روی سرعت خیلی کار نمی کنند و سرعت بالا را بالا می برند و پایلایین و بالا بردن سرعت کش

دستورات را prefetch کنند و ... که اینها دستورات گفته شده را بخورند و می‌توانیم آن دستورات را برداریم
 بی سوانح بخاری دسترسی رفته

Year: Month: Day:

Subject: **املا ۷**

- مشکل سرعت با صورتهای من نویسنده یک تصویر است به ما دستور العمل می‌خورد اجرا شده برای هر
 دستور باید در مرحله decode, fetch اجرا شده، اگر هر کدام از این کارها در یک cycle اجرا
 شود زمان اجرای این cycle ها t_{cycle} در نظر بگیریم یک دستور العمل $t_{cycle} \times 5$ طول می‌کشد
 به برای این سه دستور t_{cycle} طول می‌کشد
 - برای عملیات با لایف pipeline را اینجا در نظر بگیریم دستورات به صورت موازی انجام شوند که در هر
 یک دستور همان $t_{cycle} \times 5$ طول می‌کشد ولی به موازاتی که اجرای این سه دستور $t_{cycle} \times 4$ طول می‌کشد و این
 ترتیب در مدت زمان کوتاه‌تری این دستور العمل ها را می‌توانیم اجرا کنیم به بایلان دستورات را در عملیات branch
 در دستورات وجود داشته باشد که این توانیم دستورات را prefetch کنیم می‌توانیم زمان اجرای اینها را بخوریم
 و بایلان تا یک جری می‌تواند سرعت را بیشتر کند در حد ۱۳ تا ۱۴ برابر می‌تواند افزایش می‌دهد

- حتی بایلان می‌توانیم بایلان و cach و ... ما هم می‌توانیم این بخاری sequential است و دستورات
 ما در سری می‌توانیم انجام شوند حالا برای ضرب معکوس دو دو ما در درایه‌ها را می‌توانیم و عملیات ضرب را انجام دهیم

املا ۸

multi core / many core

برای افزایش سرعت می‌توانیم چند تا پردازنده می‌توانیم ما هم داریم و برای این می‌توانیم parallelism را اجرا
 کنیم چند تا core دارد ما هم می‌توانیم که بتوانیم همان ما هم داریم به صورت موازی می‌توانیم چند عملیات را
 اجرا کنیم به این ترتیب multi core و many core به وجود می‌دهند multi core برای وقتی استفاده
 می‌شود که تعداد core ها زیاد باشد و many core برای وقتی است که تعداد core ها زیاد باشد

این **multi core** و **many core** ها به یکدیگر می‌توانیم موازی سازی را **CACTUS**
 به انجام دهیم به core های که در آن سیستم وجود دارد ولی مشکلی که می‌تواند وجود دارد دسترسی به هر یک از

و Core ها باید ارتباط برقرار کنند. این ارتباط Core ها با هم در یک ترانسپون حافظه ای است که به نام System Memory است. برای این که دسترسی به حافظه راحت تر شود Local memory حافظه ای است که داخل هر Core قابل استفاده است. Shared memory برای این که ارتباط بین Core ها با سرعت خوبی بتوان برقرار کنند به اینها Year

ارتباط راحت تر می شود. دسترسی به حافظه ملی از دستورات سیستم های multi core و many core است که speed up به این جهت می رود است.

اسلامیہ
یک قانون به اسم Amdahl's law ارائه گردید که نشان میدهد که محدودیت افزایش سرعت در multi core در many core در چه حد است. رابطه speed up با تعداد core ها را مشخص میکند.
توجه: عامل محدود کننده در سیستم های multi و many است این است که جدیدترین می تواند موازی انجام شود و جدیدترین نام به صورت sequential انجام شود

$$\text{speed up} = \frac{1}{\text{serial} + \frac{1 - \text{serial}}{N}}$$

در موارد اینجانب، علاوه بر عددی speed up، در کنار هم تعداد core ها نیز، مثلاً در صفحه سبز اگر
۲۰ = serial و تعداد core ها ۶۴ باشد speed up مناسب برای می شود. در ضمن تعداد core ها
زیادتر شده خطای اندازه گیری سرعت ندارد و تغییر چندانی نمی کند.

اولاً: مخفی سازی به صورت حذف است. نشان می دهد که اسر serial part ها مخفی باشد و
ثانیاً: مخفی سازی به صورت انجام شود که آن وقت اسر ۶ بر رازیده داشته باشیم می توانیم اسر را سریعاً بررسی
کنیم. مخفی سازی در حجم مثل مخفی سازی است به صورت کلی

اسلام آباد ۲۰۱۵

حالت ۳
ما امیرک الملک داشته باشیم نه 88 کیلومتر از توهم موازی احبار کنیم و باید به صورت سریال انجام شود
و حد اکثر speed up که می توانیم در نظر بگیریم حدی است نهایت core داشته باشیم از ۱۳۴ تا ۱۴۰
CACTUS
تسلیت

- اگر می‌خواهیم برای یک میکروکنترلر speed up داشته باشیم (۵۰ برابر افزایش سرعت داشته باشیم)
و تعداد core خاصیم بماند باید ساختار پردازش را تغییر دهیم.

Year: Month: Day:

Subject: اسلاید ۱۱

«Domain specific processor»
برای حوزه‌های خاص برای یک دسته خاص از الگوریتم‌ها ساخته می‌شوند. ساختار اینها بخوبی منبسط شده است
اما data path را برای آن دسته از الگوریتم‌ها optimize کردن مثلاً برای کاربردهای DSP
در سیستم‌های MAC multiple accumulate می‌تواند استفاده می‌شود. منظور از عملیات MAC،
$$A \leftarrow A + (B * C)$$

است یا مثلاً $A = \sum B_i * C_i$ یا استفاده از عملیات MAC به طریقی است که
همه بایت‌ها operand می‌شوند و جمع می‌شوند و این عملیات می‌تواند به اتمام برسد و استفاده از MAC این عملیات
را یک فنر به صورت $A \leftarrow A + (B * C)$ در یک instruction انجام دهد بنابراین عملیات Fetch
و decode و غیره یکبار انجام می‌شود و نیازی نیست که در هر مرحله از اجرای آن دوباره این عملیات تکرار شود
- کارایی که می‌تواند بیش از ۱۰۰٪ باطریق‌های دیگر efficient انجام دهد یعنی instruction
خاص looping را انجام دهد که به صورت زیر است:

اسلاید ۱۲: اول اسلاید ۱۳ نوشته شده بود (۱۲)

کارهایی که توی loop انجام شود در یک حلقه می‌تواند منبسط شود.

۱) شمارنده counter loop را به‌صورت خودکار

۲) counter loop است و در هر مرحله باید شمارش شود

۳) برگرداندن به ابتدای loop

برای هر یک از اینها باید دستورات مقدری اجرا شود. در میکروکنترلر DSP انجام دارند
یعنی کمترین تعداد instruction را می‌خواهیم

اسلاید ۱۲

CACTUS

در منبسط شدن این MAC می‌تواند انجام شود.

داره سه مرحله عملیات $A \leftarrow A + (B * C)$ را انجام می‌دهد که اول عملیات Fetch و decode و ... برای

منتهی انجام می‌شود بدون عملیات برای جمع انجام می‌شود. پس چند cycle با بدای انجام شود تا یک MAC انجام شود. اگر ما دستور MAC داشته باشیم می‌توانیم با تعداد کمی cycle همان کار را انجام دهیم که کاری است که در DSP ها انجام می‌شود

Year: Month: Day:

اسلام ۱۴۸۸

یک نوع پردازنده‌های هستند اسم Application specific instruction set processor / ASIP این ASIP ها برای طراحی می‌شوند برای یک Domain یا اپلیکیشن خاص می‌توانند effition انجام دهند بنابراین می‌توانیم در دسته Domain specific دسته بندی کنیم. مثلاً ASIP های هستند برای Dsp و neural network

مثلاً process هست اسم tensa یا tensilica که سرعت بالا دارد، کارهای انجام می‌داد در طراحی ساختارهای سخت و مشخصات اپلیکیشن را از آنجا می‌گرفت و آنجا این اپلیکیشن ها را تحلیل می‌کردند و instruction های لازم را پیدا می‌کردند و آن را طراحی می‌کردند که در آن پردازنده‌ها می‌گذاشتند و سرعت خیلی بیشتر از پردازنده‌های عام می‌شود. کارهایی که انجام می‌دادند آن پردازنده‌ای که ایجاد می‌کردند هم می‌شد layout را باز می‌کردند هم bit stream را برای کاربرد خاص می‌توانست روی FPGا انجام می‌شد

اسلام ۱۴۸۹

Application specific processor

- سیستم‌های پردازشی که برای یک Function مشخص optimize می‌شوند

- Dsp و Domain specific ها معماری‌شان می‌تواند است و می‌تواند مدارهای application specific را دارند

در این ساختارها می‌تواند بهینه‌سازی در بخش می‌تواند و هر optimize را برای اون اپلیکیشن خاص می‌توانیم انجام بدهیم و معماری خاص آن را ارائه بدهیم

- یک نمونه Application specific integrated circuit / ASIC (حاکم کننده منطقی مدارهای خاص) که برای اپلیکیشن خاص طراحی می‌شوند و در اینجا از مدار سازی به عنوان خطی زیاده‌تر می‌توانیم و دستورات قوی‌تر و منطقی‌تر می‌توانیم

optimization را می توانیم با ساختار موازی که در chip ایجاد می کنیم به شدت بهبود دهیم
 - قطعی که وجود دارد که آن data path که برای منظم کردن قطار برای یک الیستش می تواند استفاده شود و این امر تقو
 یک الیستش داشته باشیم subject بهبود سرعت ما را می تواند داشته باشد
 Year: Month: Day:

- دستور العمل جدید لازم است Fetch و decode داشته باشد یعنی اینها را داشته می باشد
 - دسترسی به حافظه را می توان کم کرد و در حقیقت دسترسی به این دسترسی می تواند نیاز داشته و به این ترتیب می توانیم
 در ساختار جدید مستقیم دریافت و بفرستیم

املا ۱۷

حال ASIC

منظم کنیم یک دستور العمل الیستش داریم که عملیات این املا باید در دست انجام شده . برای این که این عملیات
 انجام شود حالت ۳ instruction فول می کشد ، اگر شرط اول برقرار نباشد یک تقاسیم داریم و یک جمع و یک
 ضرب ، اگر هم شرط برقرار نباشد یک تقاسیم داریم یک inc و یک dec ، به عنوان انجام عملیات
 حالت ۳ instruction فول می کشد در instruction ۵ تا ۱۵ cycle فول می کشد می کشد ۱۵ cycle
 اگر بیاوریم شده باشد می کشد cycle ۱۵ اما اگر این عملیات را به صورت ASIC انجام دهیم عمل است این
 عملیات متفاوت cycle فول می کشد ، این ترتیب که می کشیم تقاسیم باید انجام دهیم پس یک تقاسیم کشده
 فراموش دهیم ، یک جمع کشده یک ضرب کشده می داریم ، inc و dec داریم پس یک واحد inc و یک واحد
 dec هم می داریم و دو max هم می داریم که هر دو می a و b را به جامی دهد . در اینجا ما نرم افزار سازی
 را می توانیم داشته باشیم یعنی جمع و ضرب و inc و dec هم به صورت موازی می توان انجام شود و وقتی
 نتیجه compare شدن یک تا آخر چیزی نتیجه max هم مشخص می شود
 در نتیجه در این ساختار هم در دست موازی سازی را داریم هم Fetch و decode و ... را داریم
 دسترسی به حافظه ندارد و a و b را از رجیسترها می خواند و این را می سرعت بالایی دارند

« جلسه دوم »

املا ۱۷

ASIC ها هزینه ساخت I ها خیلی بالاست و باید هزینه زیادی توسط شرکت خاصی که می خوان از این آنها
 استفاده کنند پرداخت بجهت . هزینه خاصی که در اینجا لازم دارد به دو دسته تقسیم می شده :
 ۱) هزینه های سری ۲) هزینه های غیر سری

هزینه‌های غیر تکراری (NRE cost = Non Recurring Engineering cost) یعنی هزینه‌هایی که یکبار پرداخت می‌شود برای تولید chip که به کار برسم این هزینه فقط یکبار پرداخت می‌شود

هزینه‌های تکراری به ازای هر chip این هزینه را داریم: هزینه test، هزینه بایگ کردن، هزینه سیلوک

آیا هزینه طراحی NRE cost است؟ به ازای تعداد یکبار طراحی می‌کنیم. هزینه طراحی در مقابل هزینه ساخت IC خیلی ناچیز است

اسلاید ۲۰

یک هزینه خیلی بالایی برای ساخت IC است. در آن هزینه ساخت mask حساب می‌آید. هزینه design انجام می‌دهم layout را به صورت فایل جایگزین کارخانه ساخت می‌دهیم که آن کارخانه آن را به ما می‌دهد و نسوز mask تولید می‌کند. حالا برای تکنولوژی ۱۳۰nm حدود ۲۰۰۰ دلار باید پرداخت شد و هرچی که تکنولوژی پیشرفت کرده هزینه ساخت mask حاسم بالاتر رفته و فقط یکبار هزینه آن پرداخت می‌شود یعنی هزینه NRE cost است

پس NRE cost، ASIC، هزینه‌های مالااست. متراسته تعداد IC هایی که به بازار می‌خواهیم عرضه کنیم در هر چند هزار یا چند میلیون باشد. آن وقت NRE cost سرشان می‌شود و ای تعداد و چون هزینه متواکسار NRE پرداخت می‌شود، هر آن وقت هزینه‌ای که برای هر IC داریم کم خیلی کم است ولی به عنوان تعداد IC هایی که به بازار عرضه می‌شود در این حد مالااست

اسلاید ۲۱

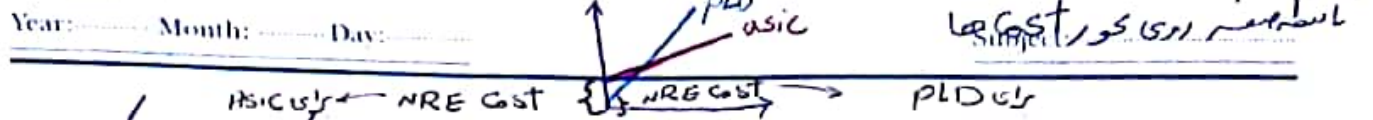
پس هزینه manufacturing cost ها که بعد از mask set cost به علاوه این design cost هم در حال افزایش است و به شدت در حال افزایش است علاوه بر این NRE cost هم در حال افزایش است. هزینه طراحی برای ASIC خیلی کم است و اگر هزینه برای طراحی نرم افزار یا FPGA است

از طرفی طول عمر محصولات دیجیتال رو به کاهش است. زیرا هر چقدر محصولات جدید به بازار می‌آید و مشتری ها محصولات جدیدتری را می‌خواهند و وقتی یک ASIC ساخته می‌شود به نایاب می‌گردد و باید دور انداخته شود و باید از اول ساخته شود اگر من خواهم ریزگرایی جدیدی ایجاد کنیم.

CACTUS
در نتیجه باعث شده تعداد wafer start حاسم شوند یعنی تعداد IC هایی که می‌توانیم به کار درآیم

برای PLD ها که یک نوع خاص از Fpg هستند یا شبیه بهتری افزایش پیدا می کند

NRE Cost در این معنی همیشه هم قابل برابر ASIC تا غیر برای PLD هم قابل برابر برای سرعت PLD



ASIC ها NRE Cost بالایی دارند ولی به ازای هر یک از این هزینه می دارد و شبیه بهتری از تمام

دارد ۰ در PLD به ازای هر محصول باید یک PLD غیرم
تفاوتی بین PLD و ASIC که در این مورد داریم، تفاوتی هم نیست که نشان می دهد اگر مشتری حاضر است
تعدادی نمونه بردارد و سرعت PLD به و سرعت ASIC نیز و اگر مشتری حاضر است تعدادی نمونه بردارد و سرعت ASIC

به و با PLD کاری نیست
اینست ۸۲۷
حرفه manufacturing است شده که هزینه اولیه برای ASIC ها بالاتر از و هزینه هر PLD

حجم از آن تر شده و این باعث شده که اول نقطه تقاطع حدودی به دست می آید و است به
و استفاده از ASIC ها کمتر شده و بیشتر از PLD ها استفاده کنند (نقطه ای که محور قبلی است و خطوط
سبز و زرد ای محور جدید است). ما این استفاده از ASIC ها در آینده انتظار شرکت ها است
برای مواردی ~ volume بالاست مثلاً cryptocurrency miner ها به عنوان صورت ASIC
می سازند چون می توانند مشتری زیادی دارد و به chip ها به عنوان صورت FPGA (PLD) تولید می شوند
در دست به عنوان سرعت ASIC می روند چون از لحاظ performance و Area و Power خیلی بهتر
از FPGA هستند ولی به دلایلی که در آینده می بینیم در به چلی از FPGA استفاده نمی شود

اینست ۸۸
افزایش سرعتی که سخت افزار در مقابل نرم افزار دارد ۰

یک سال به سال ۲۰۱۱ چاپ شده به سرعت ها فراوانی FPGA را در مقایسه با نرم افزار CPU بررسی کرده است
مثلاً در این شکل ۳۰٪ تا ۵۰٪ در image process در FPGA با CPU را نشان می دهد و در یک مشخص می کند
چند برابر افزایش سرعت داشته است. مثلاً علامت ۴۶ نشان می دهد که ۴۶ برابر افزایش
سرعت داشته است نسبت به CPU (یا این نمودار نشان داده شده که هر علامت ۱۰ برابر است)
می بینیم که افزایش سرعت خیلی بالاست و ما حدود ۲۸۵۰۰۰ افزایش سرعت داشته ایم که به این
CACTUS دلیل است که اینها سخت افزاری می تواند افزایش سرعت های خیلی زیادی داشته باشد

و نشان می دهد برای سازی بالا می دارند

یک مثال دیگر که آورده کاربردهای خاص را در انواع سیستم های سخت افزاری است که این کار در

کاربرد image processing و SR (super resolution) بوده
Year: _____ Month: _____ Day: _____

SR کارش اینست که چندتا image با رزولوشن پایین (low-resolution) را می بیند و از روشی که می بیند
high resolution را می سازد

- این کاربرد اینست که ممکن است افزایش سرعت برامون مهم باشه مثلاً در یک ریدر تعداد زیادی از image را
در زمانیکه پردازش کنیم و اکثر سیستم های که برای این کار هست از CPU، GPU، و FPGA بررسی کردند

که در این مورد کار خورافه سازی از این image را نشان میده و می گوید زمان کابسته ای است که عملیات SR
می خواهد انجام دهد و خوب می باشد نشان دهنده سرعت بالای آن است

می بینیم که FPGA حاصلی بهتر از CPU حاصل می کنه و در حدی حاصلی بهتر از GPU حاصل می کنه
(البته برای این کاربرد این بهتر است برای کاربرد دیگر ممکن است مثلاً CPU بهتر از FPGA و ... باشد
یا حتی CPU بهتر از FPGA باشد ...)

حالا بنگریم به FPGA و CPU در این اوضاع حدود ۵۰ هزار برابر بهتر هسته در اسلايد ۲۸ هم می بینیم
speedup حدود ۱۰ هزار برابر داریم پس دنیا AMDAL's نه می تونه مثلاً ۵۰ برابر می توونه افزایش سرعت
داشتن باشه ما برای ۹۸٪ بریم و هیچ جوجهایی روی اینطوری نیست ۱۰ هزار برابر می توونه speedup
داشتن باشه چه جوری؟

۱. اتصال وجود داره - این اندازه من باشه درای کامنت parallel تا سینه ایام دارند فقط و اینها مقدار زیادی
نیست و یک مقدار دستکاری شده در مقاله نوشته اند

۲. بعضی از اطلاعات کاروا رفا من serial part می دارند و فقط بخش کابسته ای که می تونه موازی هم هست
مثلاً در نظر گرفته اند و چون بخش سریال خیلی کوچیک است اصلاً در نظر گرفته اند (مثلاً در یک ده هزارم درصد است)

از لحاظ energy efficiency و power efficiency این منحنی نشون دهنده تفاوت سخت افزاری

است! برای کم کردن ابعاد سیستم های سخت افزاری را از لحاظ energy efficiency
CACTUS

بر حسب mops/mw می کشند نه نشان دهنده سرعت نیست آهسته بر حسب مقدار توان مصرفی است

در بحث energy efficiency در نظر گرفتن مصرف انرژی این اسلاید ایجاد شده و مقایسه برقرار می‌گردد
microprocessor ها حدود ۵ تا ۸۰ امپا دارند ، ASIC ها در حدود ۱ تا ۲۰ امپا و DSP ها هم

آن در سطح آمپا دارند که این مقیاسی است که می‌تواند microprocessor ها
Year: / Month: / Day: /

از لحاظ کارایی انرژی می‌توان گفت که ASIC ها از لحاظ مصرف انرژی بهتر هستند. در بحث انرژی مصرفی داریم سه سری خواص
انرژی بالایی مصرف نمی‌کنند microprocessor ها هم این را می‌کنند و سرعت ASIC و DSP ها هم
در بحث performance می‌توانیم بگوییم که بسته به این که چه کاری می‌خواهیم انجام دهیم. FPGA ها هم از لحاظ انرژی و power
power مصرف می‌کنند. FPGA از لحاظ کارایی و مصرف انرژی و CPU ها هم از لحاظ کارایی

اسلاید ۳۲

تقریباً بین CPU و DSP و FPGA از لحاظ مصرف انرژی Development (از لحاظ مصرف انرژی می‌توانیم بگوییم)
CPU ها خیلی راحت تر هستند برای پیاده سازی توسط توسعه دهنده و در مقابل سیستم های DSP ها هم مصرف انرژی می‌کنند
و در بحث کارایی انرژی و در بحث DSP ها از لحاظ مصرف انرژی و در بحث FPGA ها هم مصرف انرژی می‌کنند
امداد هم سال های مختلف این Gap ها کمتر شده و به هم نزدیک می‌شوند و برای CPU ها هم مصرف انرژی می‌کنند و multi-core
و CPU ها هم مصرف انرژی می‌کنند و در بحث مصرف انرژی می‌توانیم بگوییم که

اسلاید ۳۳

موقع بندی مطالب گفته شده

اسلاید ۳۴

من بگویم:

Flexibility زیادی وجود دارد، speed و energy داشتن و مدیریت روی program و شبای که
وجود داشته چون برنامه باید خودش را با خودش adapt کند این را می‌توانیم بگوییم که برنامه خودش را تغییر می‌دهد

DSP

Flexibility و energy داشتن و مدیریت روی program و شبای که
وجود داشته چون برنامه باید خودش را با خودش adapt کند این را می‌توانیم بگوییم که برنامه خودش را تغییر می‌دهد

ASIC

از لحاظ سرعت، استفاده از منابع سخت افزاری و Area های efficiently هستند و در بحث مصرف انرژی هم
المنی که هستند مناسب هستند در بحث مصرف انرژی و در بحث Flexibility ندارند

CACTUS

در ASIC سخت افزار باید تغییر داده بشود به Application یا چون برنامه ما تغییر می‌دهد این را می‌توانیم بگوییم که