

سخنرانی: 11 تعبیه شده

پردازنده ها

سید حسین عطارزاده نیاکی

بر اساس اسلایدهای P. Marwedel

سیستم های زمان واقعی جاسازی شده

1

بررسی کنید

MoC • واکنشی همزمان

MoC • های زمان بندی شده

-مدل با زمان راه اندازی

-مدل رویداد گسسته

-جزر و مد

سیستم های زمان واقعی جاسازی شده

2

طرح کلی

• انواع واحدهای پردازشی • کارایی پردازنده های تعبیه

شده

- بهره وری انرژی/انرژی

- کارایی اندازه کد

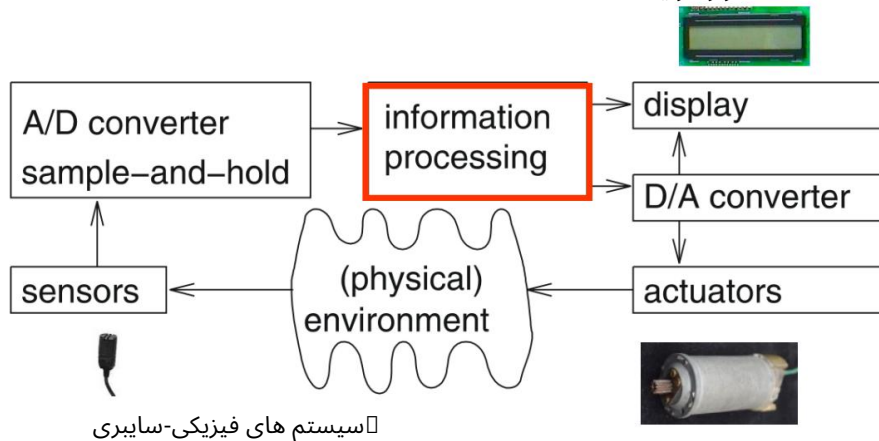
- راندمان زمان اجرا • قابلیت بلادرنگ

سیستم های زمان واقعی جاسازی شده

3

سخت افزار سیستم جاسازی شده و CPS

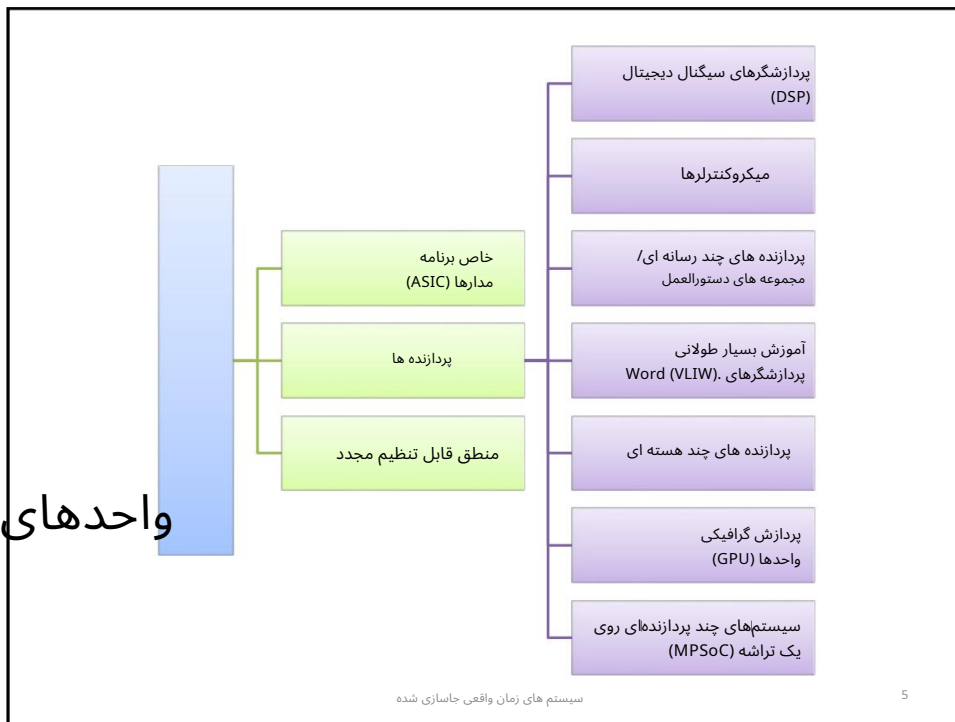
• سخت افزار سیستم جاسازی شده اغلب در یک حلقه استفاده می شود
("سخت افزار در یک حلقه"):



سیستم های زمان واقعی جاسازی شده

4

واحدهای پردازش



مدارهای خاص برنامه (ASIC) یا مدارهای سفارشی کامل

• رویکرد از

-زمان طراحی طولانی،

-عدم انعطاف پذیری (تغییر استانداردها) و

-هزینه های بالا (به عنوان مثال آسیاب. \$ هزینه ماسک). • مدارهای

سفارشی طراحی شده لازم است

-اگر سرعت نهایی یا

-بهره وری انرژی هدف است و

□ سنتز HW در این دوره پوشش تعداد زیاد قابل فروش است.

ها نگاه کنیم

سیستم های زمان واقعی جاسازی شده

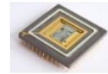
6

کارایی: برای پردازش اعمال شد

CPS - و ES باید کارآمد باشند



• اندازه کد کارآمد
(به خصوص برای سیستم های روی تراشه)



• زمان اجرا کارآمد



• وزن کارآمد



• مقرون به صرفه



• انرژی کارآمد

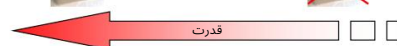


سیستم های زمان واقعی جاسازی شده

7

چرا به بهره وری انرژی اهمیت می دهیم؟

پلت فرم اعدام	در حین استفاده مرتبط است؟		
	Plugged	Uncharged	ged
	دوره های قطع برق		
	به عنوان مثال کارخانه		سنسور ماشین
گرم شدن کره زمین	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
هزینه انرژی	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
افزایش عملکرد	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
مشکلات خنک کننده، اجتناب از نقاط داغ	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
اجتناب از جریان های زیاد و مهاجرت فلزات	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
قابلیت اطمینان	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
انرژی یک منبع بسیار کمیاب است	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



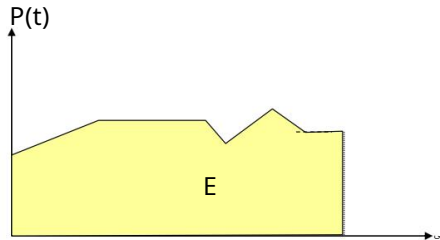
سیستم های زمان واقعی جاسازی شده

8

آیا باید به انرژی اهمیت دهیم؟ مصرف یا مصرف برق؟

$$E = \int P(t) dt$$

طراحی منبع تغذیه و رگولاتور



-ابعاد اتصال، خنک کننده کوتاه مدت

• به حداقل رساندن مصرف انرژی
مهم به دلیل

-دسترسی محدود به انرژی (سیستم های تلفن همراه)

-سرمایش: هزینه های بالا، فضای محدود

-اثرات حرارتی

-قابلیت اطمینان، طول عمر طولانی



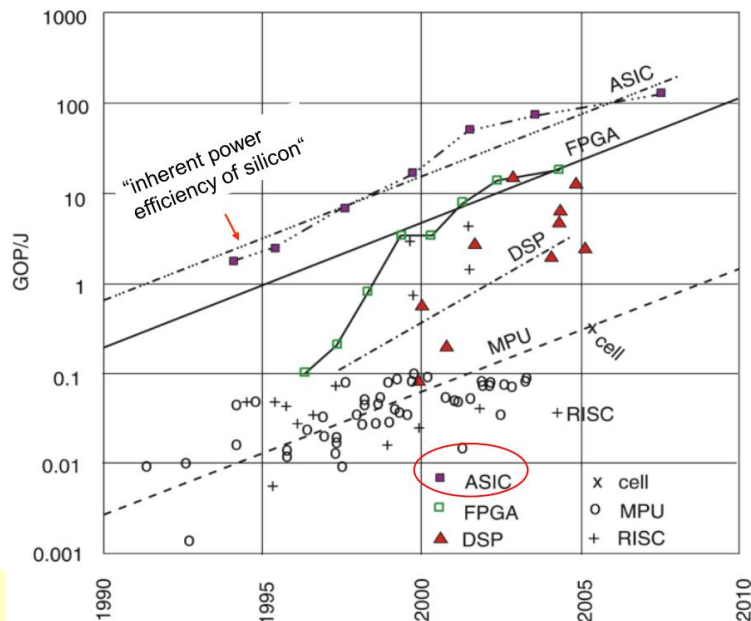
تابه طور کلی، ما باید به هر دو اهمیت دهیم

هر دو ارتباط نزدیک دارند،
اما هنوز متفاوت است

سیستم های زمان واقعی جاسازی شده

9

انرژی
بهره وری از
مختلف
پلتفرم های
هدف



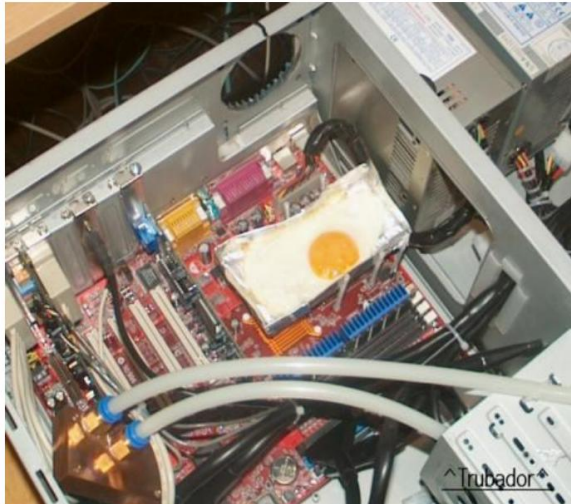
© هوگو دی من،
IMEC، فیلیپس، 2007.

سیستم های زمان واقعی جاسازی شده

10

رایانه های شخصی: از بشقاب داغ (آشپزخانه) پیشی گرفت...؟

چرا از آن استفاده نمی کنید؟



به بیان دقیق، انرژی
"مصرف" نمی شود،
بلکه از انرژی الکتریکی
به انرژی گرمایی تبدیل
می شود

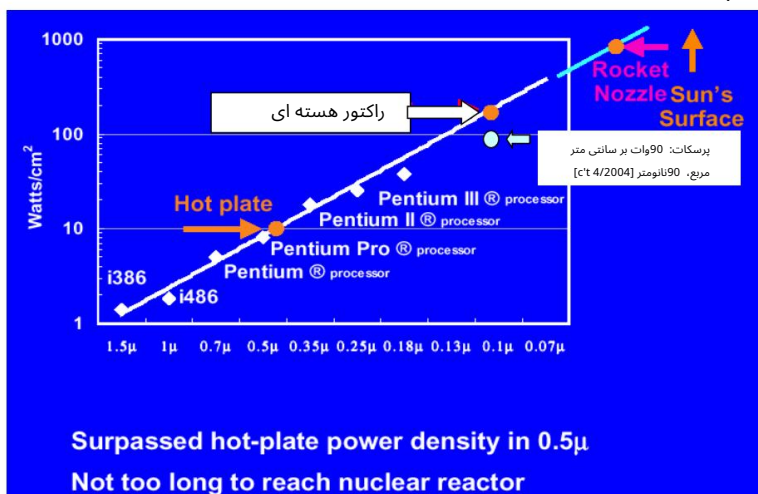
http://www.phys.nku.edu.tw/~htsu/humor/fry_egg.html

سیستم های زمان واقعی جاسازی شده

11

رایانه های شخصی

مشکل: چگالی توان افزایش می یابد

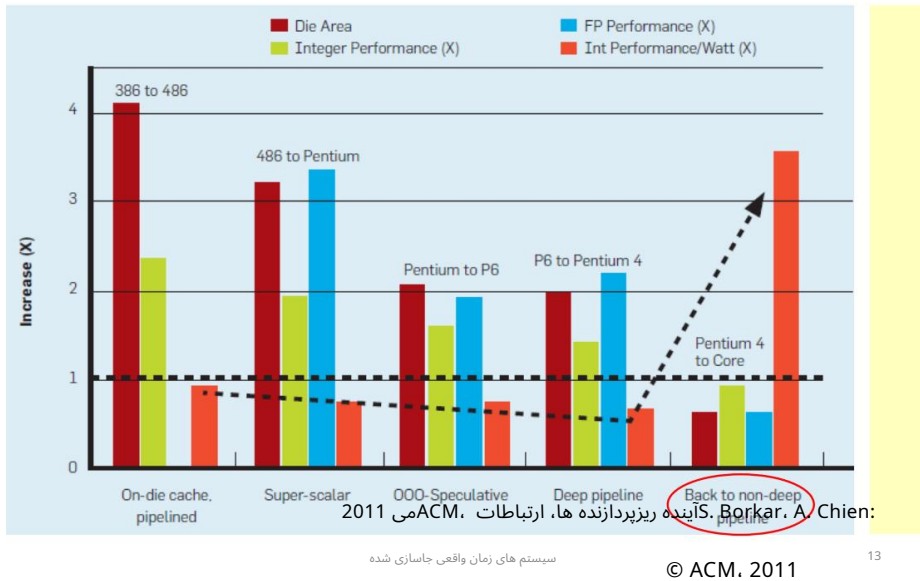


© اینتل
Micro-32
M. Pollack.

سیستم های زمان واقعی جاسازی شده

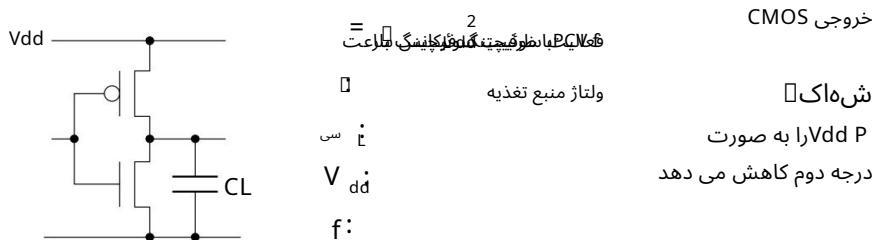
12

ساده نگه دار، احمقانه (KISS)



مصرف برق استاتیک و دینامیک

-مصرف انرژی پویا: مصرف برق ناشی از شارژ خازن ها در هنگام تغییر سطوح منطقی.



مصرف برق ساکن (ناشی از جریان ناشی): برق مصرف شده در غیاب سیگنال ساعت

مصرف برق استاتیک و دینامیک

مصرف برق مدارهای CMOS (بدون توجه به نشتی):

$$P_{CV} = \frac{1}{2} C_L V_{dd}^2 f$$

سوئیچینگ بار فعالیت

سوئیچینگ ظرفیت منبع

تغذیه فرکانس ساعت ولتاژ: V_{dd}

f :

تاخیر برای مدارهای CMOS:

$$\tau = k C_L \frac{V_{dd}}{(V_{dd} - V_{th})^2}$$

آستانه از V_{th}

$$V_{th} \propto V_{dd}$$

به عنوان مثال، اگر V_{dd} دو برابر شود، تاخیر τ یک چهارم می‌شود. در حالی که مصرف انرژی P_{CV} چهار برابر می‌شود. این نشان می‌دهد که افزایش ولتاژ به نفع مصرف انرژی نیست.

سیستم‌های زمان واقعی جاسازی شده

15

ساخت پردازنده‌های انرژی کارآمد

سه تکنیک

- اجرای موازی

- مدیریت انرژی پویا (DPM)

- مقیاس بندی ولتاژ و فرکانس دینامیکی (DVFS)

سیستم‌های زمان واقعی جاسازی شده

16

ولتاژ پایین، عملکرد موازی کارآمدتر از ولتاژ بالا،

متوالی

معادلات پایه

$$P \sim VDD^2 f$$

حداکثر فرکانس ساعت: $f \sim VDD$

انرژی برای اجرای یک برنامه: $E = P \cdot t$ (ثابت)

زمان اجرای برنامه: $t \sim 1/f$

تغییرات ناشی از پردازش موازی، با

فرکانس ساعت به:

ولتاژ را می توان به موارد زیر کاهش داد:

قدرت پردازش موازی: $P' = P \cdot t$

قدرت برای: $P' = P \cdot t$

زمان اجرای یک برنامه هنوز است:

انرژی مورد نیاز برای اجرای برنامه:

تقریب های

خشن!

استدلال به نفع مقیاس ولتاژ،

و پردازش موازی

سیستم های زمان واقعی جاسازی شده

17

مدیریت توان پویا (DPM)

مثال: STRONGARM SA1100

• RUN: عملیاتی

• IDLE: یک روال SW ممکن است

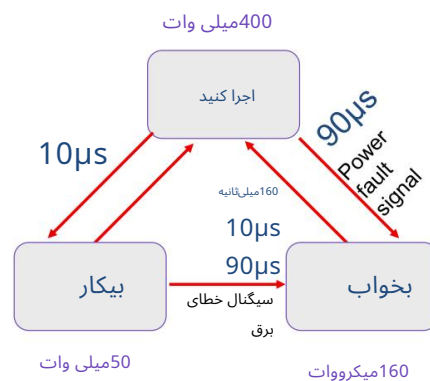
CPU را هنگامی که استفاده نمی شود

متوقف کند، در حالی که نظارت قطع

می شود

• SLEEP: خاموش شدن روشن-

فعالیت تراشه

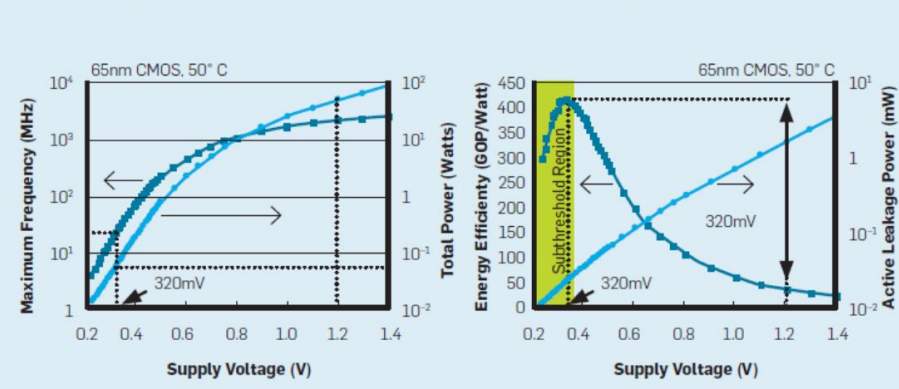


سیستم های زمان واقعی جاسازی شده

18

مقیاس ولتاژ: مثال

Figure 13. Improving energy efficiency through voltage scaling.



© ACM, 2011

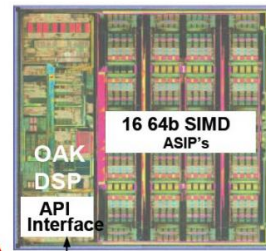
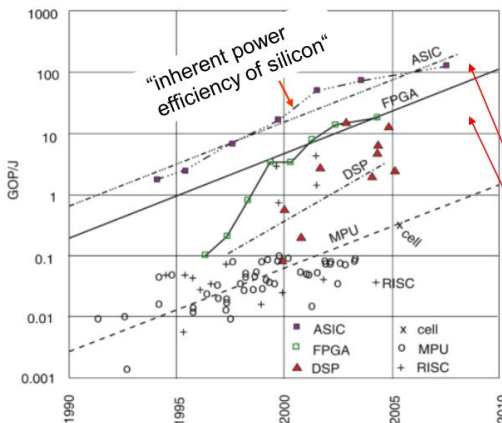
S. Borkar, A. Chien: آینه ریزپردازنده ها، ارتباطات، ACM، می 2011

سیستم های زمان واقعی جاسازی شده

معماری های کم مصرف تر:

دامنه و برنامه خاص

VIP for car mirrors
Infineon



200MHz, 0.76 Watt
100Gops @ 8b
25Gops @ 32b

نزدیک به راندمان
قدرت سیلیکون

© Hugo De Man: From the Heaven of Software to the Hell of Nanoscale
ACACES, 2007. فیزیک: صنعتی در گذار، اسلایدهای کلیدی.

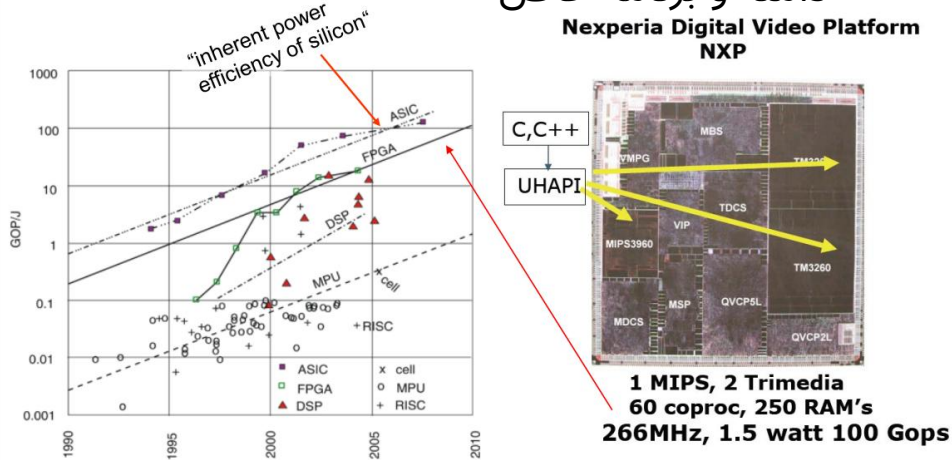
سیستم های زمان واقعی جاسازی شده

20

معماری های کم مصرف:

دامنه و برنامه خاص

Nexperia Digital Video Platform NXP



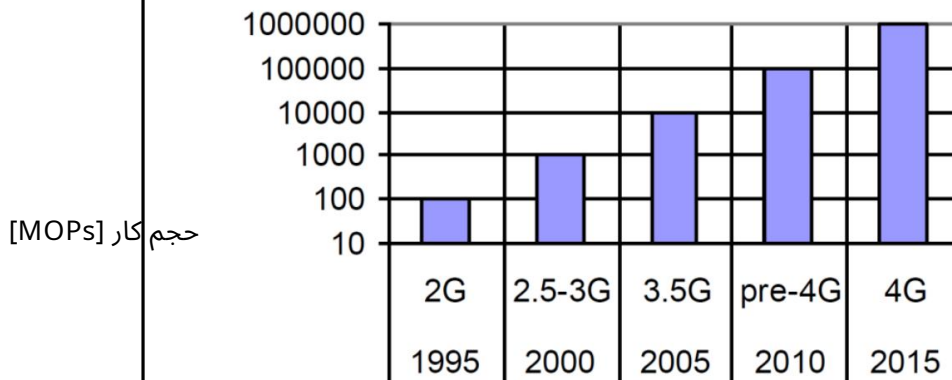
سیستم های زمان واقعی جاسازی شده

21

نزدیک به راندمان
قدرت سیلیکون

تلفن های همراه: افزایش الزامات عملکرد

CH van Berkel: Multi-Core for Phones, DATE, 2009;



بسیاری از موارد دیگر از مشکل برق/انرژی

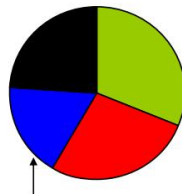
سیستم های زمان واقعی جاسازی شده

22

تلفن های همراه:

قدرت کجا می رود؟

• استفاده از تلفن همراه، تفکیک بر اساس نوع محاسبات



- گرافیک
- رسانه ها
- رادیو
- برنامه

پردازشی هندسه، شطرنجی سازی، سایه زنی پیکسل)
(پردازش نمایش و دوربین، (دکدگذاری ویدیو)
(پردازش آند، دمودولاسیون، رمزگشایی، پروتکل)
(لینک کاربری، مرور و ...)

با هدف ویژه HW!

Berkel: Multi-Core for Phones, DATE, 2009;
CH van (بدون درصد صریح در مقاله اصلی)

□ در حین استفاده، همه اجزا و محاسبات مرتبط هستند

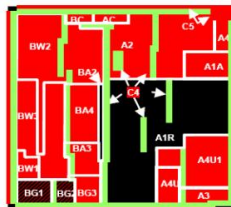
سیستم های زمان واقعی جاسازی شده

23

معماری های کم مصرف:

پردازنده های ناهمگن

(2)Telephony (W-CDMA)



■ Power on
■ Power off

Baseband part	Control	ON
	W-CDMA	ON
	GSM	ON / OFF
Application part	System-domain	ON
	Realtime-domain	OFF
Measured Leakage Current (@ Room Temp, 1.2V)		407 μ A

<http://www.mpsoc-forum.org/2007/slides/Hattori.pdf>

15 ©2007, Renesas Technology Corp., All rights reserved.

MPSoC '07

Everywhere you imagine. RENESAS

□ "سیلیکون تیره" (به دلیل محدودیت های جریان، قدرت یا دما، همه سیلیکون ها را نمی توان همزمان تامین کرد)

سیستم های زمان واقعی جاسازی شده

24

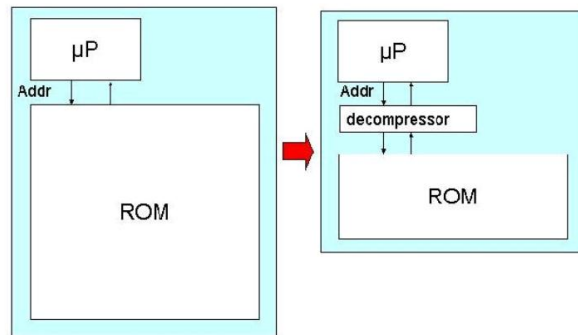
نیاز کلیدی شماره 2: کارایی در اندازه کد

• ماشین های CISC

• تکنیک های فشرده سازی: ایده کلیدی

-نمای کلی: <http://www.perso.iro.umontreal.ca/~latendre/codeCompression/codeCompression/node1.html>

<http://www.perso.iro.umontreal.ca/>



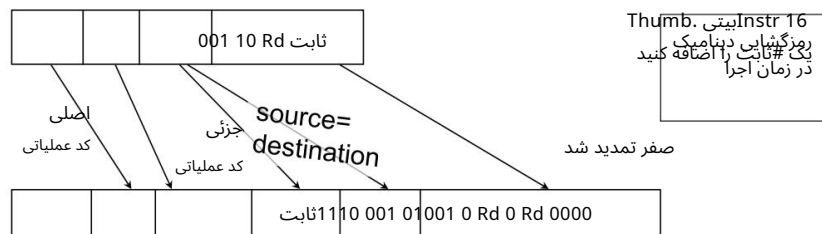
سیستم های زمان واقعی جاسازی شده

25

کارایی در اندازه کد

-تکنیک های فشرده سازی (ادامه):

• مجموعه دستورالعمل دوم ، به عنوان مثال مجموعه دستورالعمل ARM Thumb:



• کاهش به 65-70 درصد از اندازه کد اصلی

• 130% عملکرد ARM با حافظه 8/16 بیتی

• 85% عملکرد ARM با حافظه 32 بیتی

رویکرد مشابه برای ... LSI TinyRisc،

نیاز به پشتیبانی توسط کامپایلر، اسمبلر و غیره دارد.

سیستم های زمان واقعی جاسازی شده

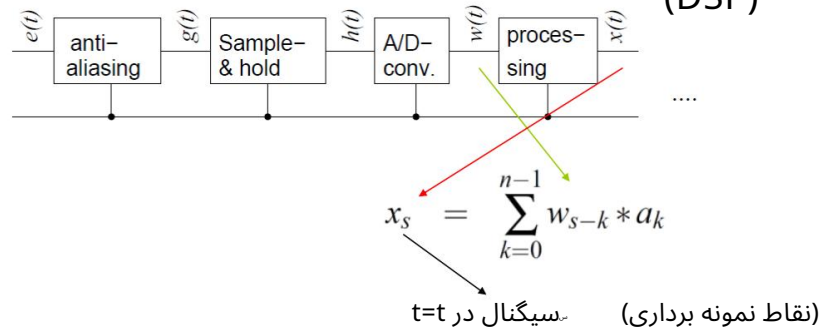
[ARM, R. Gupta]

26

نیاز کلیدی شماره 3: راندمان زمان اجرا

• معماری های دامنه گرا

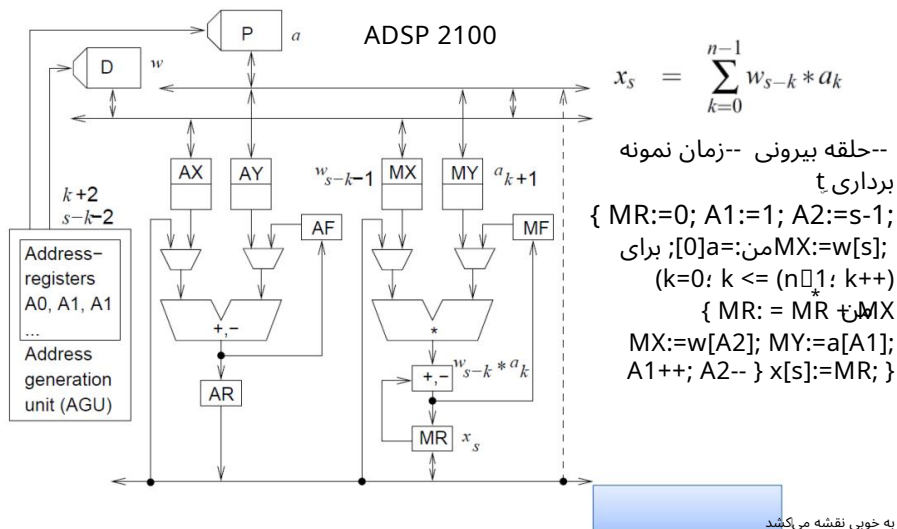
• مثال: فیلتر کردن در پردازش سیگنال دیجیتال (DSP)



سیستم های زمان واقعی جاسازی شده

27

فیلتر کردن در پردازش سیگنال دیجیتال



سیستم های زمان واقعی جاسازی شده

28

پردازنده های DSP

• دستورالعمل های ضرب/انباشت (MAC) و حلقه سربار صفر (ZOL).

;[0]a=من:MR:=0; A1:=1; A2:=s-1; MX:=w[s];

برای (k:=0 <= n-1)

{MR:=MR+MX*MY; MY:=a[A1]; MX:=w[A2]; A1++; A2--}

دستور ضرب/انباشت (MAC)

دستورالعمل حلقه سربار صفر (ZOL) قبل از
دستورالعمل MAC.

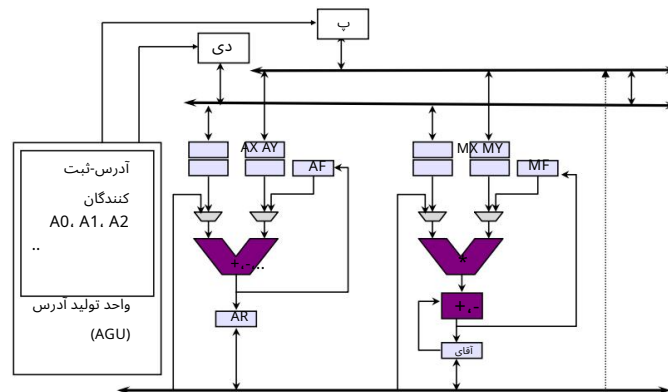
تست حلقه به موازات عملیات MAC انجام
می شود.

سیستم های زمان واقعی جاسازی شده

29

رجیسترهای ناهمگن

مثال: (ADSP 210x)



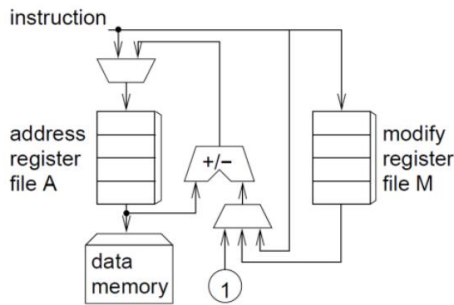
عملکردهای مختلف رجیسترهای An, AX, AY, AF, MX, MY, MF, MR

سیستم های زمان واقعی جاسازی شده

30

واحدهای تولید آدرس جداگانه (AGUs)

مثال: (ADSP 210x)



حافظه داده فقط می تواند باشد
واکشی شده با آدرس موجود در A.

اما این را می توان به موازات عملیات در مسیر
داده اصلی انجام داد (به طور موثر 0 زمان می برد).

$A := A \pm 1$ نیز 0 زمان می برد،

برای $A := A \pm M$ ؛

$A :=$ فوری در دستورالعمل <

نیاز به آموزش اضافی دارد

فورا بار را به حداقل برسانید

بهینه سازی بعدا انجام می شود

سیستم های زمان واقعی جاسازی شده

31

آدرس دهی مائول

آدرس دهی مائول:

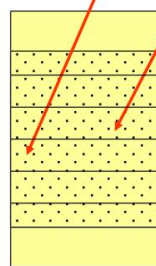
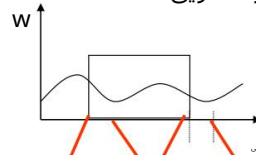
$$Am++ \quad Am := (Am+1) \bmod n$$

(بافر حلقه ای یا دایره ای را در حافظه پیاده
سازی می کند)

جدیدترین
مقادیر

$w[t1-1]$
 $w[t1]$
 $w[t1-n+1]$
 $w[t1-n+2]$
..

پنجره کشویی



حافظه، $t=t1$

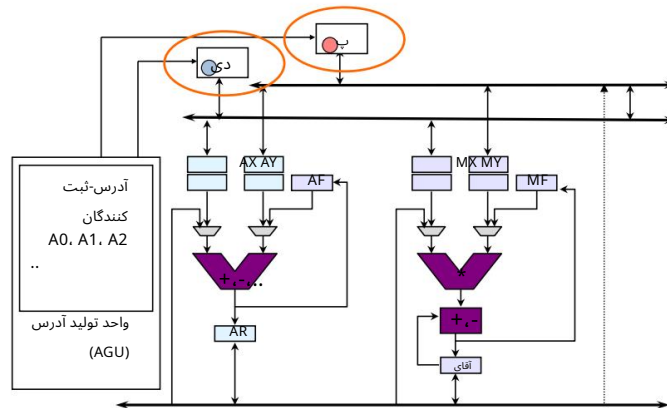


حافظه، $t2=t1+1$

سیستم های زمان واقعی جاسازی شده

32

چندین بانک حافظه یا حافظه



واکشی موازی را ساده می کند

سیستم های زمان واقعی جاسازی شده

33

اشباع حسابی

در صورت وجود، بزرگترین/کوچکترین عدد را برمی گرداند
سرازیر/زیر جریان

مثال:

الف

0111

10001 + بندی استاندارد حول حساب (1)0000

اشباع حساب 1111

1000: $(a+b)/2$ صحیح است

دور حسابی 0000 پیچید

محاسبات اشباع 0111 + جابجا شد

"تقریباً درست"

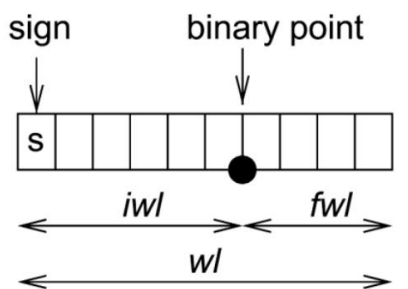
مناسب برای برنامه های DSP/چند رسانه ای:

- عدم به موقع بودن نتایج در صورت ایجاد وقفه برای سرریزها
- مقادیر دقیق کمتر اهمیت دارند
- محاسبات را بدتر کنید.

سیستم های زمان واقعی جاسازی شده

34

محاسبات نقطه ثابت



جابجایی پس از ضرب و تقسیم برای حفظ نقطه باینری لازم است.

سیستم های زمان واقعی جاسازی شده

35

قابلیت بلادرنگ

• رفتار زمان بندی باید قابل پیش بینی باشد
ویژگی هایی که باعث ایجاد مشکل می شود:

- دسترسی غیرقابل پیش بینی به منابع مشترک
• حافظه پنهان با استراتژی های جایگزینی برای پیش بینی دشوار • حافظه های پنهان یکپارچه (تعارض بین دستورالعمل ها و داده ها) • خطوط لوله با پیش بینی چرخه های توقف دشوار ("جاب ها") • زمان های ارتباط غیرقابل پیش بینی برای چند پردازنده ها

- پیش بینی شعبه، اجرای حدس و گمان - وقفه هایی که در هر زمان ممکن است - حافظه به روزرسانی می شود که در هر زمان ممکن است - دستورالعمل هایی که زمان های اجرای وابسته به داده ها دارند □ تلاش برای جلوگیری از بسیاری از این موارد.

[کارگاه آموزشی داکستول در مورد پیش بینی پذیری، 19-17 نوامبر 2003]

سیستم های زمان واقعی جاسازی شده

36

پردازنده های تعبیه شده برای

کاربردهای بلادرنگ بحرانی ایمنی • الزامات بالا از نظر قابلیت پیش

بینی زمان

-کران های پایین و بالایی در زمان اجرای کار

BCET و WCET نامیده می شود

-باید ایمن و محکم باشد • تهدیدات قابل پیش بینی

-ویژگی های معماری

-نرم افزار

-سطح وظیفه

-عملیات توزیع شده

-لایه متقاطع

سیستم های زمان واقعی جاسازی شده

37

میکروکنترلرها

مثال: Intel 8051

CPU 8 • بیتهای، بهینه سازی شده برای برنامه های کاربردی، • مجموعه وسیعی از عملیات بر

روی انواع داده های بولی، فضای آدرس برنامه 64 کیلوبایت، • فضای آدرس داده جداگانه

• 64 کیلوبایت، 4 • کیلوبایت حافظه برنامه روی تراشه، 128 بایت حافظه داده روی تراشه،

32 • خط ورودی/خروجی، که هر کدام به صورت جداگانه قابل آدرس دهی هستند، 2 •

شمارنده روی تراشه، • ناهمزمان جهانی گیرنده / فرستنده برای خطوط سریال

موجود بر روی تراشه، • تولید ساعت بر روی تراشه، بسیاری

از تغییرات تجاری موجود است.

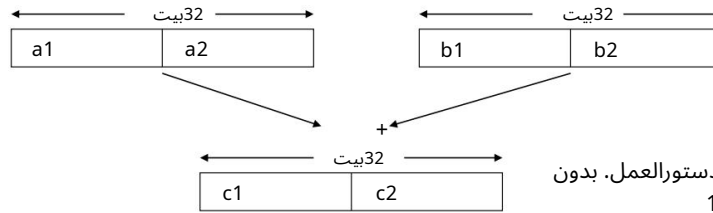
•

سیستم های زمان واقعی جاسازی شده

38

دستورالعمل‌های چند رسانه‌ای، پسوندهای برداری کوتاه، پسوندهای جریانی، دستورالعمل‌های SIMD

دستورالعمل‌های چند رسانه‌ای از بسیاری از ثبات‌ها سوء استفاده می‌کنند،
جمع‌کننده‌ها و غیره بسیار گسترده هستند (32/64 بیت)، در حالی که بیشتر انواع داده
های چند رسانه‌ای باریک هستند
2-8 مقدار را می‌توان در هر ثبات ذخیره و اضافه کرد. به عنوان مثال:



2 اضافه در هر دستورالعمل. بدون
حمل در بیت 16

روش ارزان استفاده از موازی سازی
پسوندهای مجموعه دستورالعمل SSE، دستورالعمل‌های SIMD

سیستم‌های زمان واقعی جاسازی شده

39

پردازنده‌های چند هسته‌ای ناهمگن تک ISA به عنوان مثال ARM's big.LITTLE

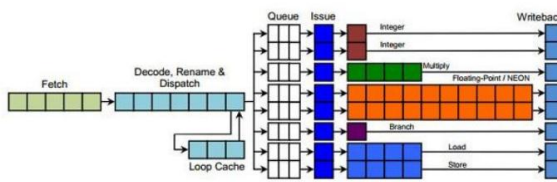


Figure 2 Cortex-A15 Pipeline

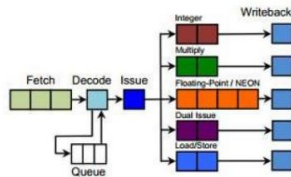
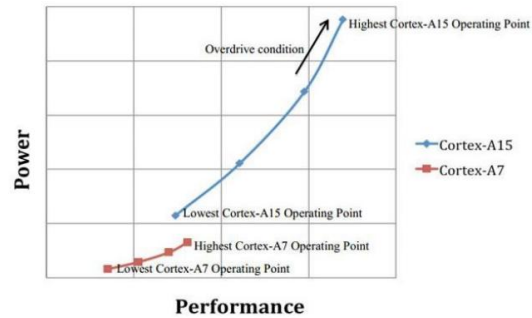


Figure 1 Cortex-A7 Pipeline



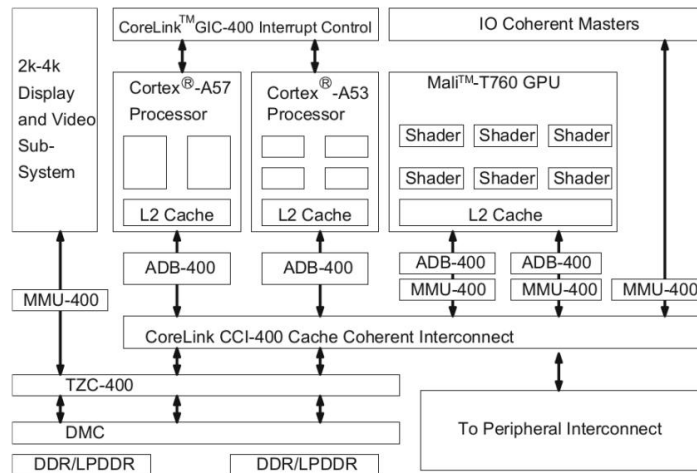
سیستم‌های زمان واقعی جاسازی شده

© ARM. 2013

40

سیستم‌های چند پردازنده‌ای روی یک تراشه (MPSoC)

ARM® big.LITTLE سیستم روی تراشه (SoC)

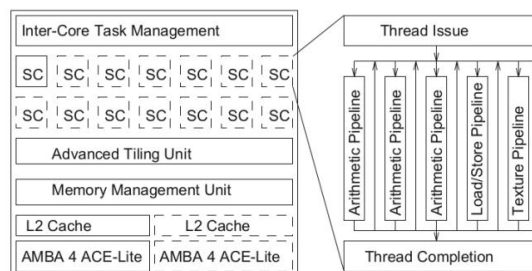


سیستم‌های زمان واقعی جاسازی شده

41

واحدهای پردازش گرافیکی (GPU)

GPU • های قابل برنامه ریزی • اجرای همزمان بسیاری از رشته های ریز دانه • بهره وری انرژی مهم در سیستم های جاسازی شده • رابط به OpenGL، OpenCL و غیره.



ARM® Mali TM
پردازنده گرافیکی T880

سیستم‌های زمان واقعی جاسازی شده

42

واحدهای پردازش عصبی ARM

- طبقه‌بندی شیء • تشخیص شیء • تشخیص/تشخیص
- چهره • تشخیص ژست انسان/تشخیص حرکت دست •
- تقسیم‌بندی تصویر • زیباسازی تصویر • وضوح فوق‌العاده •
- تنظیم فریم (فوق آهسته) • تشخیص گفتار • تشخیص صدا
- حذف نویز • سنتز گفتار • ترجمه زبان •

Key Features		Ethos-N78	Ethos-N77	Ethos-N57	Ethos-N37
	Performance	10.5, 2, 1 TOP/s	4 TOP/s	2 TOP/s	1 TOP/s
	MAC/Cycle (8x8)	40v6, 2048, 1024, 512	2048	1024	512
	Efficient convolution	Winograd support delivers 2.25x peak performance over baseline			
	Configurability	90+ Design Options	Single Product Offering		
	Network support	CNN and RNN			
	Data types	Int-8 and Int-16			
	Secure mode	TEE or SEE			
	Multicore capability	8 NPUs in a cluster 64 NPUs in a mesh			
Memory System	Embedded SRAM	384KB - 4MB	1-4 MB	512 KB	512 KB
	Bandwidth reduction	Enhanced Compression	Extended compression technology: layer/operator fusion, clustering, and workload tiling		
	Main interface	1xAXI4 (128-bit), ACE-5 Lite			
Development Platform	Neural frameworks	TensorFlow, TensorFlow Lite, Caffe2, PyTorch, MxNet, ONNX			
	Inference deployment	Ahead of time compiled with TVM Online interpreted with Arm NN Android Neural Networks API (NNAPI)			
	Software components	Arm NN, Arm NPU software (compiler and support library, driver)			
	Debug and profile	Heterogeneous layer-by-layer visibility in Development Studio 5 Streamline			
	Evaluation and early prototyping	Ethos-N Static Performance Analyzer (SPA), Arm Juno FPGA systems, Cycle Models			

سیستم های زمان واقعی جاسازی شده

43

Arm's ML processor architecture key features

Efficient convolutions

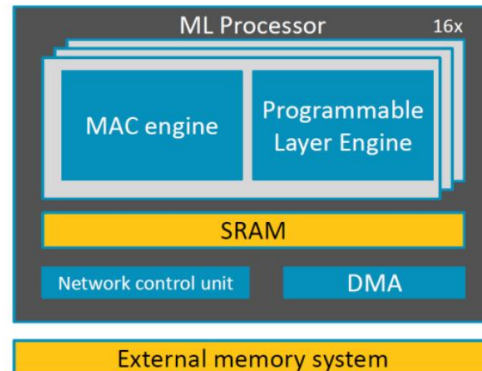
- Convolutions represent the bulk of computation
- We provide dedicated 8-bit hardware for convolutions

Efficient data movement

- More energy is spent moving data than computing
- We amortize activation accesses and compress weights

Sufficient programmability

- New operators are invented and topology is changed frequently
- We provide programmability to future-proofed as new network architectures appear

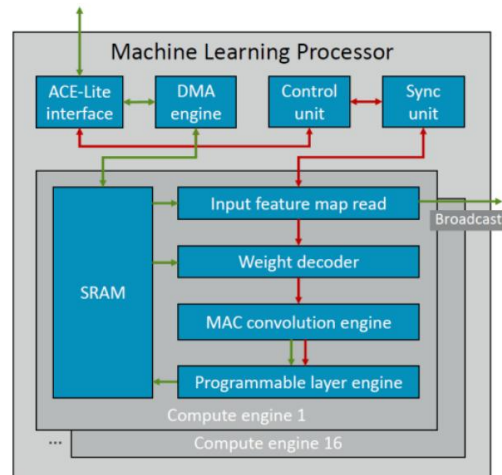


سیستم های زمان واقعی جاسازی شده

44

Arm's ML processor

- A microcontroller and DMA engine manage overall network scheduling
- The compute engine processes major sections of the neural network
 - Stores weights
 - Stores and manipulates activation data
 - Handles convolution in 128-wide MAC units
 - Handles other layer operators via PLE
 - Pipelines data to and from SRAM
- Internal broadcast network manages SRAM population and synchronization



سیستم های زمان واقعی جاسازی شده

45

منطق قابل تنظیم مجدد

- نمونه سازی سریع • برنامه های کاربردی کم حجم • سیستم های بلادرنگ
- سطح بالای پردازش موازی • آرایه های دروازه قابل برنامه ریزی میدانی (FPGA) رایج ترین هستند (Xilinx، Intel، Lattice و غیره)

-منطق قابل تنظیم -حافظه IO -

-هسته های سخت

سیستم های زمان واقعی جاسازی شده

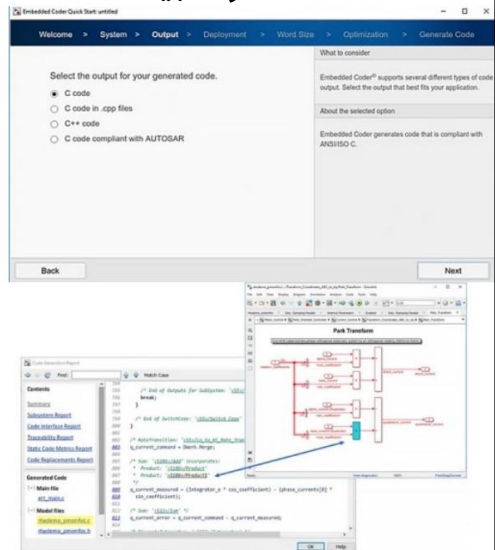
46

کدگذار تعبیه شده متلب

• قابل خواندن ایجاد می کند،
کد فشرده، قابل حمل و سریع C و C++
برای پردازنده های تعبیه شده

• بهینه سازی ها کارایی کد را بهبود می بخشد
و ادغام با کدهای قدیمی، انواع داده ها و
پارامترهای کالیبراسیون را تسهیل می کند.

• از تست نرم افزار در حلقه (SIL) و پردازشگر
در حلقه (PIL) پشتیبانی می کند.



سیستم های زمان واقعی جاسازی شده

47