technische universität
dortmund

fakultät für informatik
informatik 12

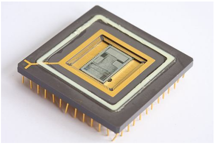
Embedded System Hardware
- Processing -

Peter Marwedel
Informatik 12
TU Dortmund
Germany

2012年 11 月 20 日

Efficiency:
slide from lecture 1 applied to processing

- CPS & ES must be **efficient**
  - ➡ Code-size efficient (especially for systems on a chip)



  - ➡ Run-time efficient

  - Weight efficient

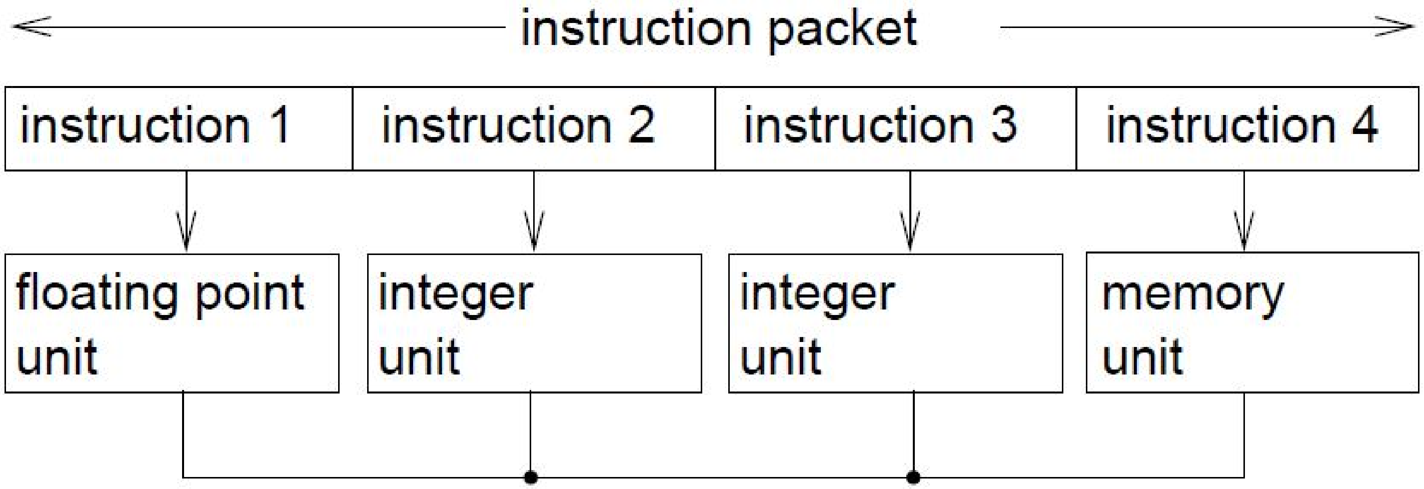
  - Cost efficient

  - ➡ Energy efficient

© Graphics: Microsoft, P. Marwedel, M. Engel, 2011

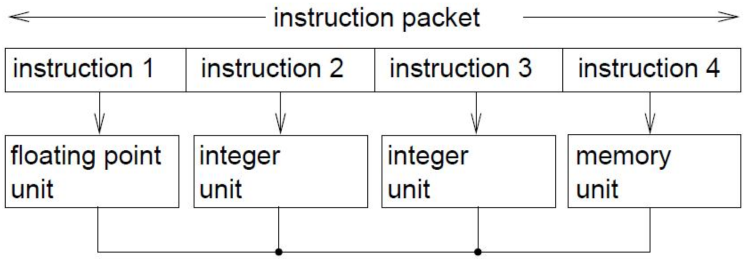
Key idea of very long instruction word (VLIW) computers (1)

- Instructions included in long instruction packets.
- Instruction packets are assumed to be executed in parallel.
- Fixed association of packet bits with functional units.



- Compiler is assumed to generate these “parallel” packets

Key idea of very long instruction word (VLIW) computers (2)



- Complexity of finding parallelism is moved from the hardware (RISC/CISC processors) to the compiler;
- Ideally, this avoids the overhead (silicon, energy, ..) of identifying parallelism at run-time.

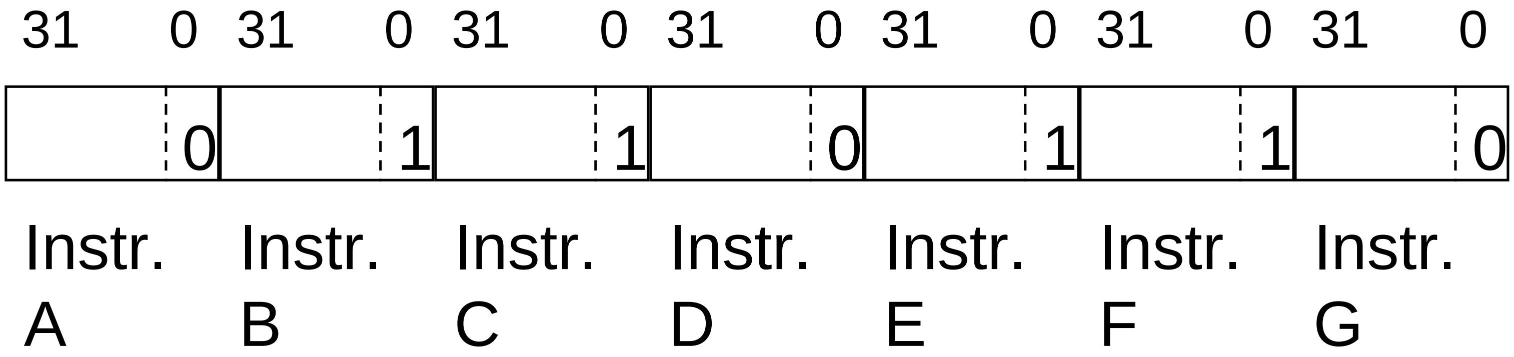
☞A lot of expectations into VLIW machines

- However, possibly low code efficiency, due to many NOPs

☞Explicitly parallel instruction set computers (EPICs) are an extension of VLIW architectures: parallelism detected by compiler, but no need to encode parallelism in 1 word.

EPIC: TMS 320C6xxx as an example

1 Bit per instruction encodes end of parallel exec.



31 0 31 0 31 0 31 0 31 0 31 0 31 0

0 1 1 0 1 1 0

Instr. A Instr. B Instr. C Instr. D Instr. E Instr. F Instr. G

Cycle	Instruction		
1	A		
2	B	C	D
3	E	F	G

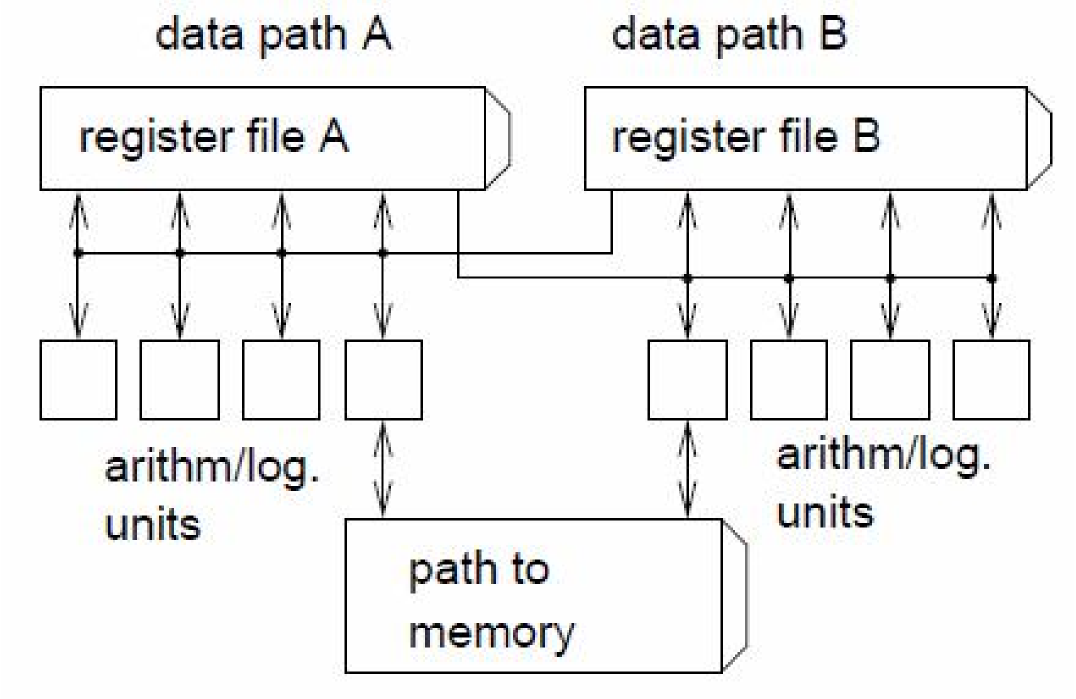
Instructions B, C and D use disjoint functional units, cross paths and other data path resources. The same is also true for E, F and G.

Partitioned register files

- Many memory ports are required to supply enough operands per cycle.
- Memories with many ports are expensive.

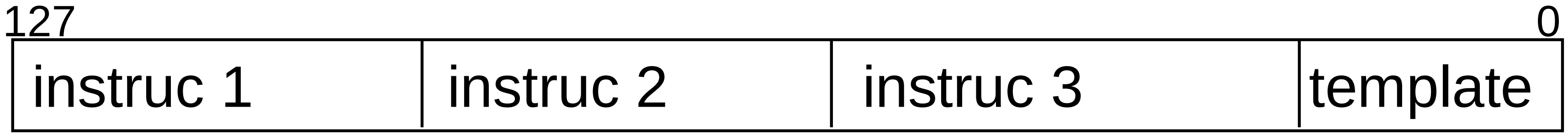
☞ Registers are partitioned into (typically 2) sets, e.g. for TI C6xxx:

Parallel execution cannot span several packets ☞ IA64



More encoding flexibility with IA-64 Itanium

3 instructions per **bundle:**



Instruction grouping information

There are 5 instruction types:

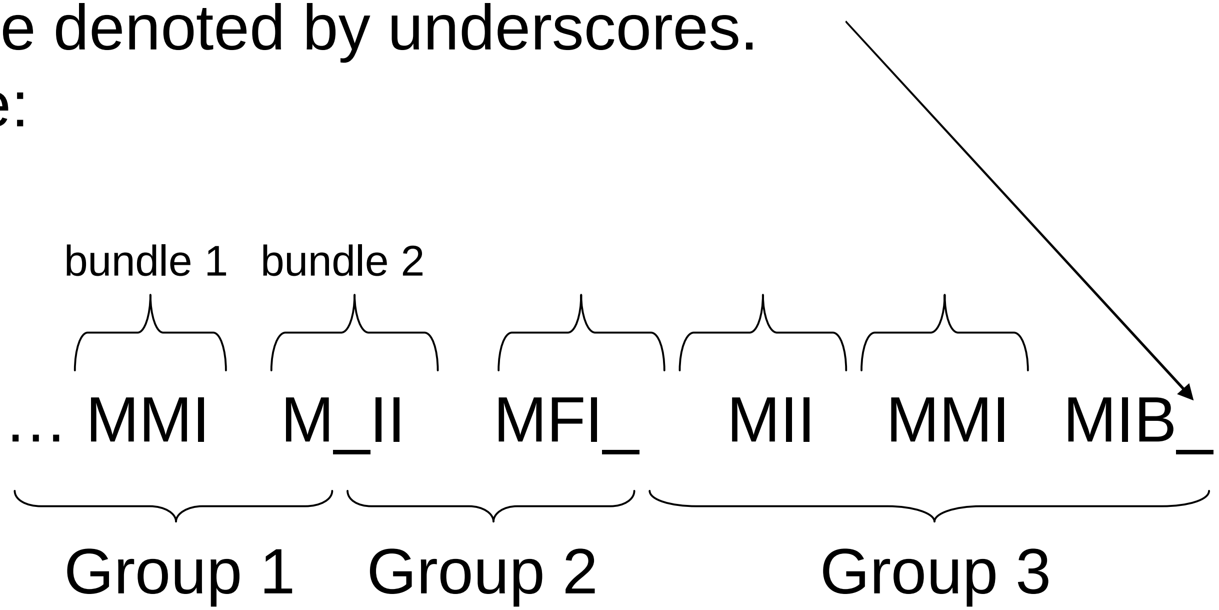
- A: common ALU instructions
- I: more special integer instructions (e.g. shifts)
- M: Memory instructions
- F: floating point instructions
- B: branches

The following combinations can be encoded in templates:

- MII, MMI, MFI, MIB, MMB, MFB, MMF, MBB, BBB, MLX with LX = *move 64-bit immediate* encoded in 2 slots

Templates and instruction types

End of parallel execution called **stops.**
Stops are denoted by underscores.
Example:



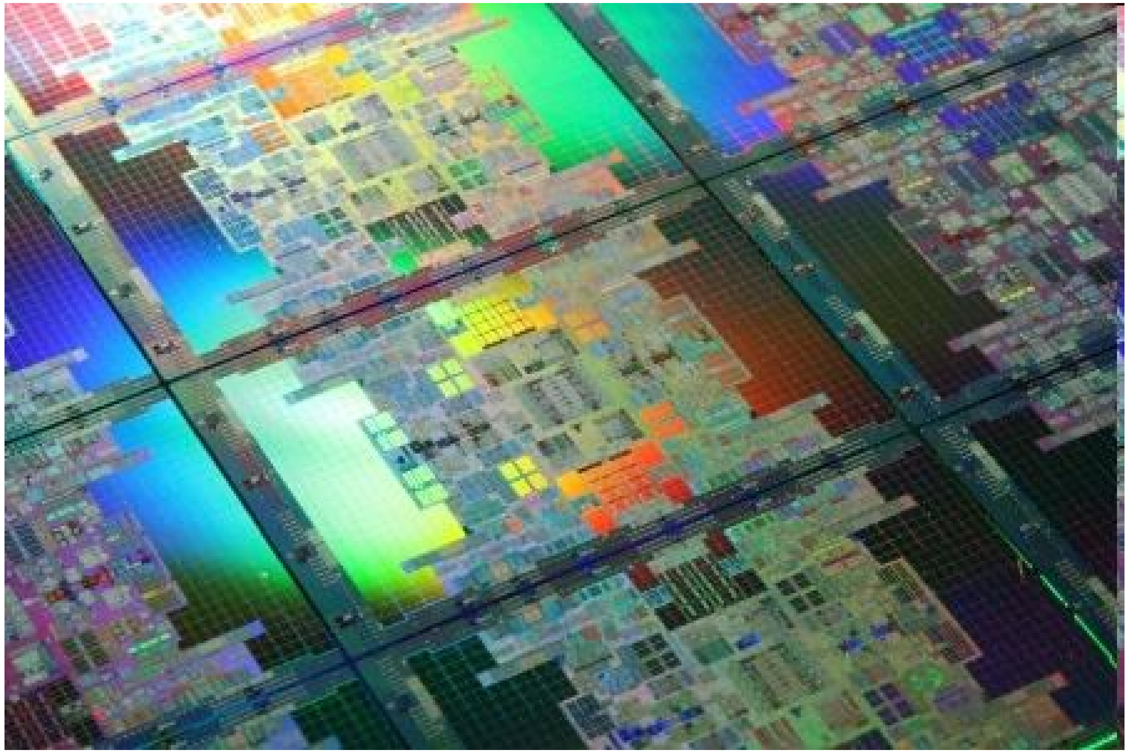
bundle 1 bundle 2

… MMI M_II MFI_ MII MMI MIB_

Group 1 Group 2 Group 3

Very restricted placement of stops within bundle.
Parallel execution within groups possible.
Parallel execution can span several bundles

Itanium® 9300 (Tukwila), 2010



- 2 G transistors
- 4 cores
- 8-fold hyper-threading/core
- 1.5 GHz at 1.3V

[http://www.intel.com/cd/corporate/pressroom/emea/deu/442093.htm]

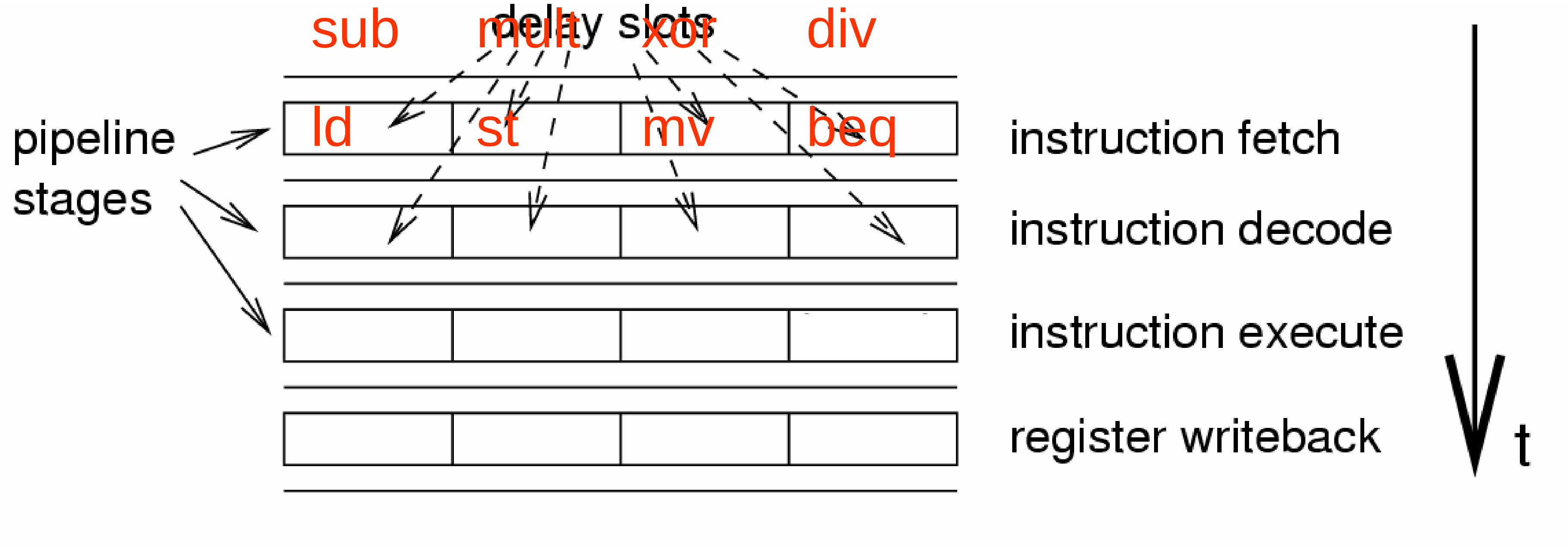
© Intel, 2010

Large # of delay slots, a problem of VLIW processors

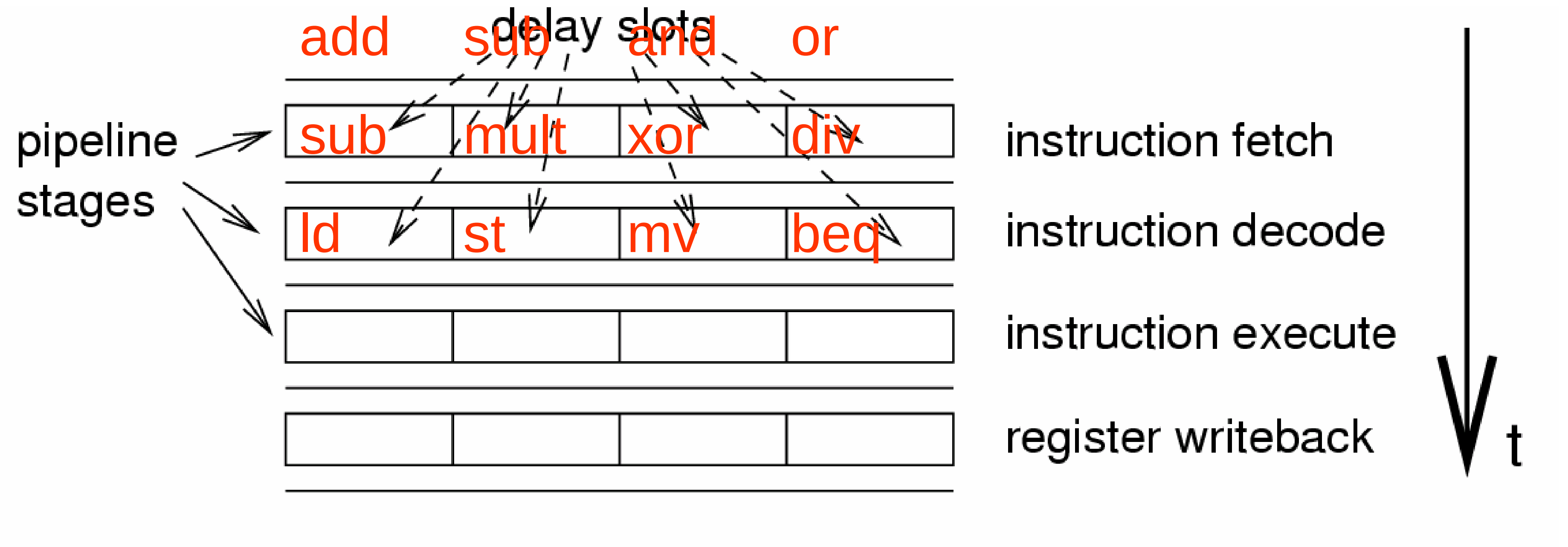
add sub and or

sub mult xor div

ld st mv beq

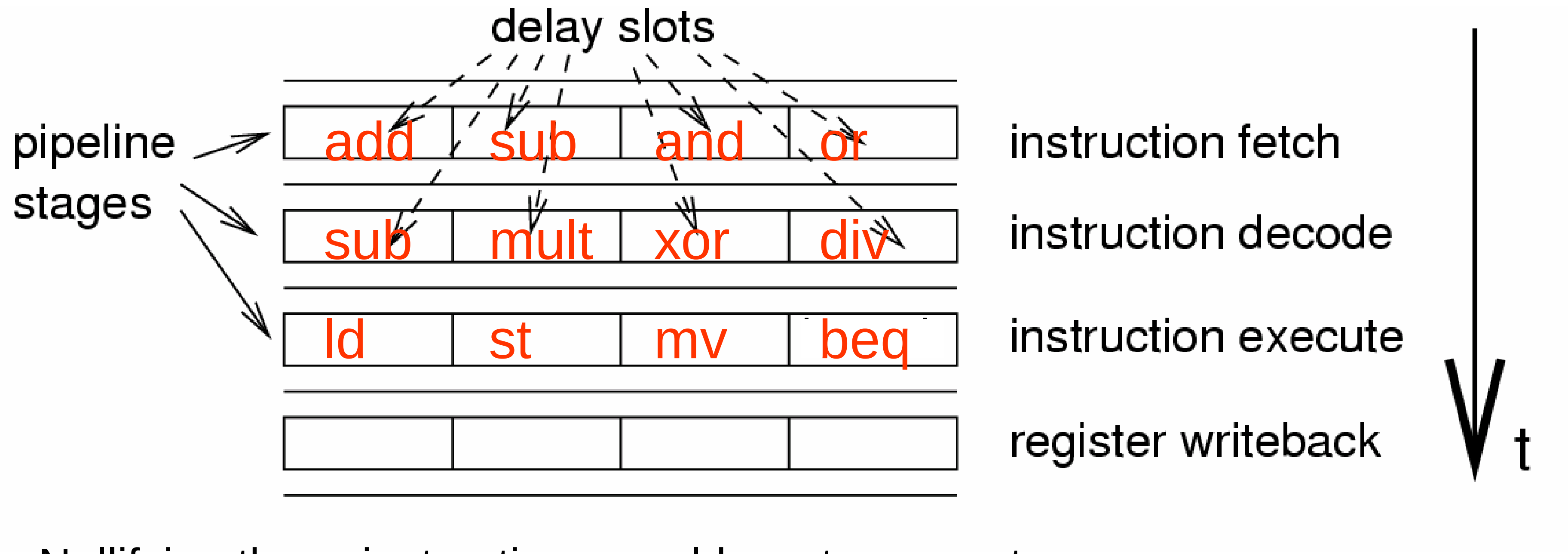


Large # of delay slots,
a problem of VLIW processors



Large # of delay slots, a problem of VLIW processors

The execution of many instructions has been started before it is realized that a branch was required.



Nullifying those instructions would waste compute power

☞ Executing those instructions is declared a feature, not a bug.

☞ How to fill all “delay slots“ with useful instructions?

☞ Avoid branches wherever possible.

Predicated execution: Implementing IF-statements “branch-free“

Conditional Instruction “[c] I“ consists of:

- condition c (some expression involving condition code regs)
- instruction I

c = true ☞ I executed
c = false ☞ NOP

Predicated execution: Implementing IF-statements “branch-free“: TI C6xxx

```
if (c)
{ a = x + y;
  b = x + z;
}
else
{ a = x - y;
  b = x - z;
}
```

Conditional branch

```
        [c] B L1
            NOP 5
            B L2
            NOP 4
            SUB x,y,a
         || SUB x,z,b
L1:         ADD x,y,a
         || ADD x,z,b
L2:
```

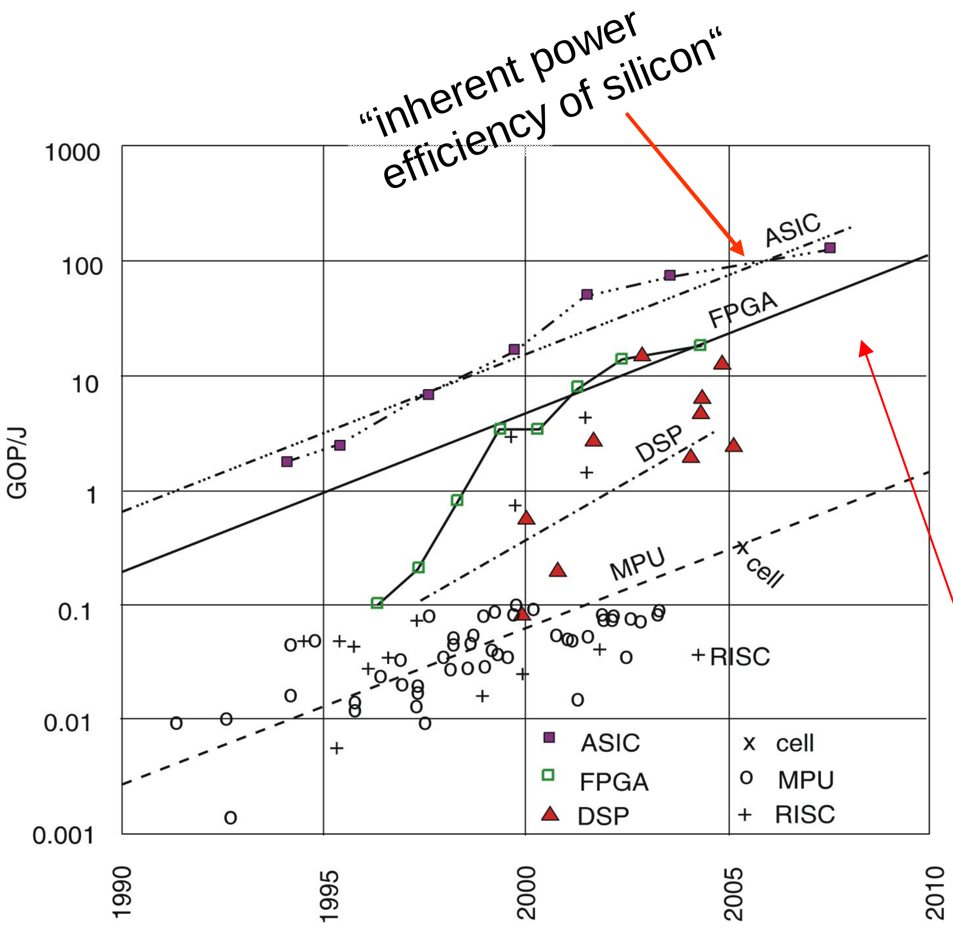
max. 12 cycles

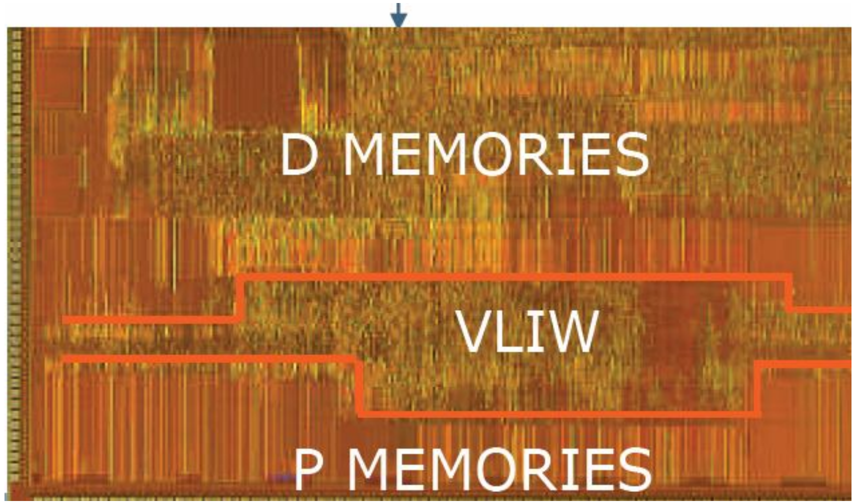
Predicated execution

```
   [c]  ADD x,y,a
|| [c]  ADD x,z,b
|| [!c] SUB  x,y,a
|| [!c] SUB  x,z,b
```

1 cycle

Energy efficiency reached with VLIW processors





© Silicon Hive
41 Issue VLIW for SDR
130 nm, 1,2 V, 6,5mm², 16 bit
30 operations/cycle (OFDM)
150 MHz, 190 mW (incl. SRAMs)
24 GOPs/W, ~1/5 IPE

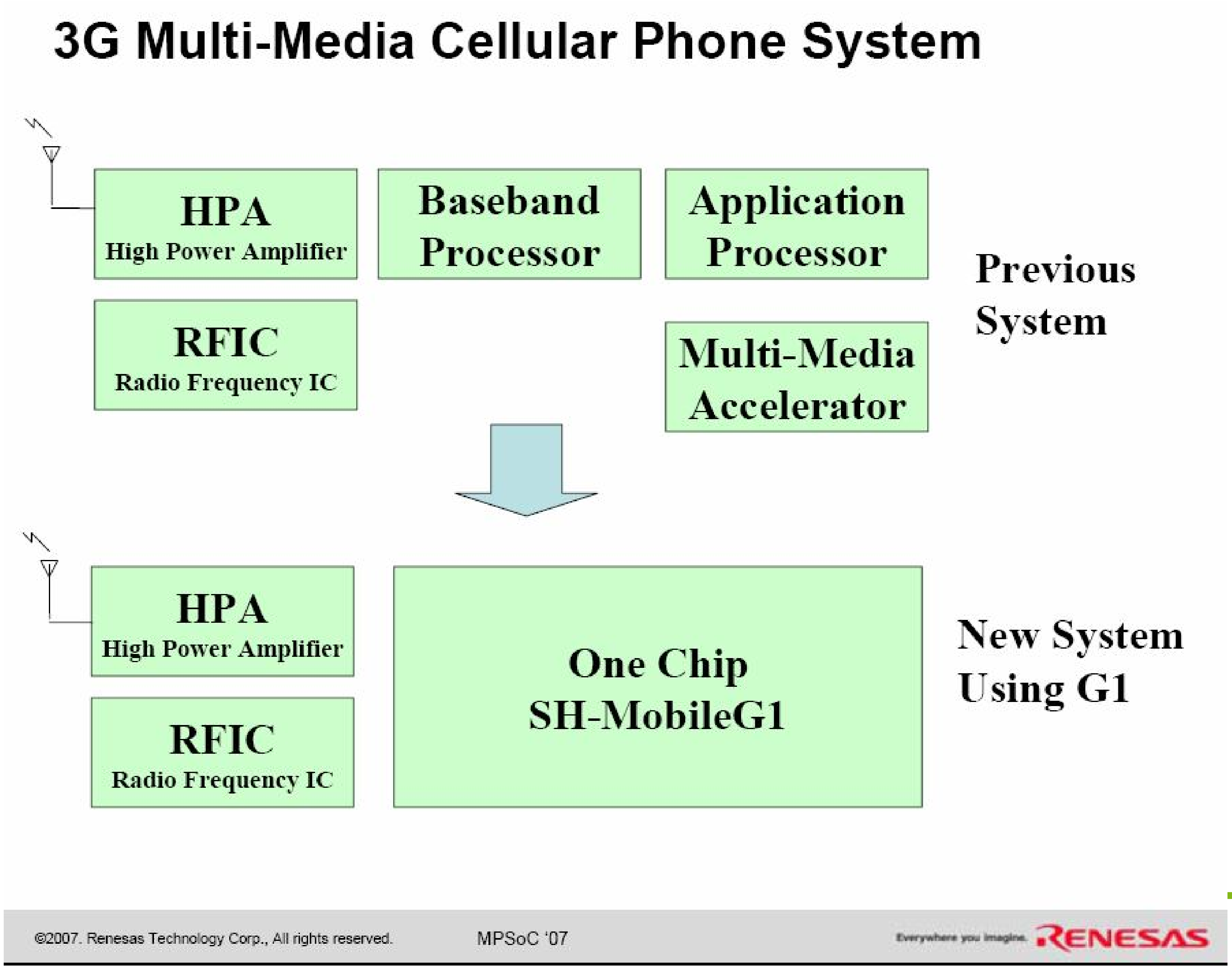
© Hugo De Man: From the Heaven of Software to the Hell of Nanoscale Physics: An Industry in Transition, *Keynote Slides*, ACACES, 2007

Microcontrollers
- MHS 80C51 as an example -

- 8-bit CPU optimised for control applications
- Extensive Boolean processing capabilities
- 64 k Program Memory address space
- 64 k Data Memory address space
- 4 k bytes of on chip Program Memory
- 128 bytes of on chip data RAM
- 32 bi-directional and individually addressable I/O lines
- Two 16-bit timers/counters
- Full duplex UART
- 6 sources/5-vector interrupt structure with 2 priority levels
- On chip clock oscillators
- Very popular CPU with many different variations

Features 4 Embedded Systems

Trend: multiprocessor systems-on-a-chip (MPSoCs)



3G Multi-Media Cellular Phone System

MPSoC '07 Everywhere you imagine. RENESAS

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

technische universität
dortmund

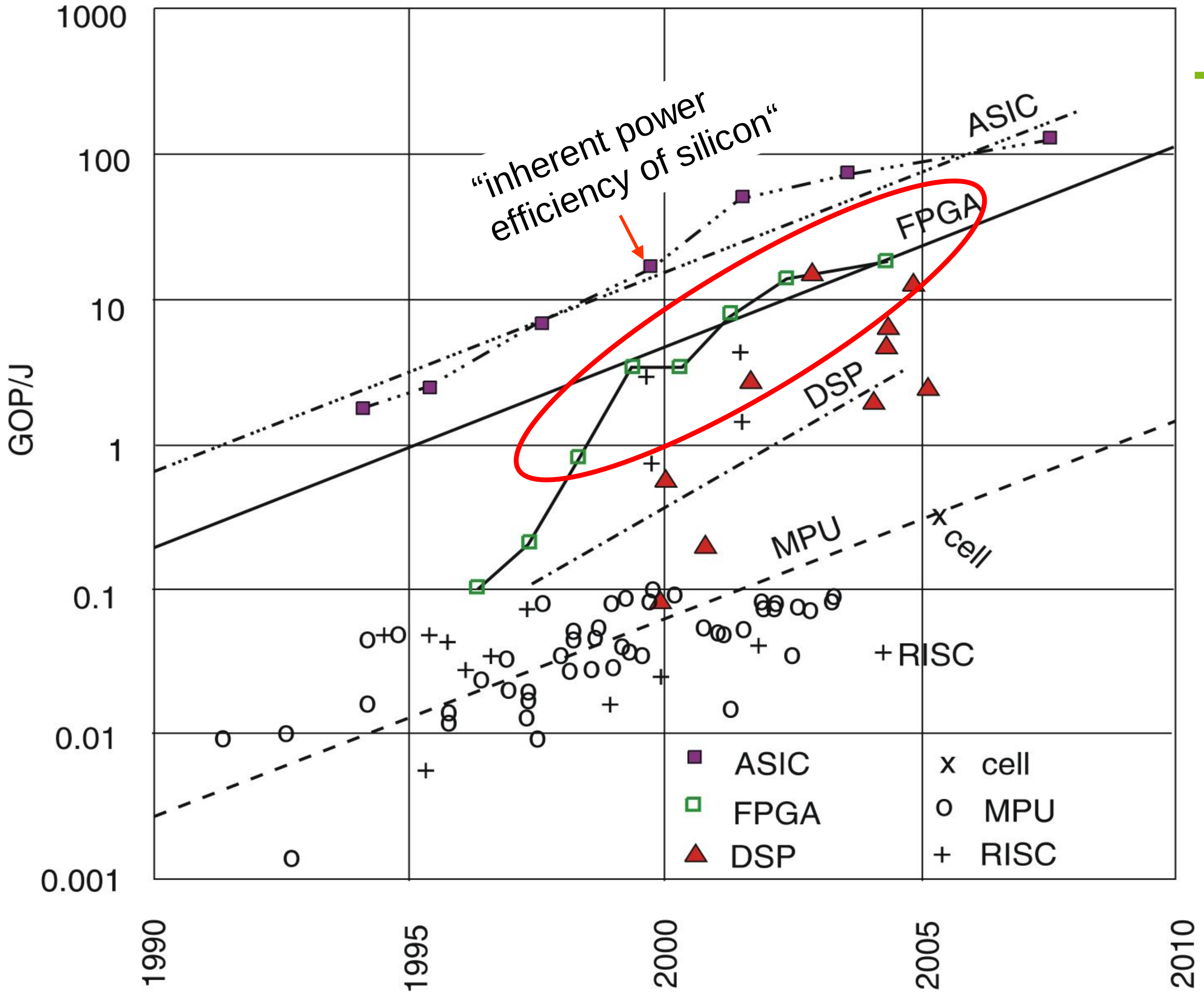
fakultät für informatik
informatik 12

Embedded System Hardware
- Reconfigurable Hardware -

Peter Marwedel
Informatik 12
TU Dortmund
Germany

2012年 11 月 20 日

Energy Efficiency of FPGAs



© Hugo De Man, IMEC, Philips, 2007

Reconfigurable Logic

Custom HW may be too expensive, SW too slow.

Combine the speed of HW with the flexibility of SW

☞HW with programmable functions and interconnect.

☞Use of configurable hardware;
common form: field programmable gate arrays (FPGAs)

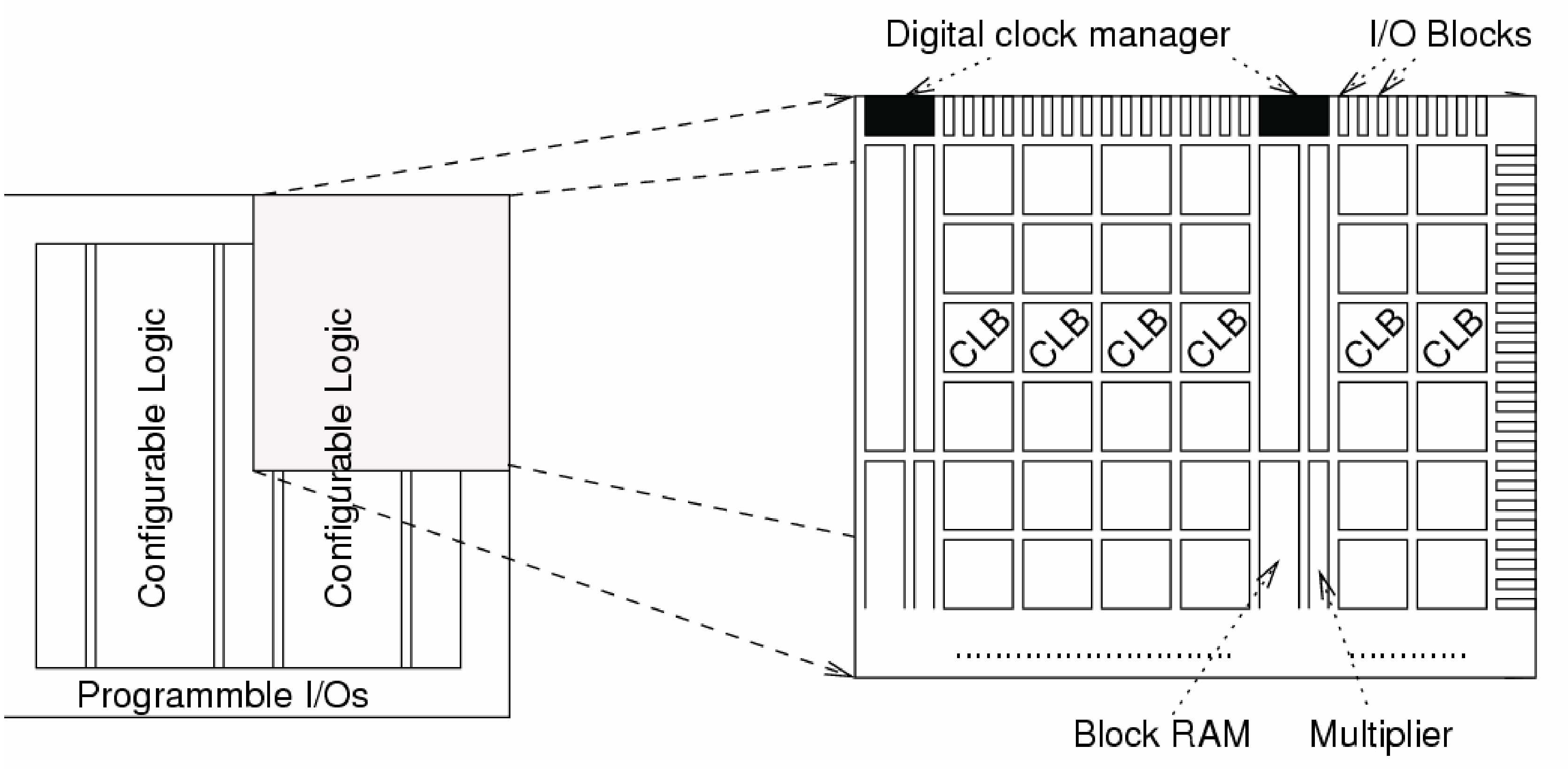
Applications:

- algorithms like de/encryption,
- pattern matching in bioinformatics,
- high speed event filtering (high energy physics),
- high speed special purpose hardware.

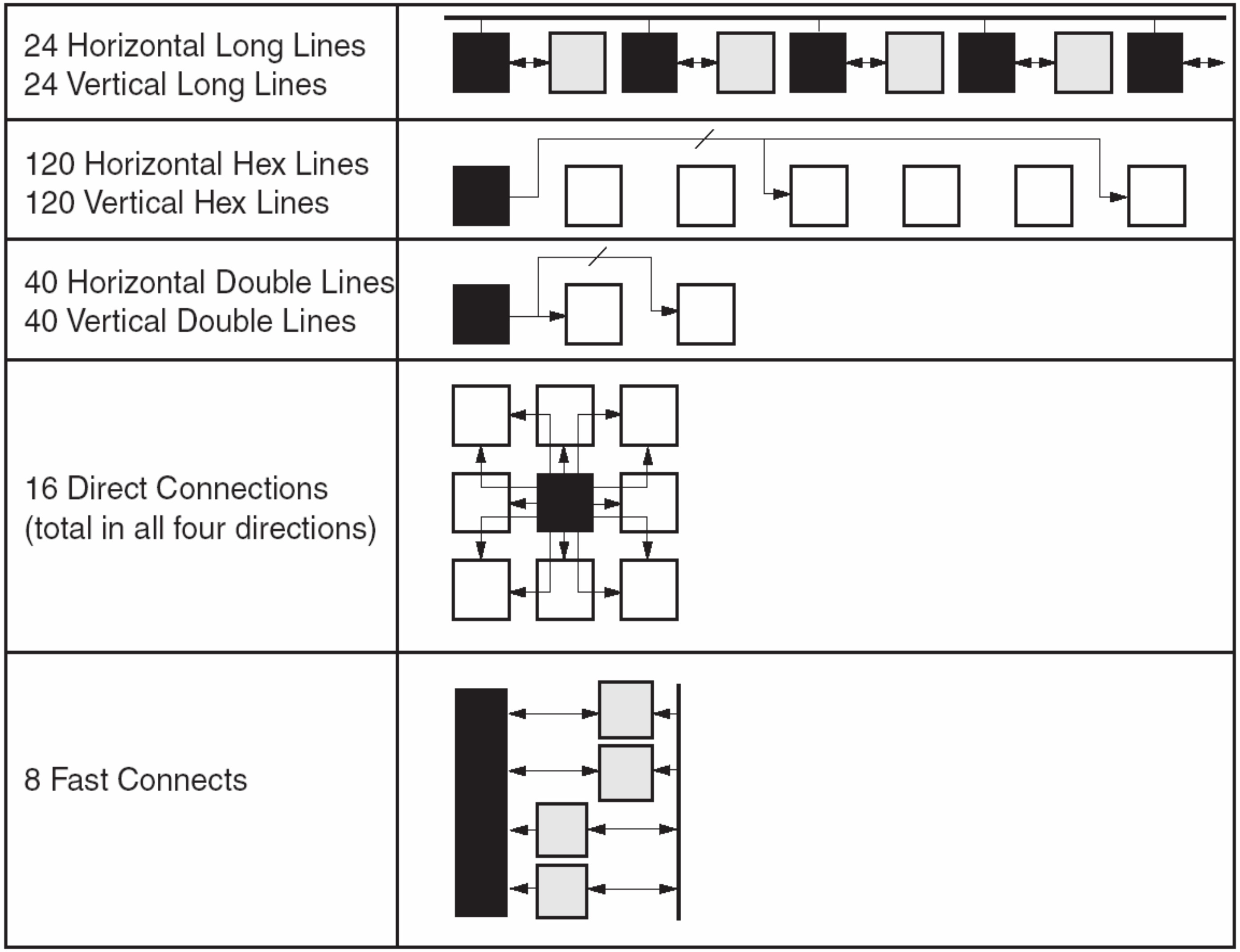
Very popular devices from

- XILINX, Actel, Altera and others

Floor-plan of VIRTEX II FPGAs



Interconnect for Virtex II



24 Horizontal Long Lines
24 Vertical Long Lines

120 Horizontal Hex Lines
120 Vertical Hex Lines

40 Horizontal Double Lines
40 Vertical Double Lines

16 Direct Connections
(total in all four directions)

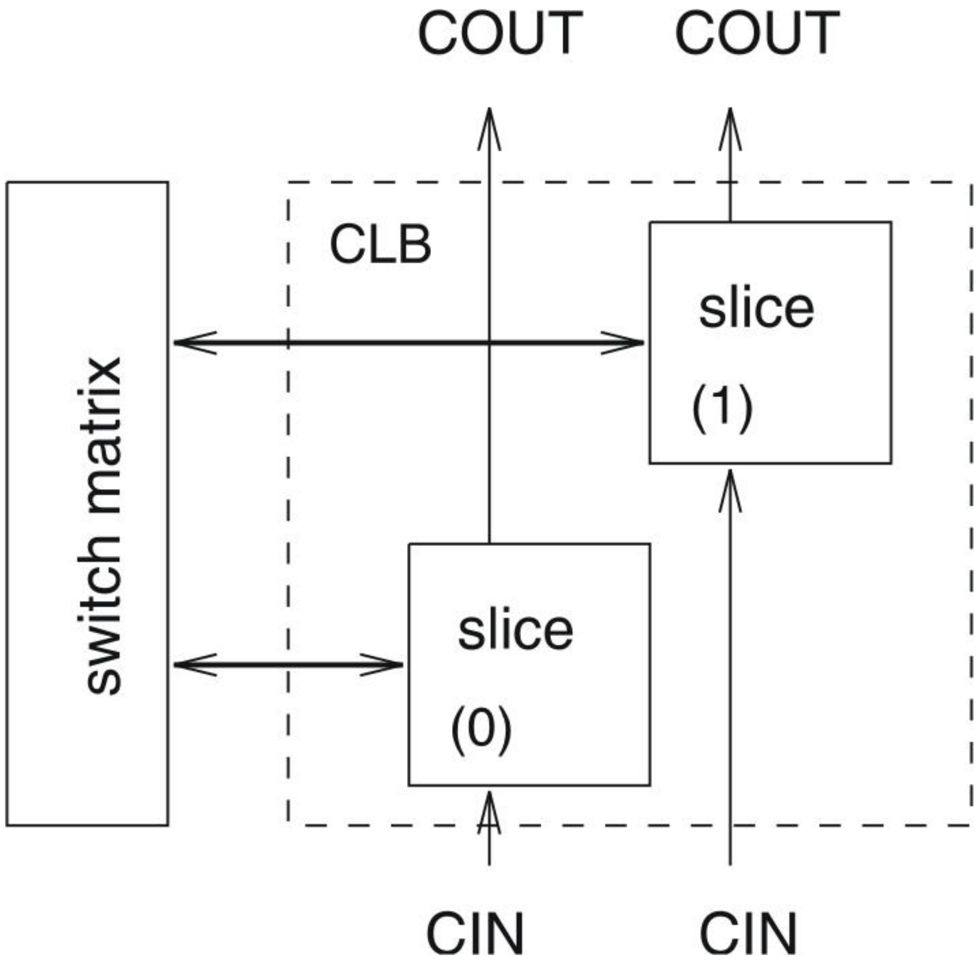
8 Fast Connects

Hierarchical Routing Resources;

More recent: Virtex 5, 6, 7

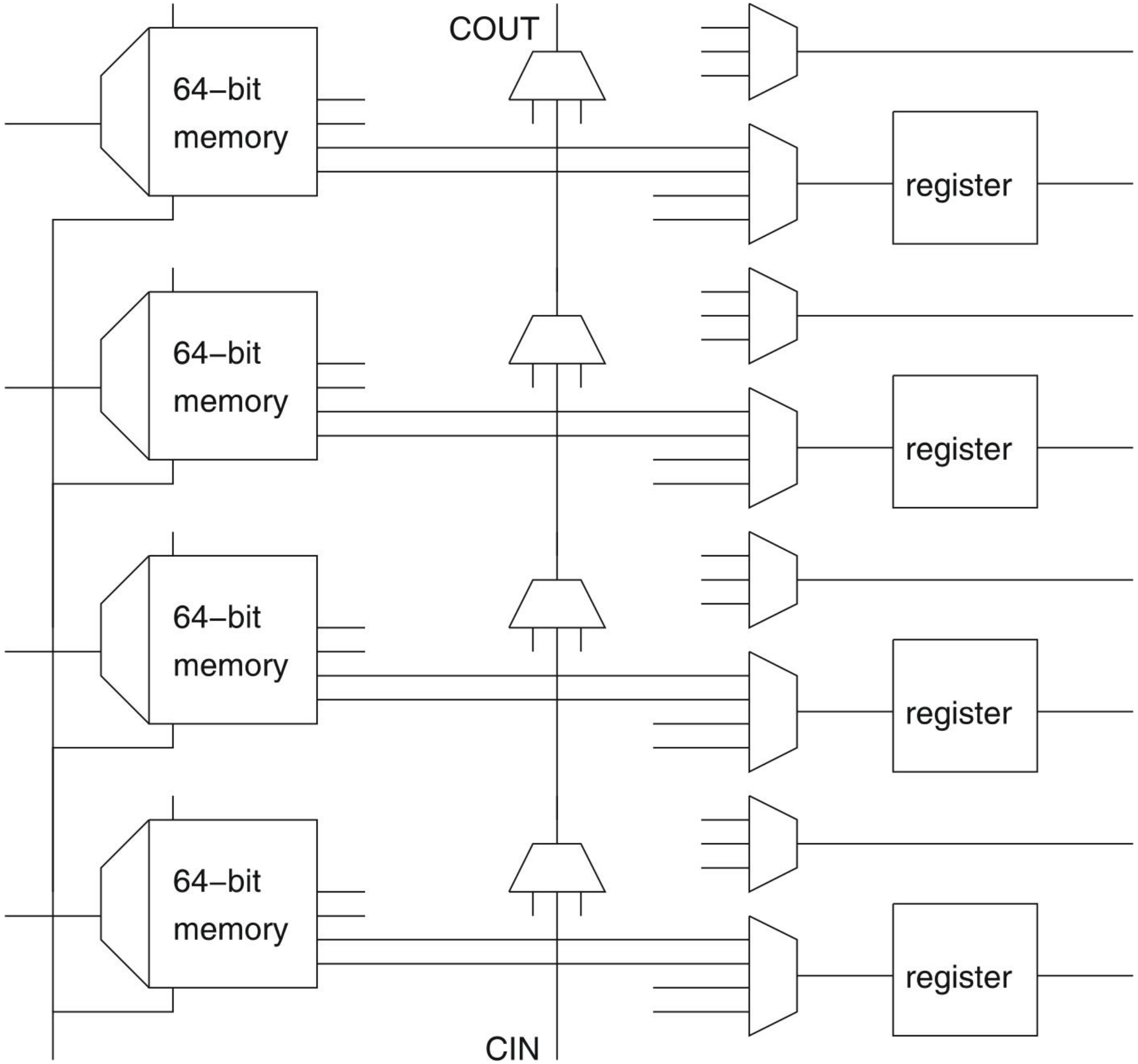
no routing plan found for Virtex 7.

Virtex 7 Configurable Logic Block (CLB)



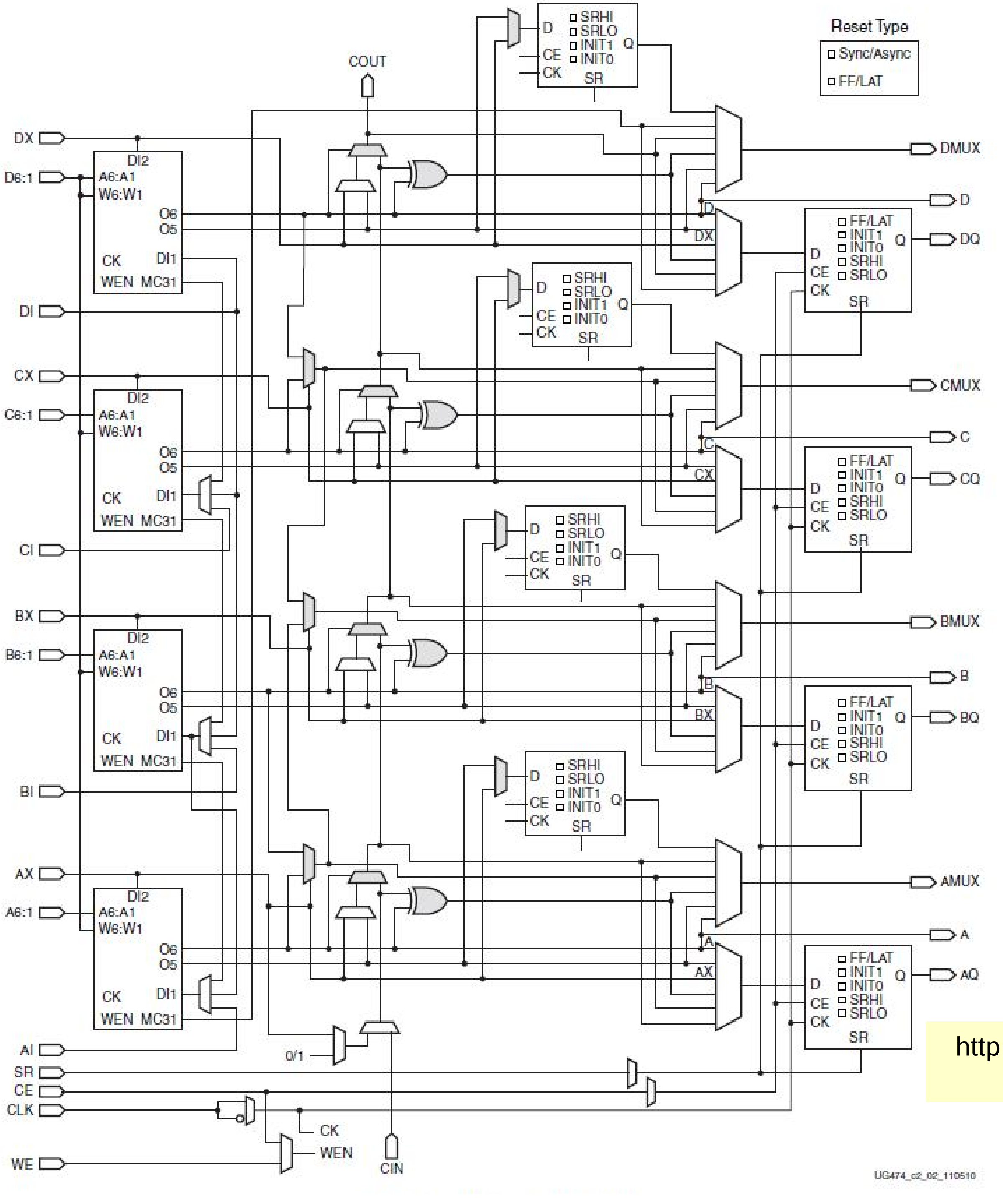
http://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf

Virtex 7 Slice (simplified)



Memories typically used as look-up tables to implement any Boolean function of ≤ 6 variables.

Processors typically implemented as “soft cores” (microblaze)



Virtex 7 SliceM

SliceM supports using memories for storing data and as shift registers

http://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf

Resources available in Virtex 7 devices (max)

Device	XC7V2000T
Logic cells	1,954,560
Slices	305,400
Max distributed RAM [Kb]	21,550
DSP slices	2,160
Block RAM blocks 18 Kb	2,584
Block RAM blocks 36 Kb	1,292
Block RAM blocks Max [Kb]	46,512
PCIe	4
GTX Transceivers	36
A/D converters	1
User I/O	1,200

[http://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf

Virtex-7 FPGAs

Maximum Capability	Virtex-7 T	Virtex-7 XT	Virtex-7 HT
Logic density	1,995 k	1,139 k	864 k
Peak transceiver speed	12.5 Gb/s (GTX)	13.1 Gb/s (GTH)	28.05 Gb/s (GTZ)
Peak bi-directional bandwidth	0.9 Tb/s	2.515 Tb/s	2.784 Tb/s
DSP throughput (symmetric filter)	2,756 G MACS	5,314 GMACS	5,053 GMACS
Block RAM	46.5 Mb	85 Mb	64.4 Mb
I/O pins	1,200	1,100	700

http://www.xilinx.com/products/silicon-devices/fpga/virtex-7/index.htm

technische universität
dortmund

fakultät für informatik
informatik 12

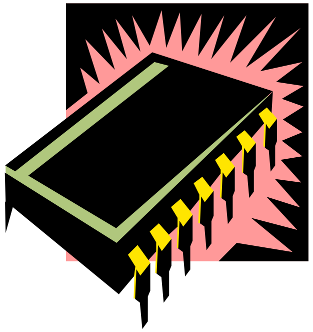
Memory

2012年 11 月 20 日

Memory

Memories?

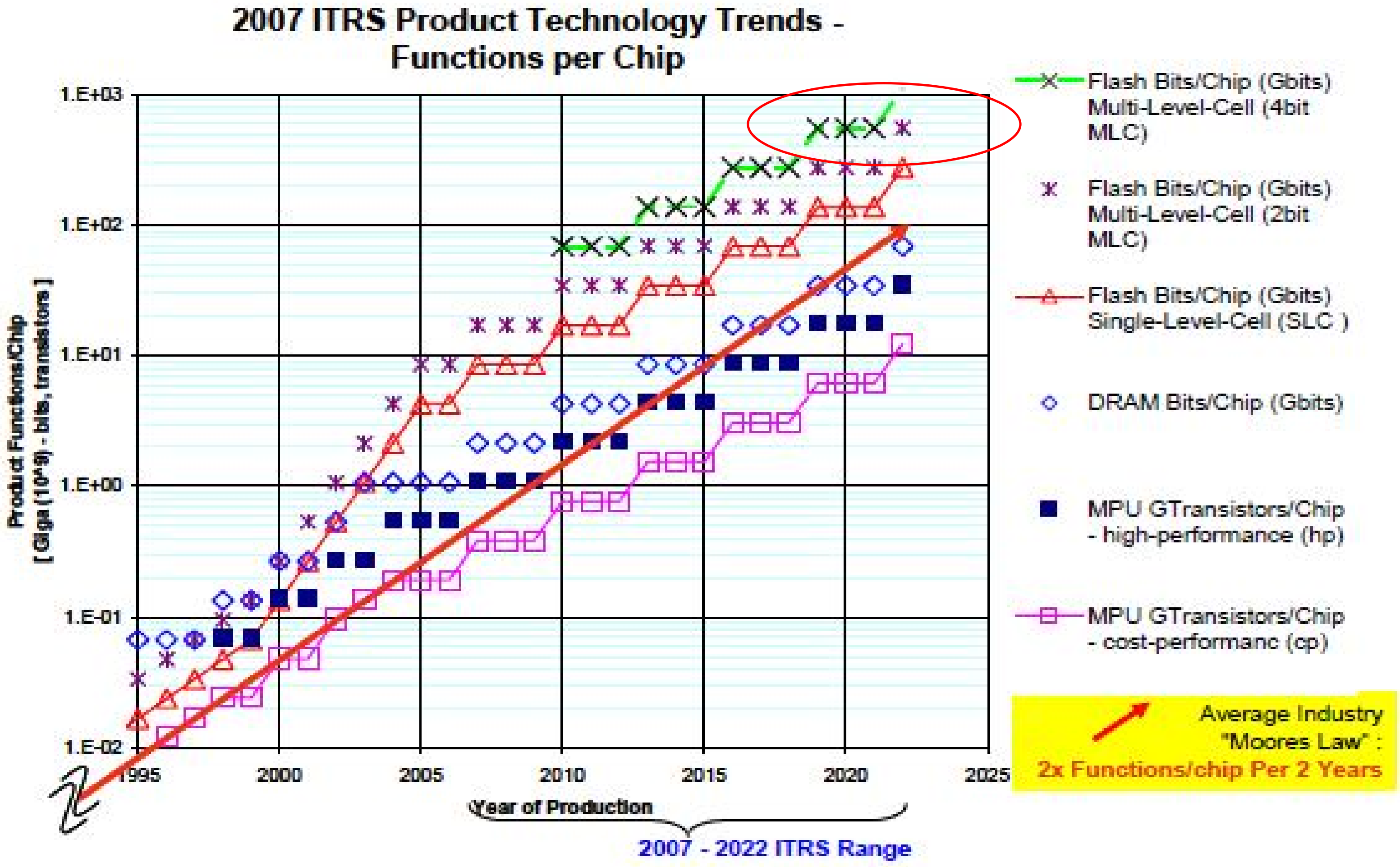
Oops!
Memories!



For the memory, efficiency is again a concern:

- capacity
- energy efficiency
- speed (latency and throughput); predictable timing
- size
- cost
- other attributes (volatile vs. persistent, etc)

Memory capacities expected to keep increasing



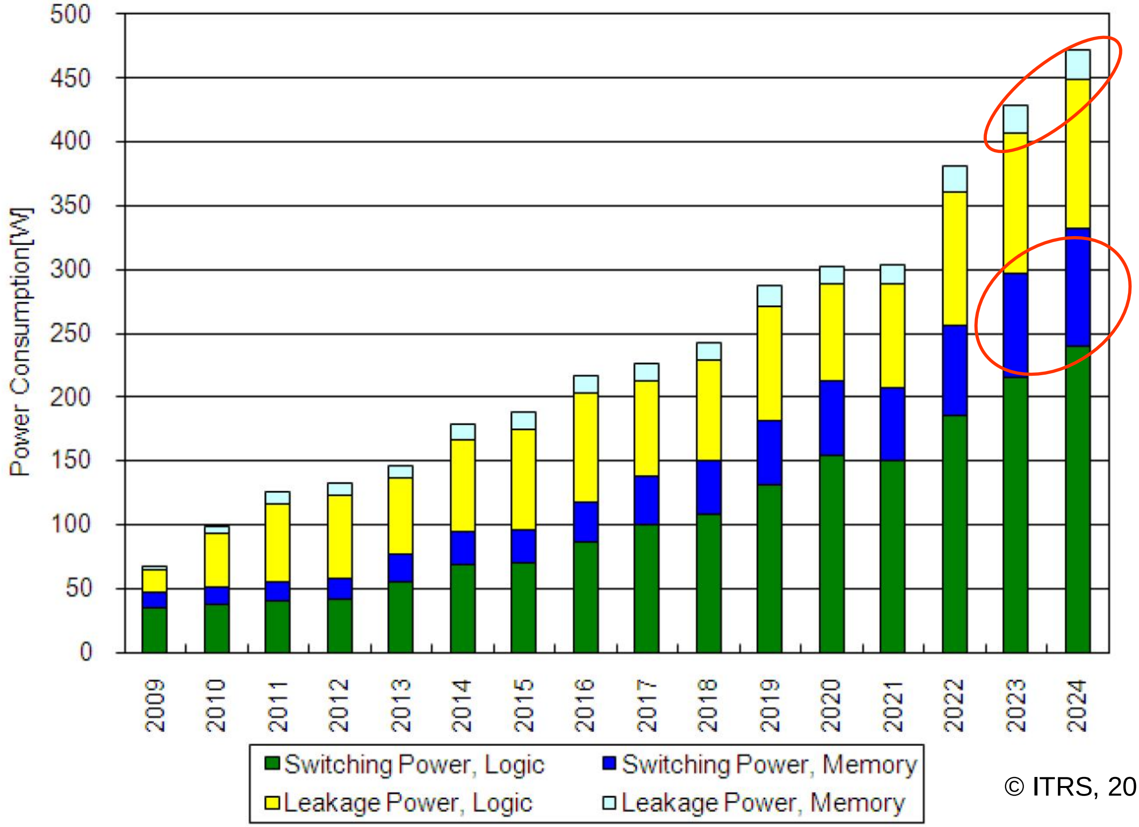
2007 ITRS Product Technology Trends - Functions per Chip

Figure ORTC2 ITRS Product Function Size Trends: MPU Logic Gate Size (4-transistor); Memory Cell Size [SRAM (6-transistor); Flash (SLC and MLC), and DRAM (transistor + capacitor)]--Updated

[ITRS Update 2008]

Where is the power consumed?
- Stationary systems -

- According to *International Technology Roadmap for Semi-conductors* (ITRS), 2010 update, [www.itrs.net]

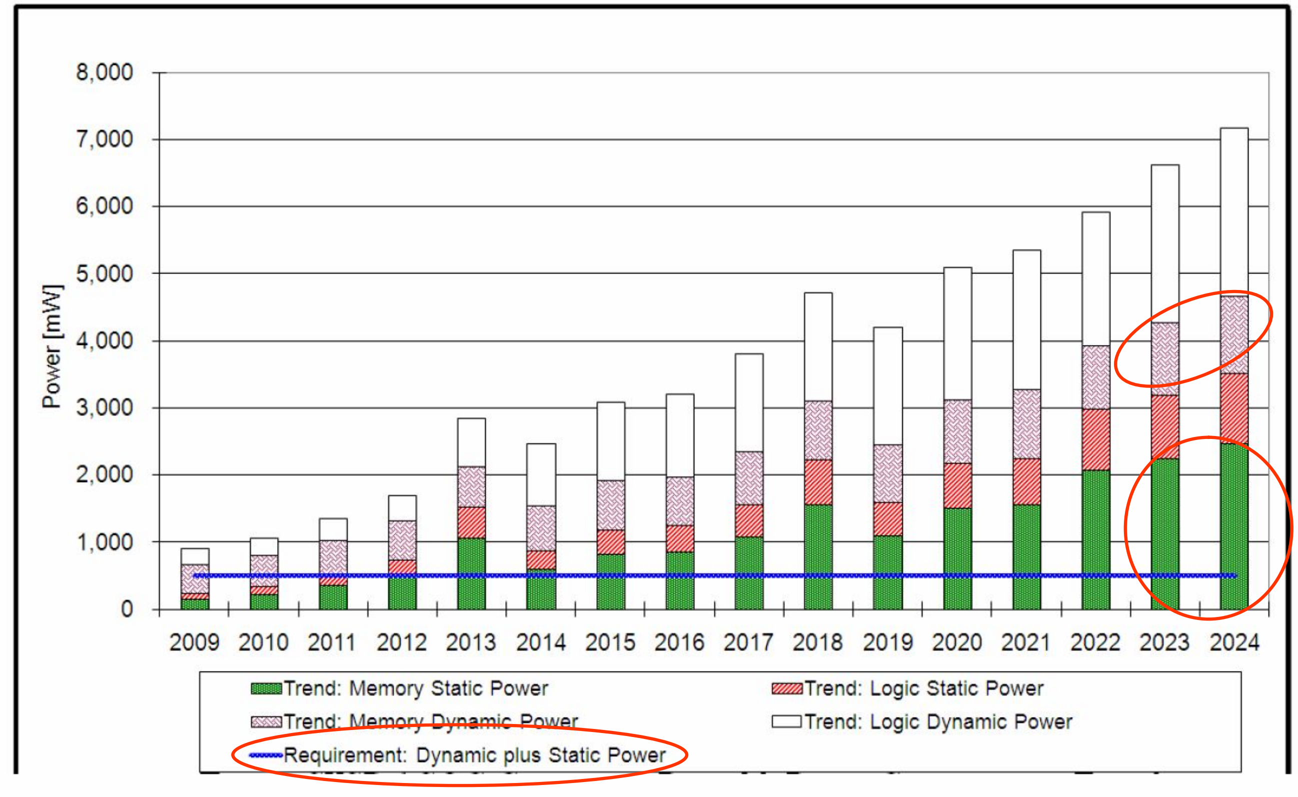


© ITRS, 2010

- Switching power, logic dominating
- Overall power consumption a nightmare for environmentalists

Where is the power consumed?
- Consumer portable systems -

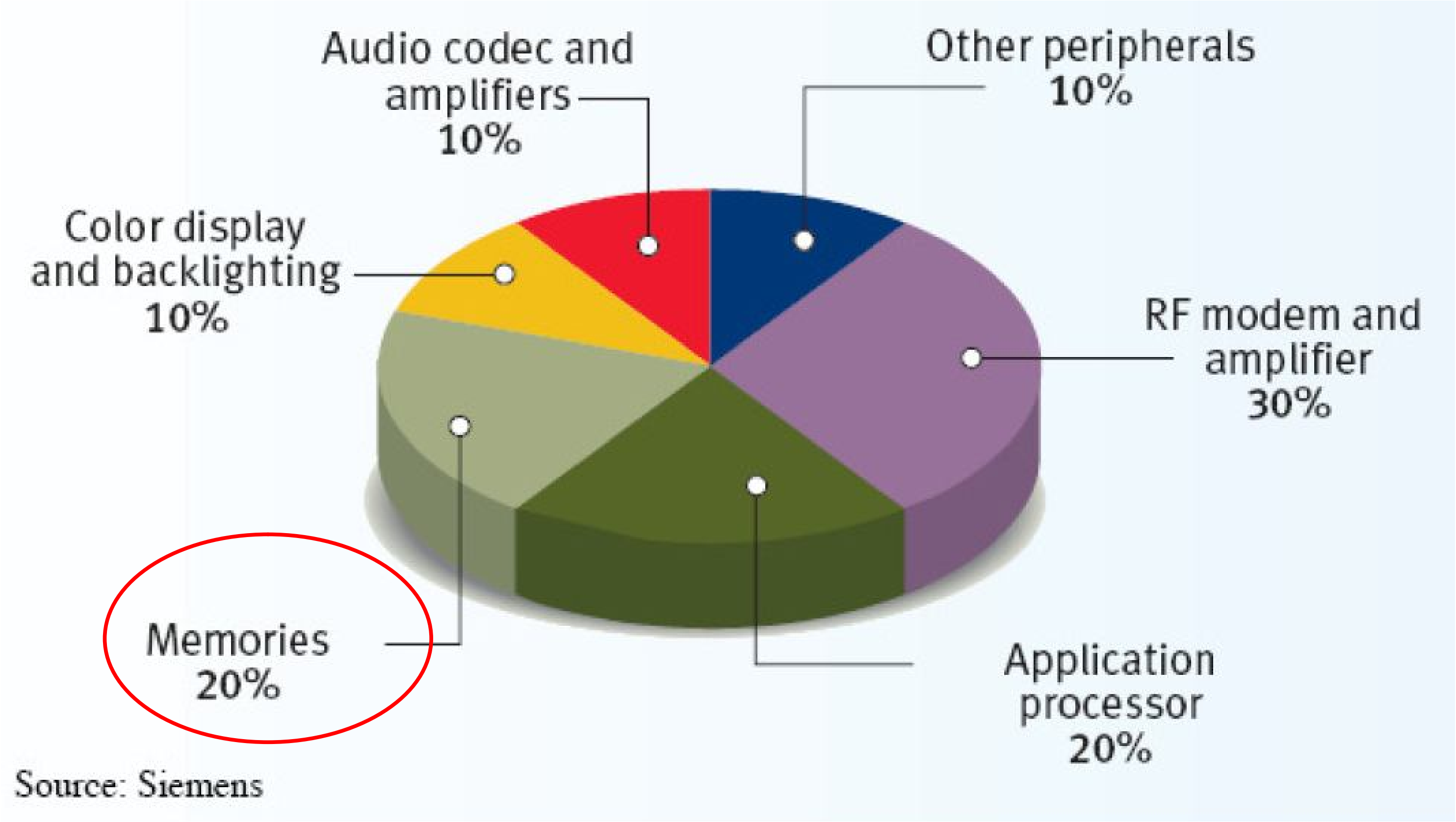
- According to *International Technology Roadmap for Semi-conductors* (ITRS), 2010 update, [www.itrs.net]
- Based on current trends



© ITRS, 2010

- Memory and logic, static and dynamic relevant
- Following current trends will violate maximum power constraint (0.5-1 W).

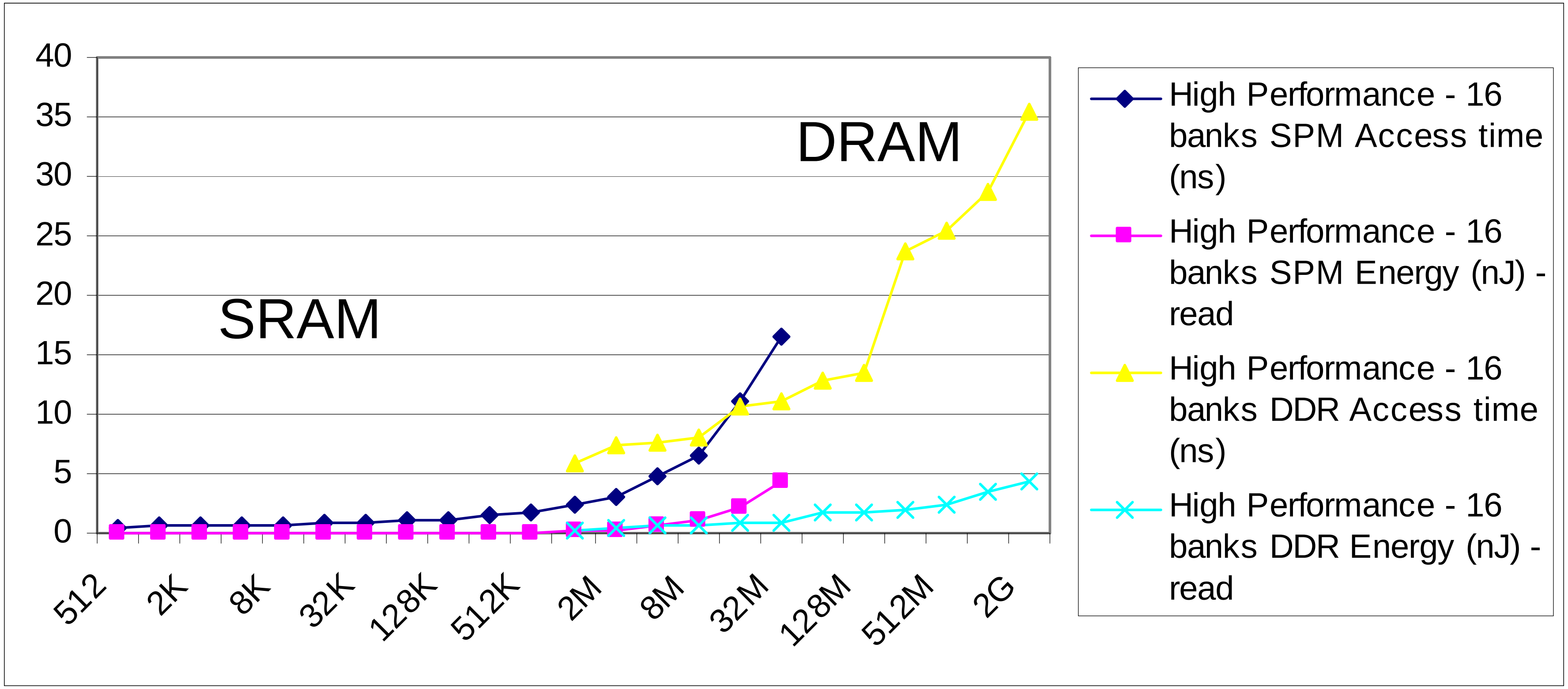
Memory energy significant even if we take display and RF of mobile device into account



[O. Vargas (Infineon Technologies): Minimum power consumption in mobile-phone memory subsystems; Pennwell Portable Design - September 2005;] Thanks to Thorsten Koch (Nokia/ Univ. Dortmund) for providing this source.

Energy consumption and access times of memories

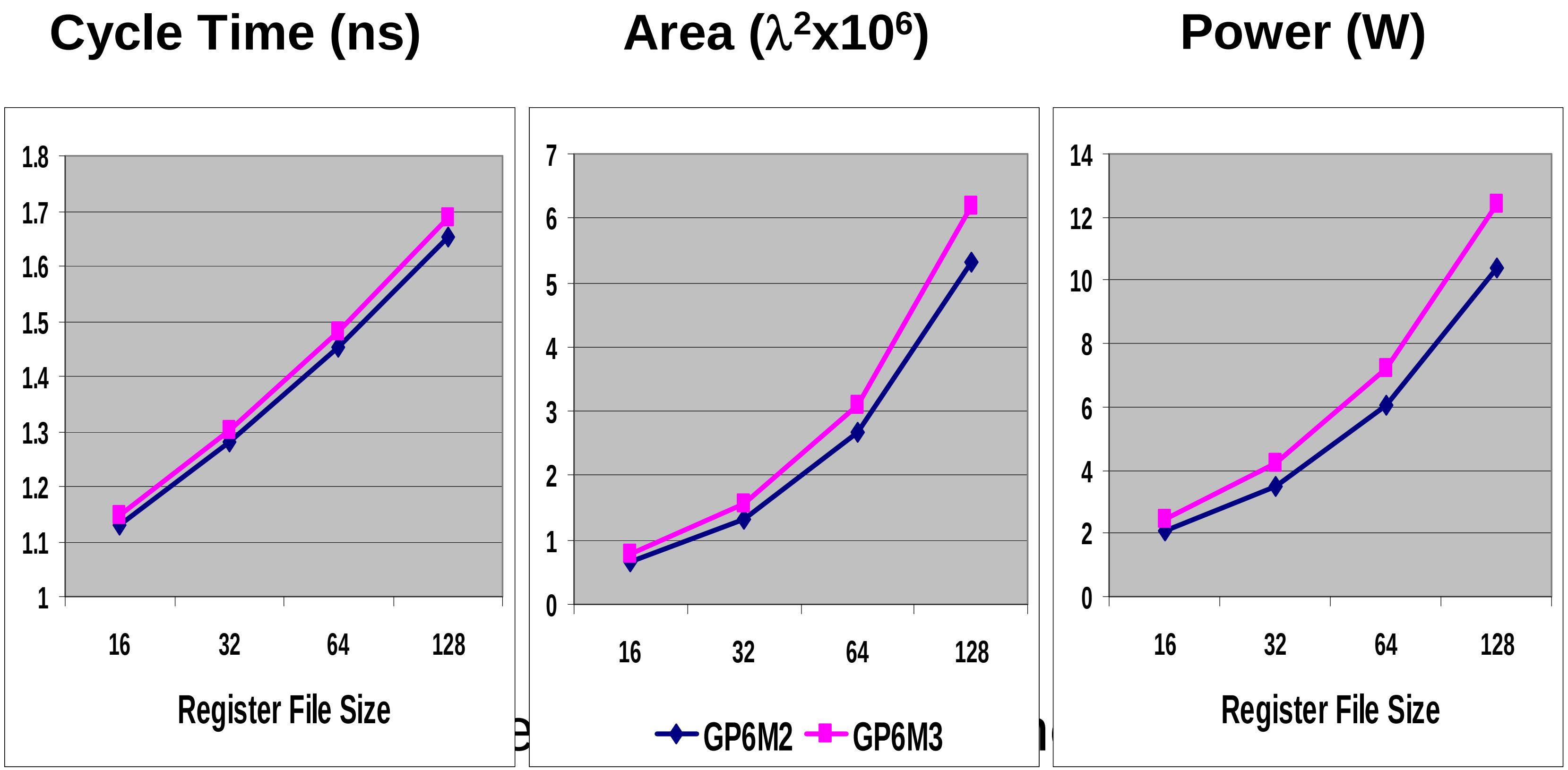
Example CACTI: Scratchpad (SRAM) vs. DRAM (DDR2):



16 bit read; size in bytes;
65 nm for SRAM, 80 nm for DRAM

Source: Olivera Jovanovic, TU Dortmund, 2011

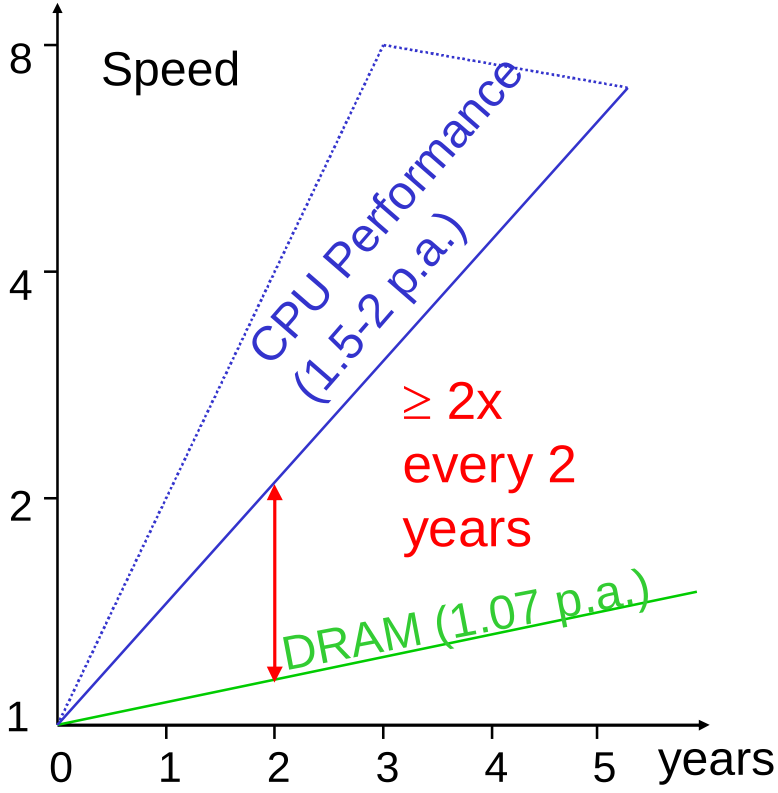
Access times and energy consumption for multi-ported register files



Source and © H. Valero, 2001

Trends for the Speeds

Speed gap between processor and main DRAM increases



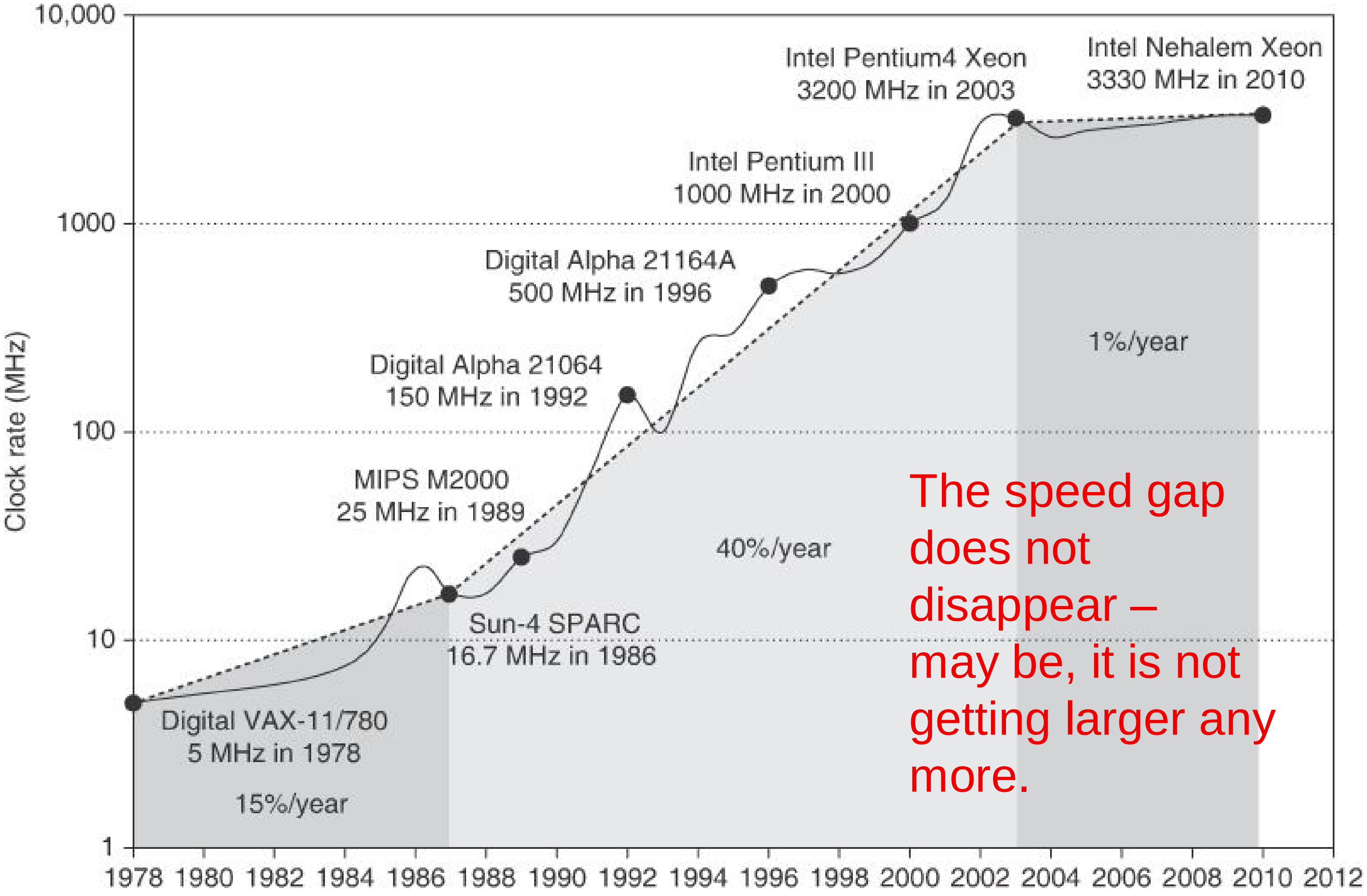
Similar problems also for embedded systems & MPSoCs

☞ Memory access times >> processor cycle times

☞ "Memory wall" problem

[P. Machanik: Approaches to Addressing the Memory Wall, TR Nov. 2002, U. Brisbane]

However,
clock speed increases have come to a halt



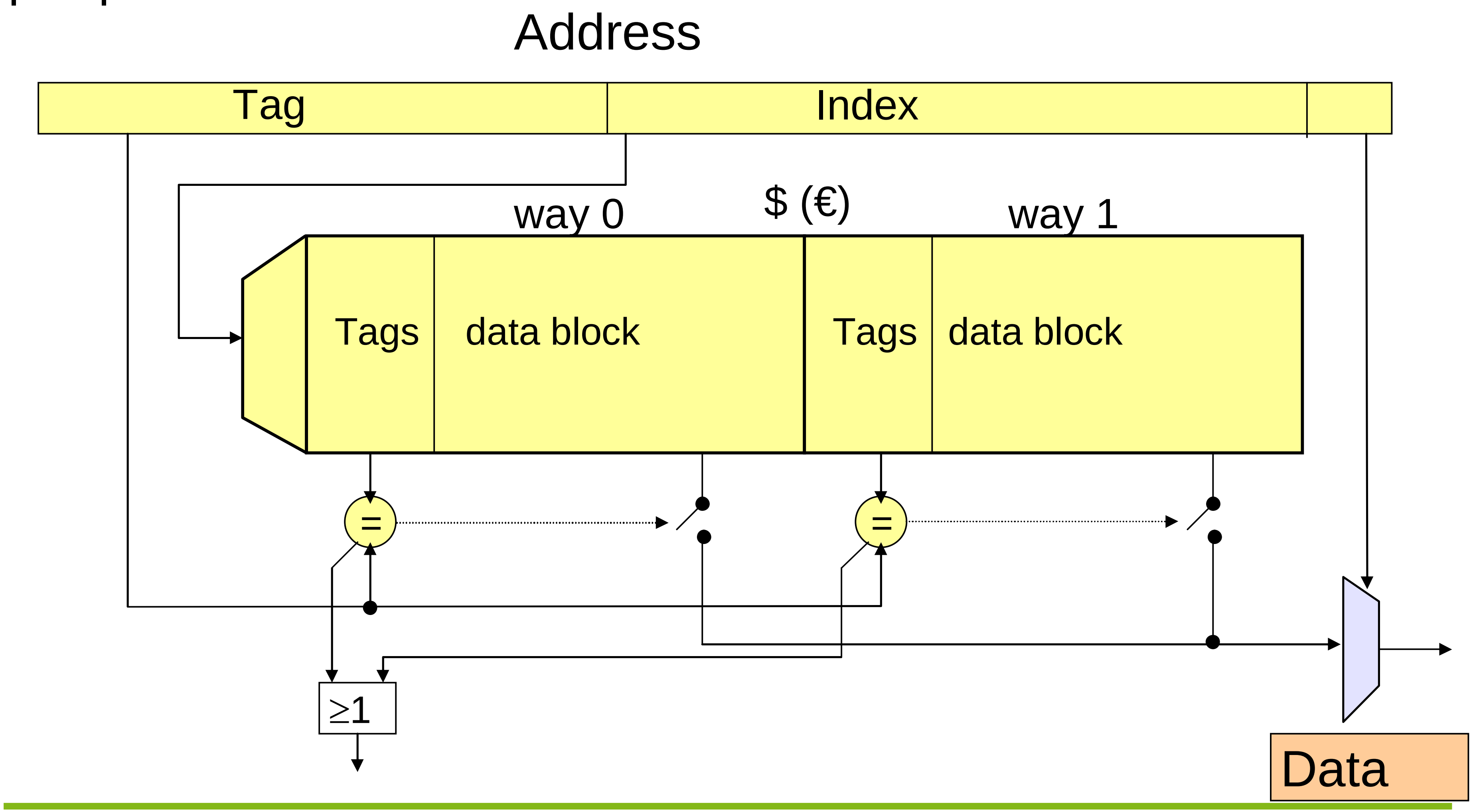
The speed gap does not disappear – may be, it is not getting larger any more.

Copyright © 2011, Elsevier Inc. All rights Reserved.

[Hennessy/Patterson: Computer Architecture, 5th ed., 2011]

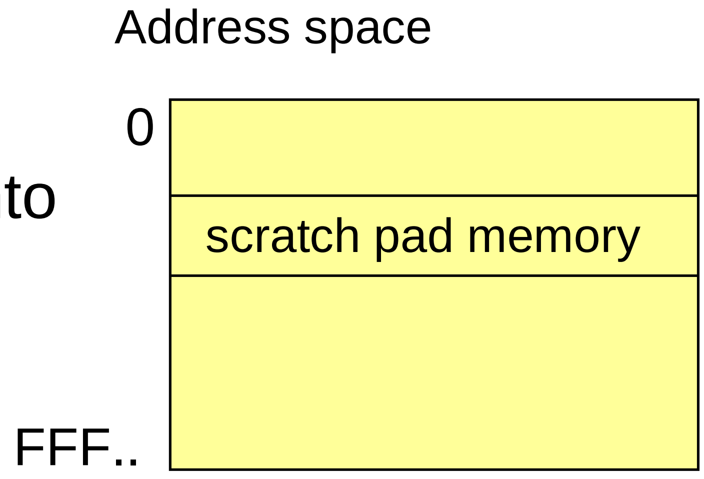
Set-associative cache *n*-way cache

|Set| = 2

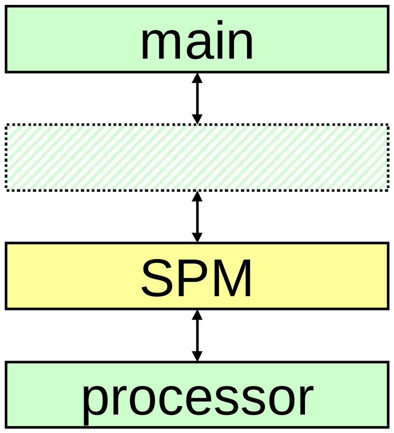


Hierarchical memories using scratch pad memories (SPM)

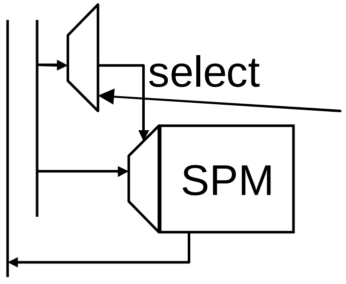
SPM is a small, physically separate memory mapped into the address space



Hierarchy



no tag memory



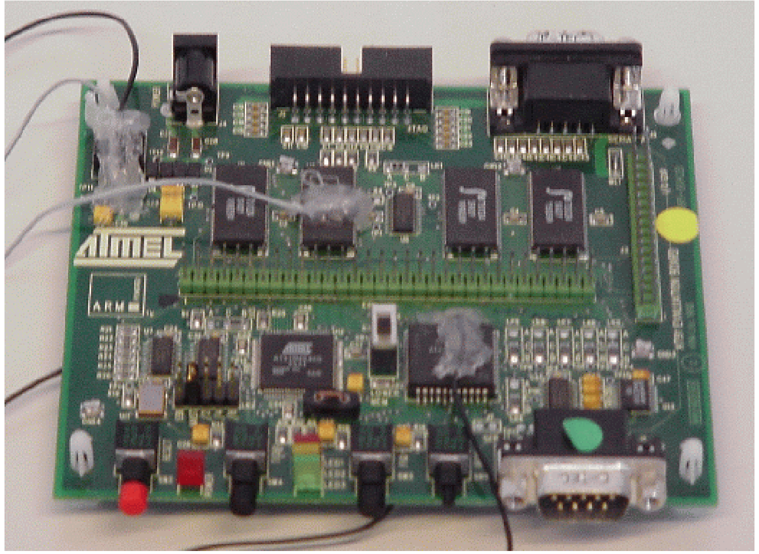
Selection is by an appropriate address decoder (simple!)

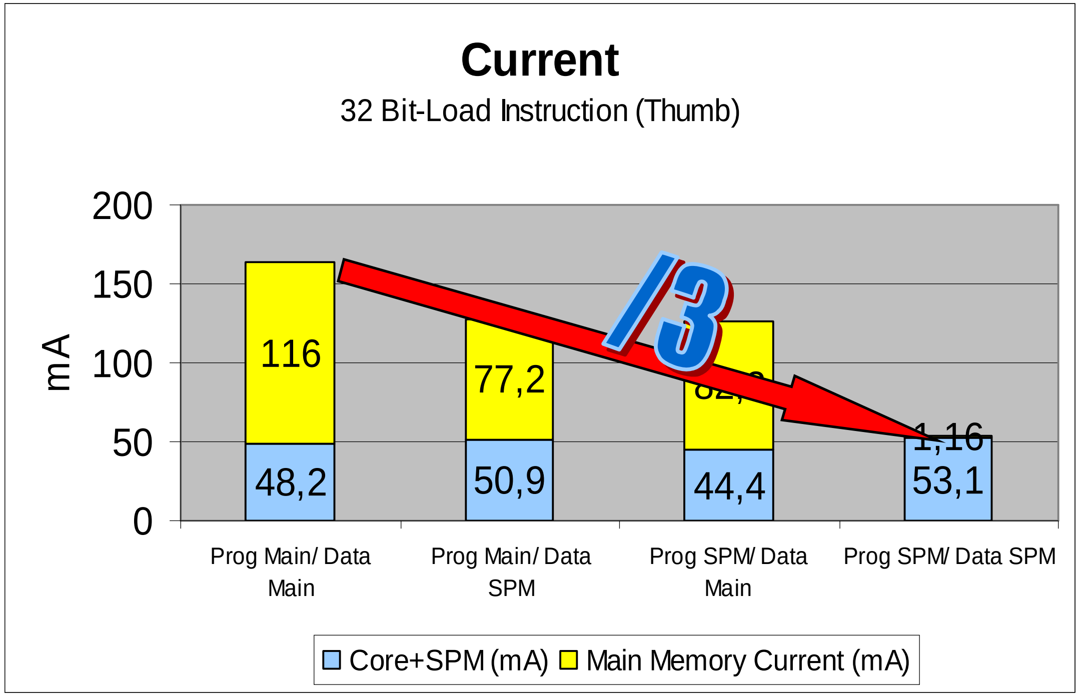
Examples:

- Most ARM cores allow tightly coupled memories
- IBM Cell
- Infineon TriCore
- Many multi-cores, due to high costs of coherent caches

Comparison of currents using measurements

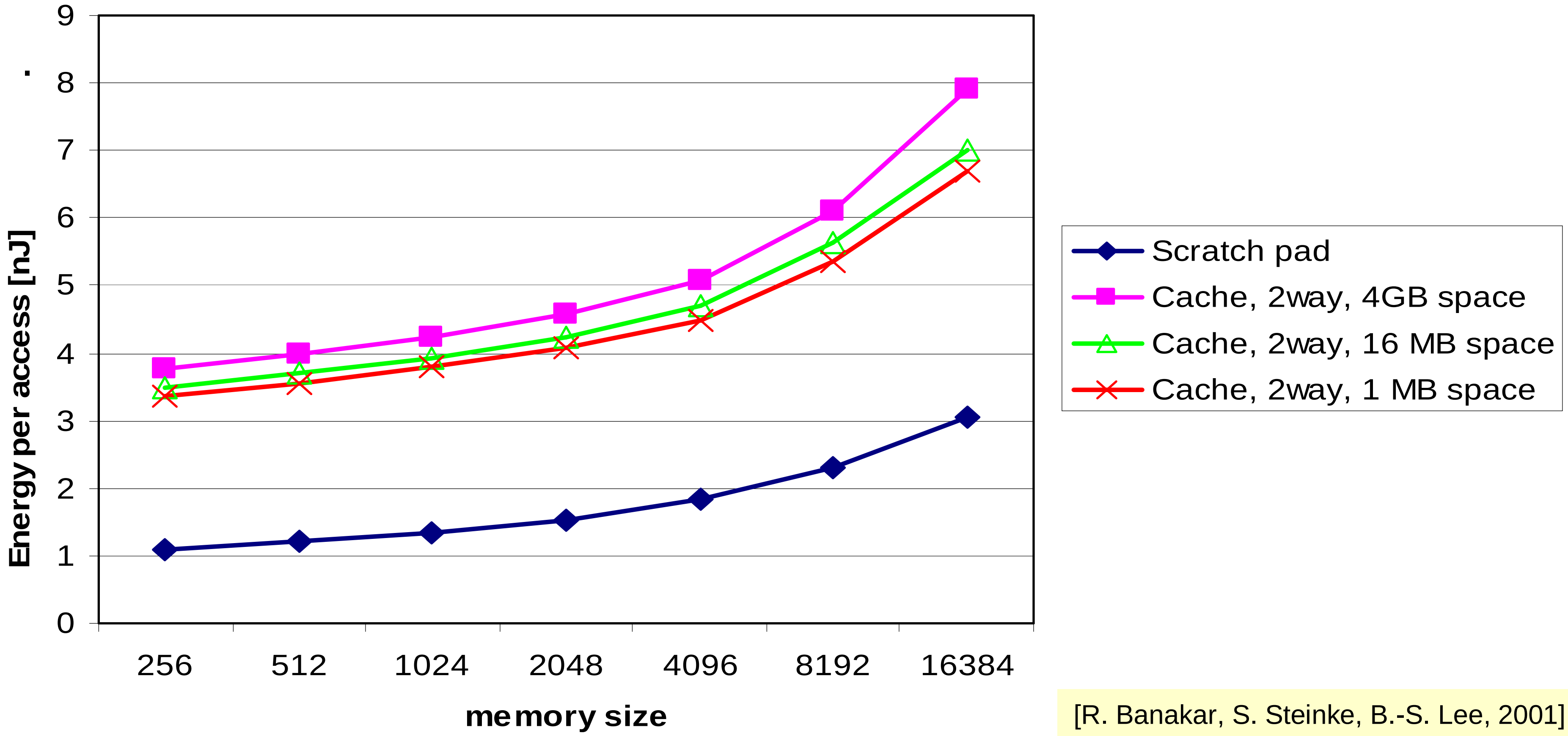
E.g.: ATMEL board with ARM7TDMI and ext. SRAM





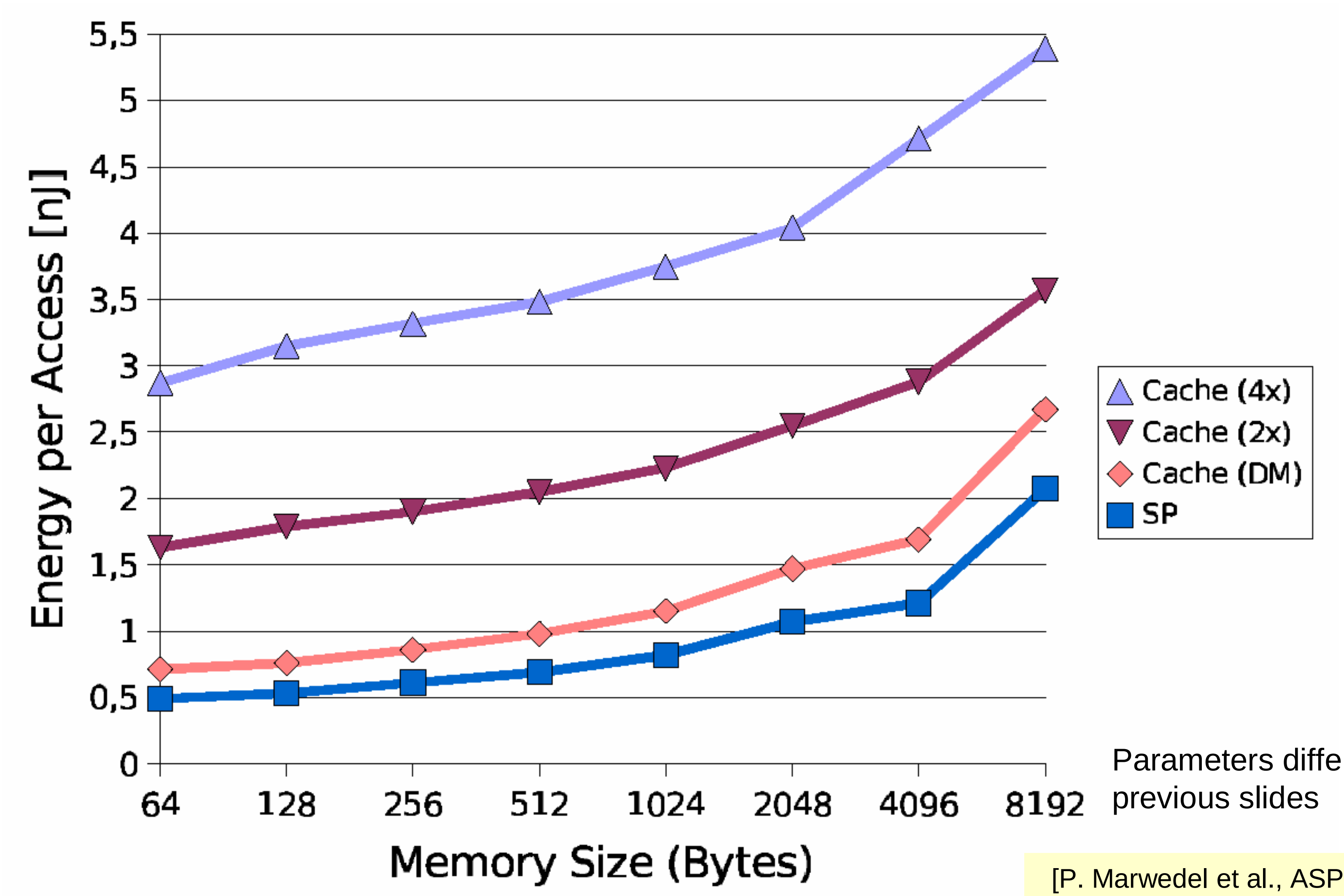
Why not just use a cache ?

2. Energy for parallel access of sets, in comparators, muxes.



[R. Banakar, S. Steinke, B.-S. Lee, 2001]

Influence of the associativity



Parameters different from previous slides

[P. Marwedel et al., ASPDAC, 2004]

Summary

- Processing
  - VLIW/EPIC processors
  - MPSoCs
- FPGAs
- Memories
  - “Small is beautiful”
    (in terms of energy consumption, access times, size)

SPARE SLIDES

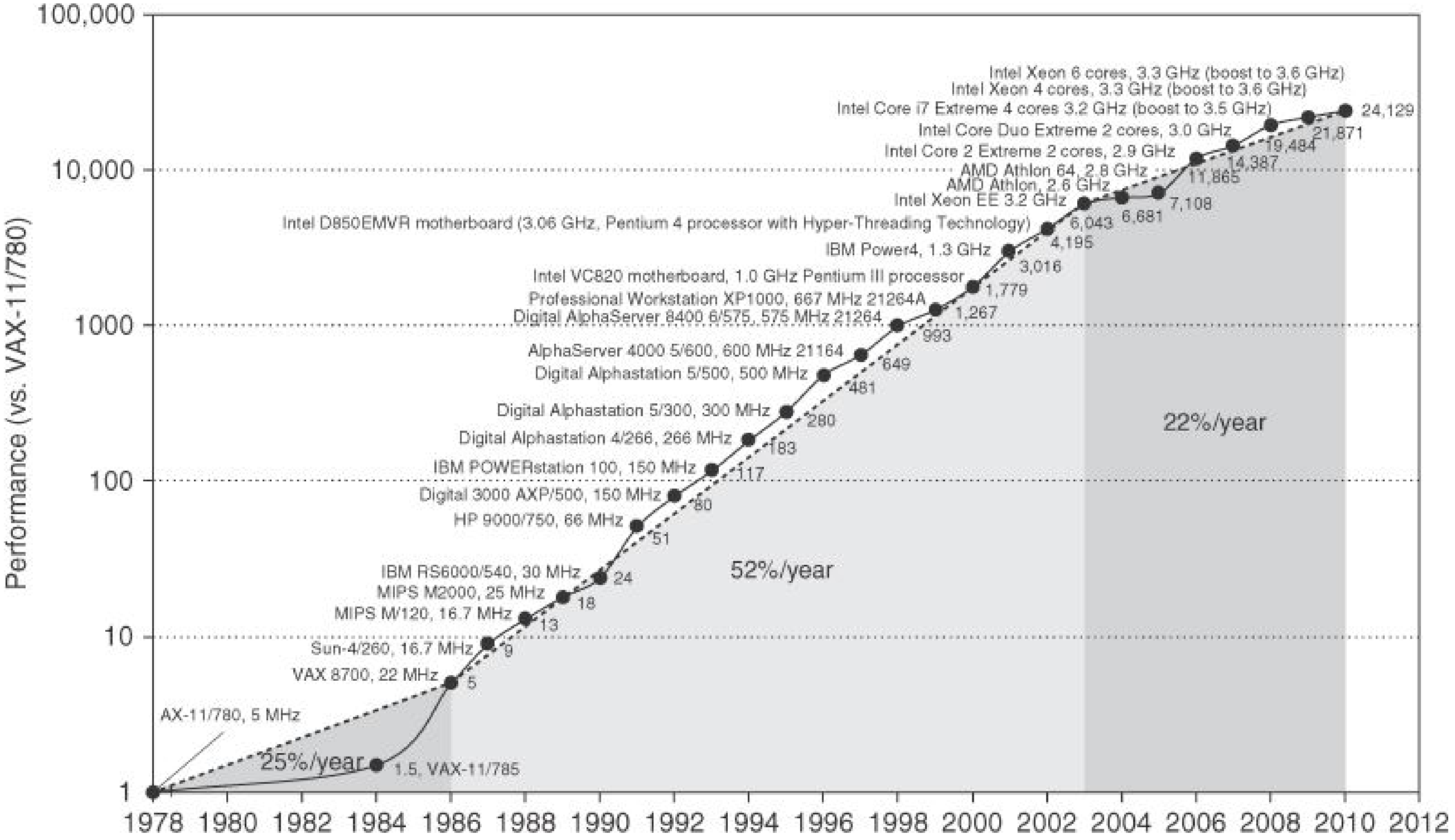
Instruction types are mapped to functional unit types

There are 4 functional unit (FU) types:

- M: Memory Unit
- I: Integer Unit
- F: Floating-Point Unit
- B: Branch Unit

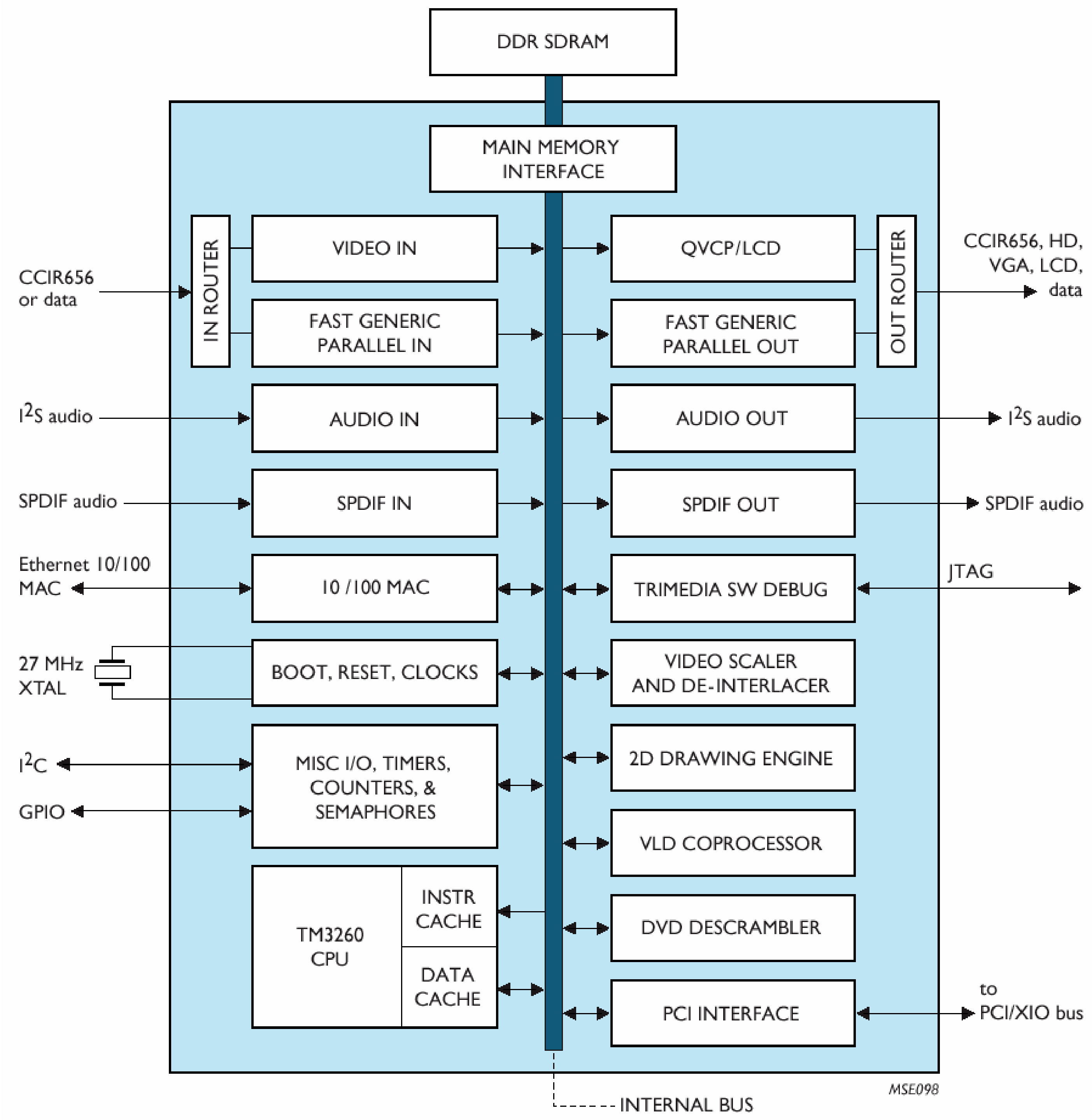
Instruction types → corresponding FU type,
except type A (mapping to either I or M-functional units).

Nevertheless, (parallel) performance keeps increasing



Copyright © 2011, Elsevier Inc. All rights Reserved.

[Hennessy/Patterson: Computer Architecture, 5th ed., 2011]

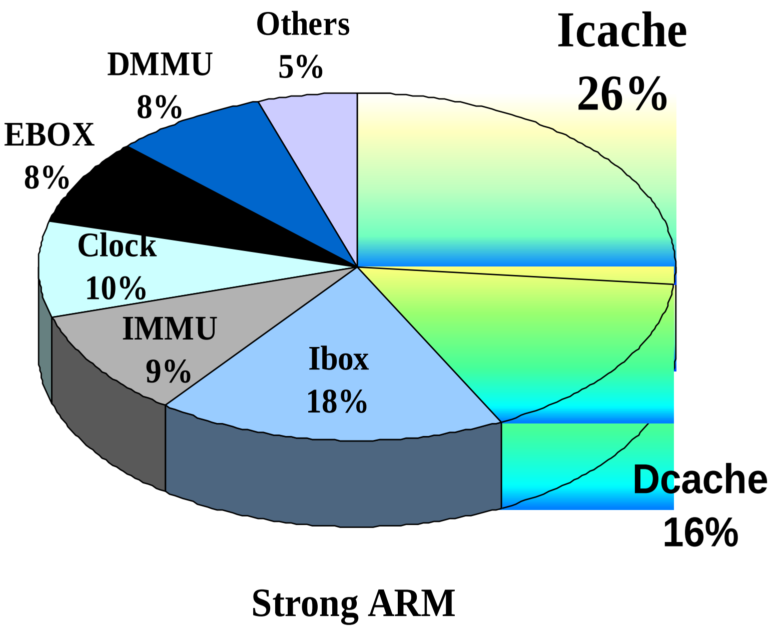


Philips TriMedia-Processor

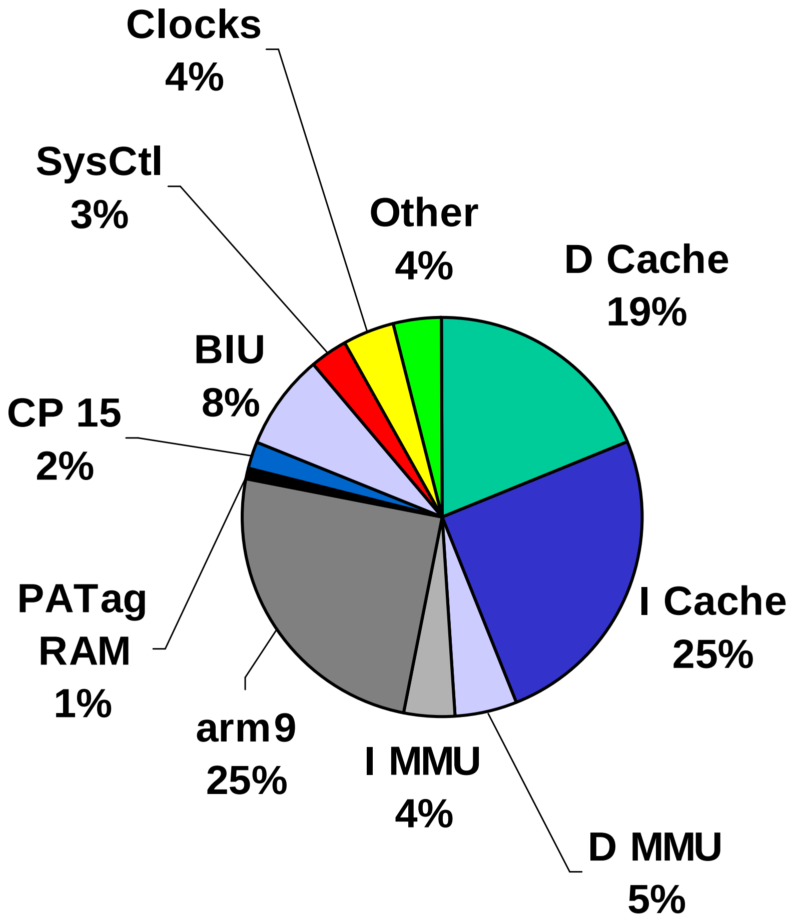
For multimedia-applications, up to 5 instructions/cycle.

http://www.nxp.com/acrobat/datasheets/PNX15XX_SER_N_3.pdf (incompatible with firefox?) © NXP

Similar information according to other sources



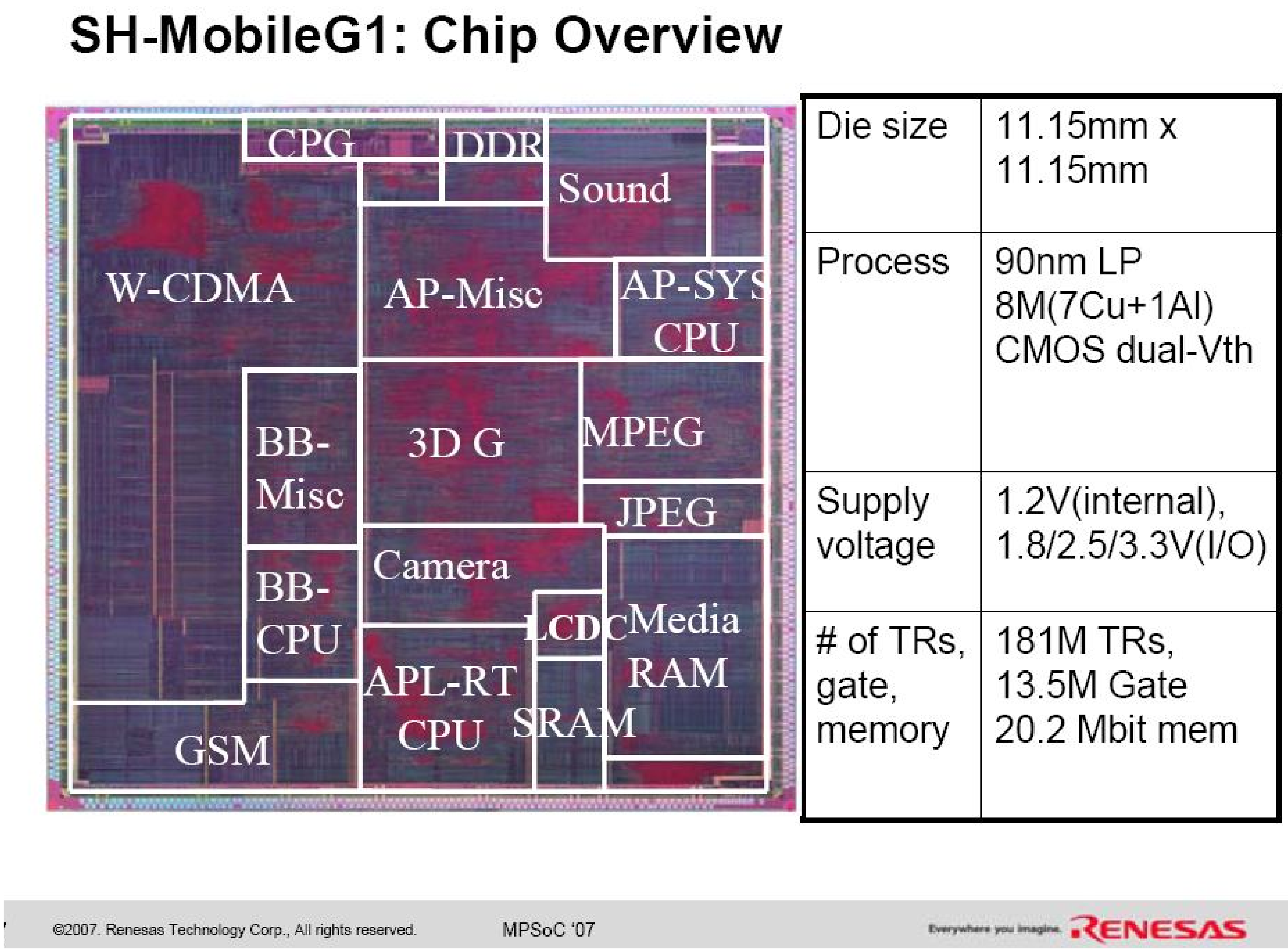
IEEE Journal of SSC
Nov. 96



[Based on slide by and ©: Osman S. Unsal, Israel Koren, C. Mani Krishna, Csaba Andras Moritz, U. of Massachusetts, Amherst, 2001]

[Segars 01 according to Vahid@ISSS01]

Multiprocessor systems-on-a-chip (MPSoCs) (2)



SH-MobileG1: Chip Overview

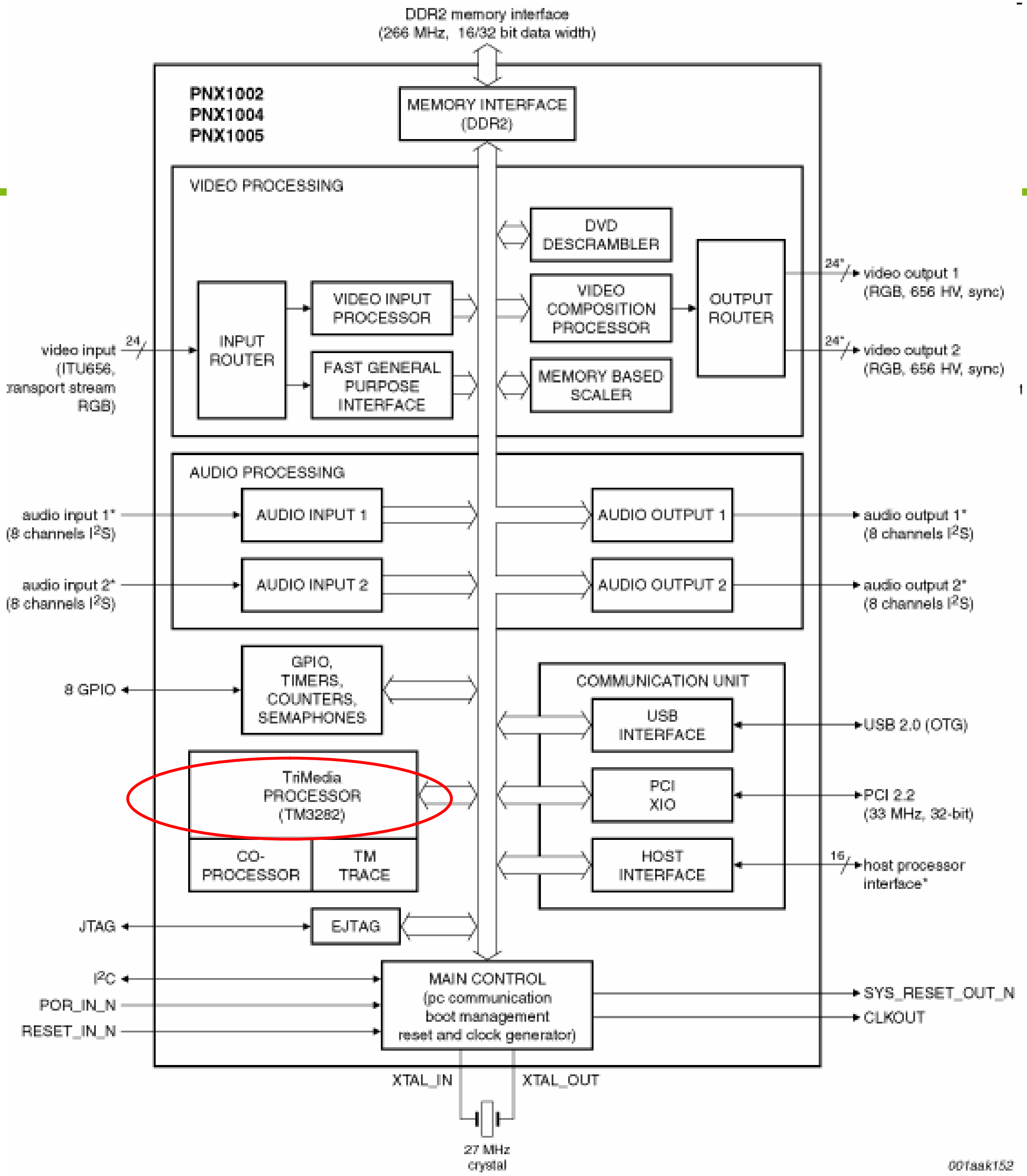
Die size	11.15mm x 11.15mm
Process	90nm LP 8M(7Cu+1Al) CMOS dual-Vth
Supply voltage	1.2V(internal), 1.8/2.5/3.3V(I/O)
# of TRs, gate, memory	181M TRs, 13.5M Gate 20.2 Mbit mem

MPSoC '07 Everywhere you imagine. RENESAS

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

TriMedia-Processor

Lastest version: starting 5-8 operations per cycle


DDR2 memory interface
(266 MHz, 16/32 bit data width)
PNX1002
PNX1004
PNX1005
MEMORY INTERFACE (DDR2)
VIDEO PROCESSING
DVD DESCRAMBLER
INPUT ROUTER
VIDEO INPUT PROCESSOR
VIDEO COMPOSITION PROCESSOR
OUTPUT ROUTER
FAST GENERAL PURPOSE INTERFACE
MEMORY BASED SCALER
video input (ITU656, transport stream RGB)
24
24*
video output 1 (RGB, 656 HV, sync)
24*
video output 2 (RGB, 656 HV, sync)
AUDIO PROCESSING
AUDIO INPUT 1
AUDIO INPUT 2
AUDIO OUTPUT 1
AUDIO OUTPUT 2
audio input 1* (8 channels I²S)
audio input 2* (8 channels I²S)
audio output 1* (8 channels I²S)
audio output 2* (8 channels I²S)
8 GPIO
GPIO, TIMERS, COUNTERS, SEMAPHONES
COMMUNICATION UNIT
USB INTERFACE
USB 2.0 (OTG)
TriMedia PROCESSOR (TM3282)
PCI XIO
PCI 2.2 (33 MHz, 32-bit)
CO-PROCESSOR
TM TRACE
HOST INTERFACE
16
host processor interface*
JTAG
EJTAG
I²C
POR_IN_N
RESET_IN_N
MAIN CONTROL (pc communication boot management reset and clock generator)
SYS_RESET_OUT_N
CLKOUT
XTAL_IN
XTAL_OUT
27 MHz crystal
001aak152

http://www.tridentmicro.com/producttree/stb/media-processor/pnx100x/
© Trident, 2010
(since 2012/1 under chapter 11)