

Samsung's mobile lines

Dezső Sima

Vers. 1.2

Mai 2018

Contents

- 1. Samsung's earliest mobile SOCs
- 2. Overview of Samsung's quad- and octa core mobile SOCs
- 3. Quad- and octa core SMPs
- 4. Octa core big.LITTLE mobile SOCs with exclusive cluster allocation
- 5. Octa core big.LITTLE mobile SOCs supporting GTS
- 6. References

1. Samsung's earliest mobile SOCs

1. Samsung's earliest mobile SOCs (1)

1. Samsung's earliest mobile SOCs -1 [71]

Model Number	Technology	CPU ISA	CPU	GPU	Memory tech.	Availability	Utilizing devices
S3C44B0	0.25 μm CMOS	ARMv4	66 MHz single-core ARM7 (ARM7TDI)	LCD controller	FP, EDO, SDRAM	2000	Juice Box, Danger Hiptop
S5L2010		ARMv5	176 MHz single-core ARM9 (ARM946E-S)	LCD controller	SDRAM, EDO		
S3C2410	0.18 μm CMOS	ARMv4	200/266 MHz single-core ARM9 (ARM920T)	LCD controller	SDRAM	2003	HP iPAQ H1930/H1937/H1940/rz1717,, Acer n30/n35/d155, Palm Z22, LG LN600, Typhoon MyGuide 3610 GO
S3C2412	0.13 μm CMOS	ARMv5	200/266 MHz single-core ARM9 (ARM926EJ-S)	LCD controller	mSDRAM		
S3C2413	0.13 μm LP	ARMv5	266 MHz single-core ARM9 (ARM926EJ-S)	LCD controller	mSDRAM, mDDR		
S3C2440	0.13 μm CMOS	ARMv4	300/400/533 MHz single-core ARM9 (ARM920T)	LCD controller	SDRAM	2004	HP iPAQ rx3115/3415/3417/3715, Everex E900, Acer n300/311, Typhoon MyPhone M500, Mio p550/P350/C710 Digi-Walker
S3C22442	0.13 μm CMOS	ARMv4	300/400 MHz single-core ARM9 (ARM920T)	LCD controller	mSDRAM		
S3C2443 ¹		ARMv4	400/533 MHz single-core ARM9 (ARM920T)	LCD controller	SDRAM, mSDRAM, mDDR	2007	Asus R300/R600/R700, Mio Digi-Walker (C620T), LG LN8xx, JL7220, Navigon 8300/8310

1. Samsung's earliest mobile SOCs (2)

Samsung's earliest mobile SOCs -2 [71]

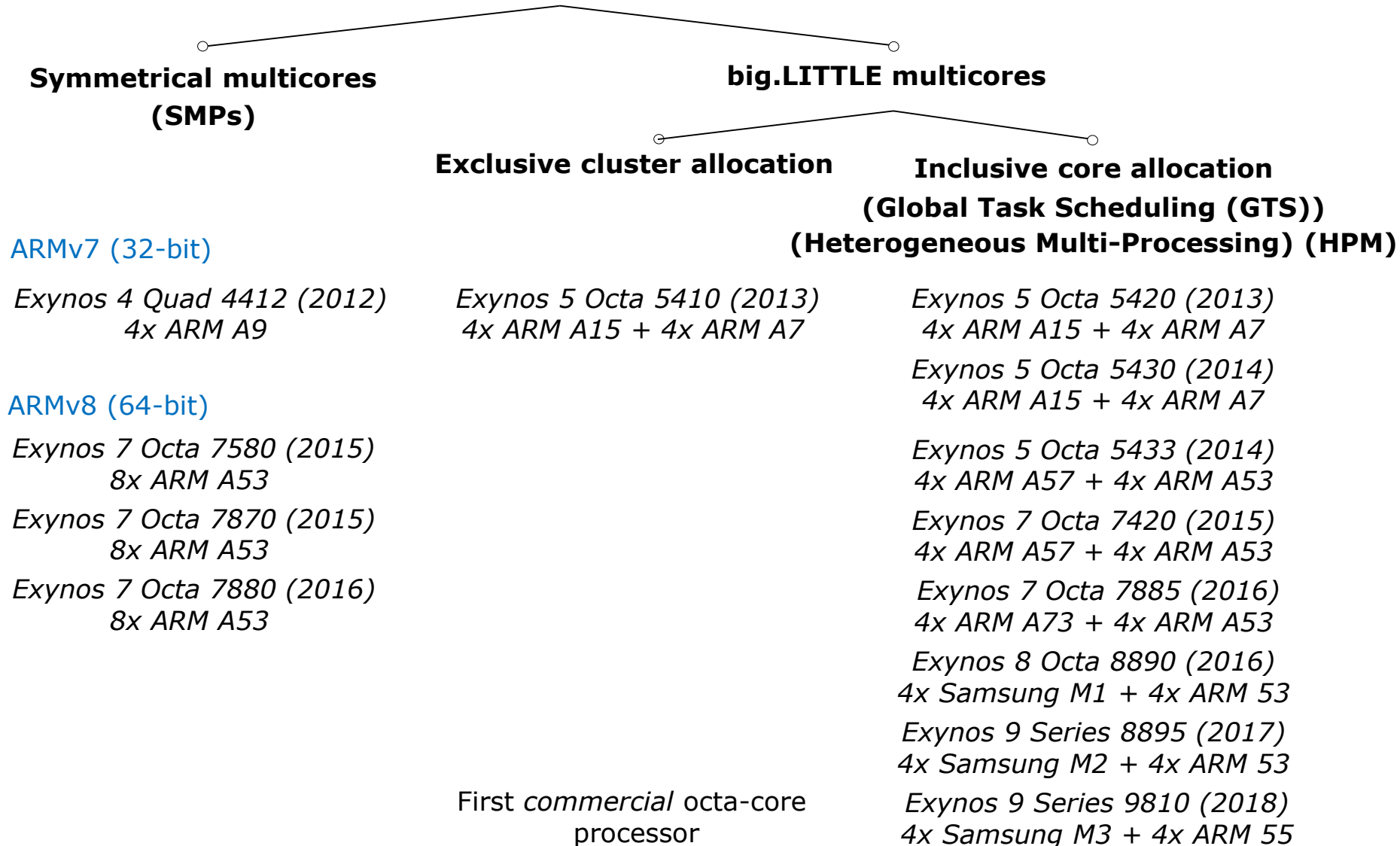
Model Number	Technology	CPU ISA	CPU	GPU	Memory tech.	Availability	Utilizing devices
S5L8900	90 nm	ARMv6	412 MHz single-core ARM11	PowerVR MBX Lite	eDRAM	2007	Apple iPhone, Apple iPod touch 1G, Apple iPhone 3G
S3C2416	65 nm LP	ARMv5	400 MHz single-core ARM9 (ARM926EJ)	2D graphics accelerator	SDRAM, mSDRAM, mDDR, DDR2	2008	iconX G310, HP Prime
S3C2450	65 nm LP CMOS	ARMv5	400/533 MHz single-core ARM9 (ARM926EJ)	2D graphics accelerator	SDRAM, mSDRAM, mDDR, DDR2	2008	Mio Moov 500/510/560/S568/580, Getac PS535F, MENQ EasyPC E720/E790, Hivision PWS0890AW,SMiT MTV-PND530 8GB
S3C6410	65 nm LP	ARMv6	533/667/800 MHz single-core ARM11 (ARM1176ZJF-S)	FIMG 3DSE graphics accelerator	mSDRAM, mDDR	2009	Samsung S5620 Monte
S5P6442	45 nm	ARMv6	533/667 MHz single-core ARM11	FIMG 3DSE graphics accelerator		2010	
S5P6450		ARMv6	533/667/800 MHz single-core ARM11 (ARM1176JZF-S)	3D graphics accelerator	mDDR, mDDR2, LPDDR	2010	
S5PC100	65 nm	ARMv7	667/833 MHz single-core ARM Cortex-A8	PowerVRSG X535	LPDDR2, DDR2	2009	Apple iPhone 3GS

2. Overview of Samsung's quad- and octa core mobile SOCs

2. Overview of Samsung's quad- and octa core mobile SOCs

2. Overview of Samsung's quad- and octa core mobile SOCs

Samsung's quad- and octa core mobile SOCs



3. Quad- and octa core SMPs

3. Quad- and octa core SMPs (1)

3. Quad- and octa core SMPs

Samsung's quad- and octa core mobile SOCs

Symmetrical multicores (SMPs)

ARMv7 (32-bit)

Exynos 4 Quad 4412 (2012)
4x ARM A9

ARMv8 (64-bit)

Exynos 7 Octa 7580 (2015)
8x ARM A53

Exynos 7 Octa 7870 (2015)
8x ARM A53

Exynos 7 Octa 7880 (2016)
8x ARM A53

big.LITTLE multicores

Exclusive cluster allocation

Exynos 5 Octa 5410 (2013)
4x ARM A15 + 4x ARM A7

*First commercial octa-core
processor*

Inclusive core allocation

(Global Task Scheduling (GTS)) (Heterogeneous Multi-Processing) (HPM)

Exynos 5 Octa 5420 (2013)
4x ARM A15 + 4x ARM A7

Exynos 5 Octa 5430 (2014)
4x ARM A15 + 4x ARM A7

Exynos 5 Octa 5433 (2014)
4x ARM A57 + 4x ARM A53

Exynos 7 Octa 7420 (2015)
4x ARM A57 + 4x ARM A53

Exynos 7 Octa 7885 (2016)
4x ARM A73 + 4x ARM A53

Exynos 8 Octa 8890 (2016)
4x Samsung M1 + 4x ARM 53

Exynos 9 Series 8895 (2017)
4x Samsung M2 + 4x ARM 53

Exynos 9 Series 9810 (2018)
4x Samsung M3 + 4x ARM 55

3. Quad- and octa core SMPs (2)

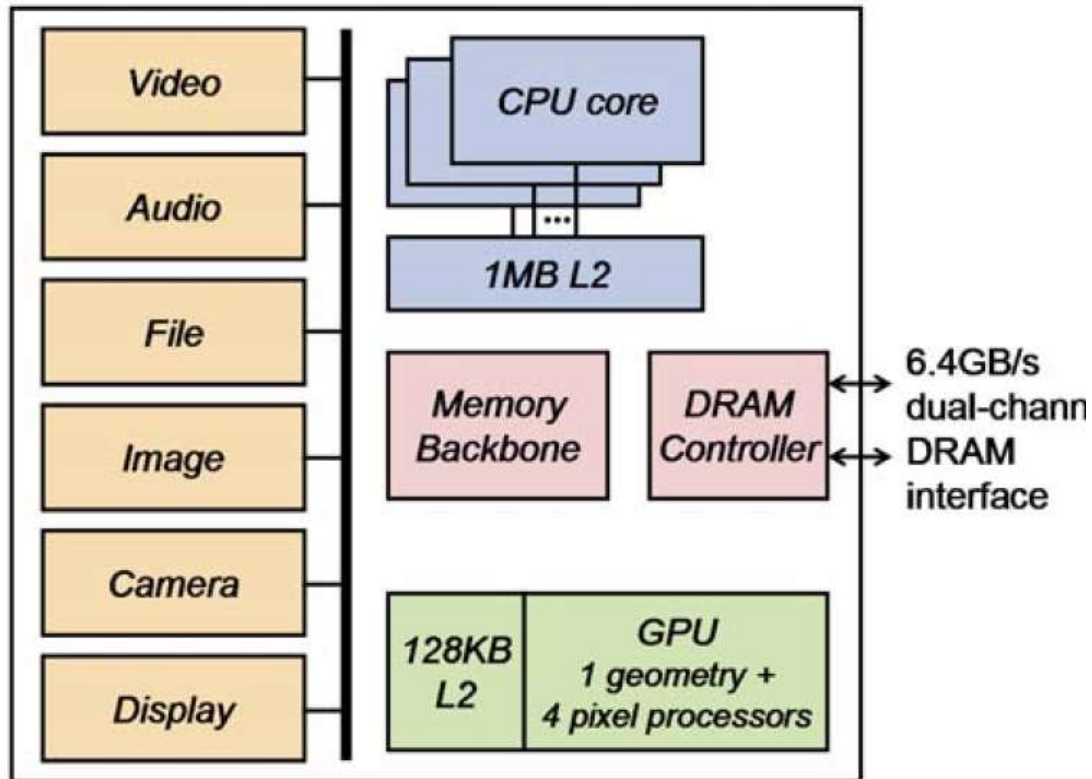
Main features of Samsung's quad- and octa core SMPs

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Microarch.	cores	fc (GHz)				
Exynos 4 Quad (Exynos 4412)	32 nm HKMG	ARM v7	Cortex-A9	4	1.4	ARM Mali-T400 MP4 @ 440 MHz; 15.8 GFLOPS	32-bit DCh. DDR3-800 LPDDR3-800 (6.4 GB/sec)	2012	Samsung Galaxy SIII Samsung Galaxy Note 2
Exynos 7 Octa (Exynos 7580)	20 nm FinFET	ARM v8-A	Cortex-A53	8	1.5	Mali-T720 MP2 @ 668 MHz; 34 GFLOPS (FP16)	32-bits DCh. LPDDR3-1866 (14.9 GB/s)	Q2 2015	Samsung Galaxy A5/ Samsung Galaxy A7
Exynos 7 Octa (Exynos 7870)	14 nm FinFET	ARM v8-A	Cortex-A53	8	1.7	Mali-T830 MP2 @ 700 MHz; 47.6 GFLOPS (FP16)	32-bits DCh. LPDDR3-1866 (14.9 GB/s)	Q1 2016	Samsung Galaxy Tab A
Exynos 7 Octa (Exynos 7880)	14 nm FinFET	ARM v8-A	Cortex-A53	8	1.9	Mali-T830 MP3	32-bits DCh. LPDDR4x	2016	Samsung Galaxy A5/ Samsung Galaxy A7

3. Quad- and octa core SMPs (3)

Example: Samsung Exynos 4412 4-core SMP (2012)

Architecture block diagram



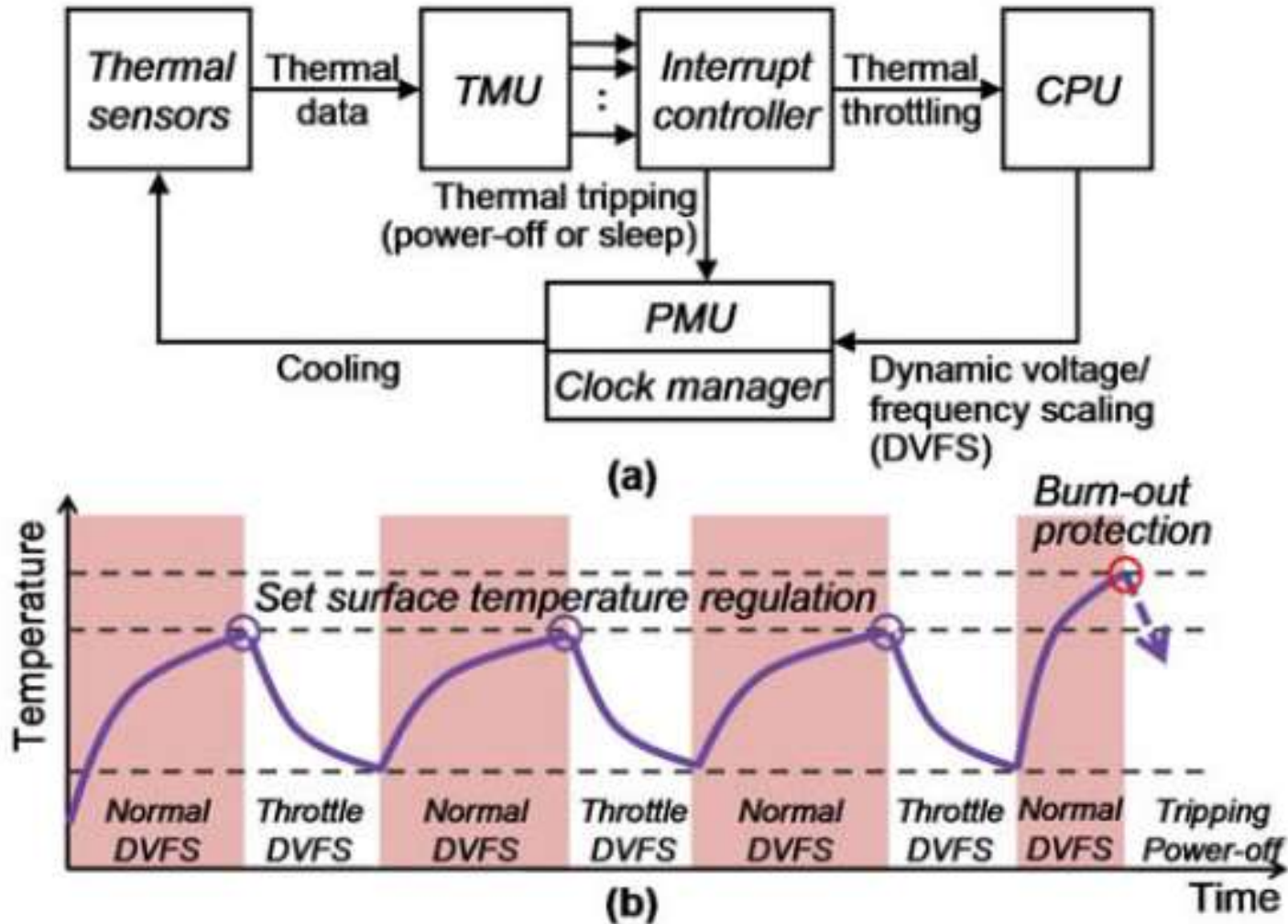
3. Quad- and octa core SMPs (4)

Power management of the Exynos 4 Quad (4412) (2012)

- It has a **platform level power management unit**, called the **PMU**.
- There are **four power planes**:
 - two for the CPUs, one for the GPU and one for the DRAM controller and the other functional blocks.
- **Per-core DVFS** is implemented [63].
- **Power gating** is used for each core and all major functional units.
- There is also a **separate thermal management unit (TMU)**.
- See the subsequent slide for an illustration of power and thermal management.

3. Quad- and octa core SMPs (5)

Power and thermal management of Samsung's Exynos 4412 (2012) [1]



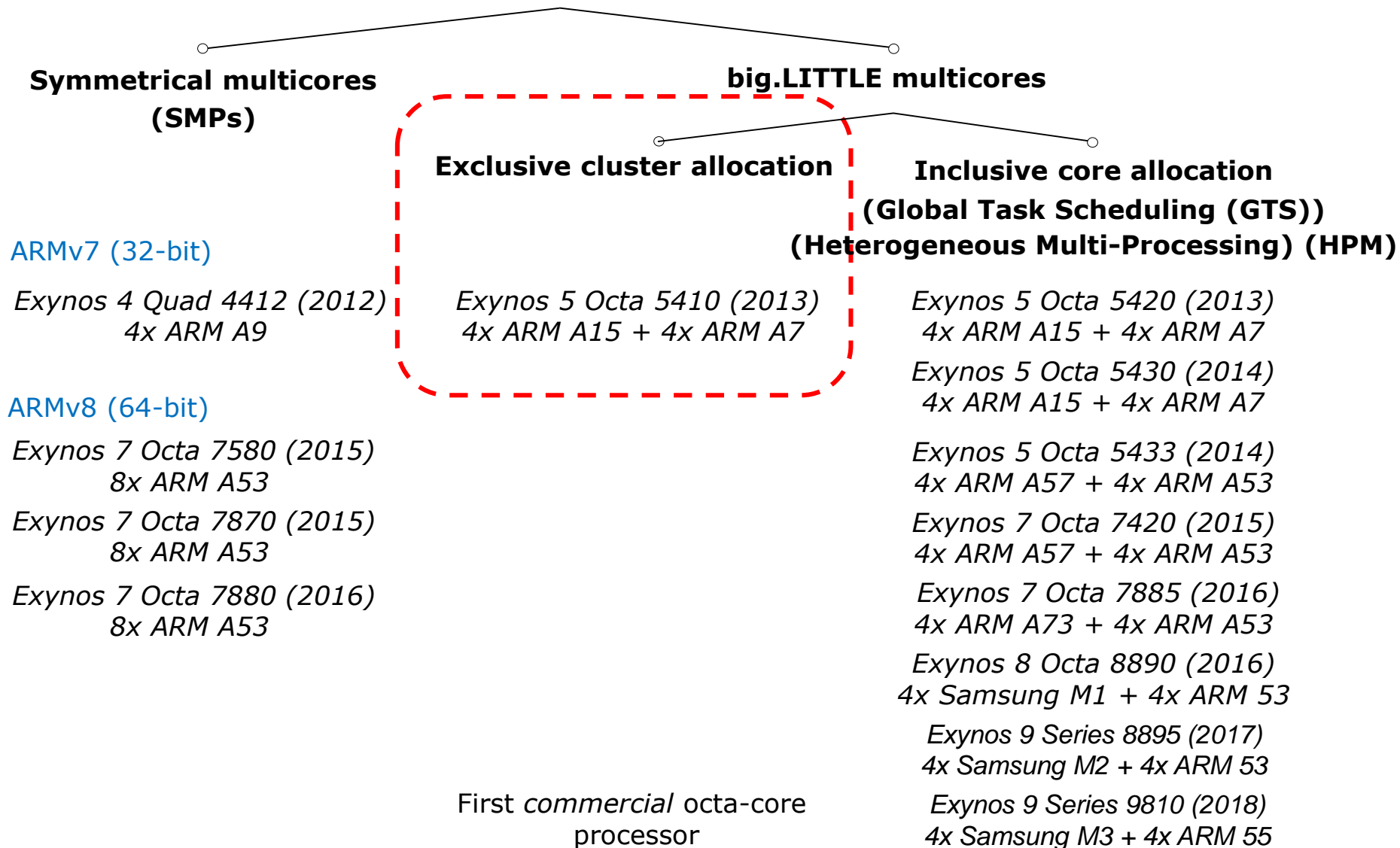
PMU: Power Management Unit
TMU: Thermal Management Unit

4. Octa core big.LITTLE mobile SOCs with exclusive cluster allocation

4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (1)

4. Octa core big.LITTLE mobile SOCs with exclusive cluster allocation

Samsung's quad- and octa core mobile SOCs



4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (2)

The world's first octa core mobile processor: Samsung's Exynos Octa 5410 (2013) [2]

- It implements the 32-bit ARMv7 ISA.
- It operates in the big.LITTLE configuration with **cluster allocation for scheduling**.
- Announced in 11/2012, **launched** in Galaxy S4 models in **4/2013**.

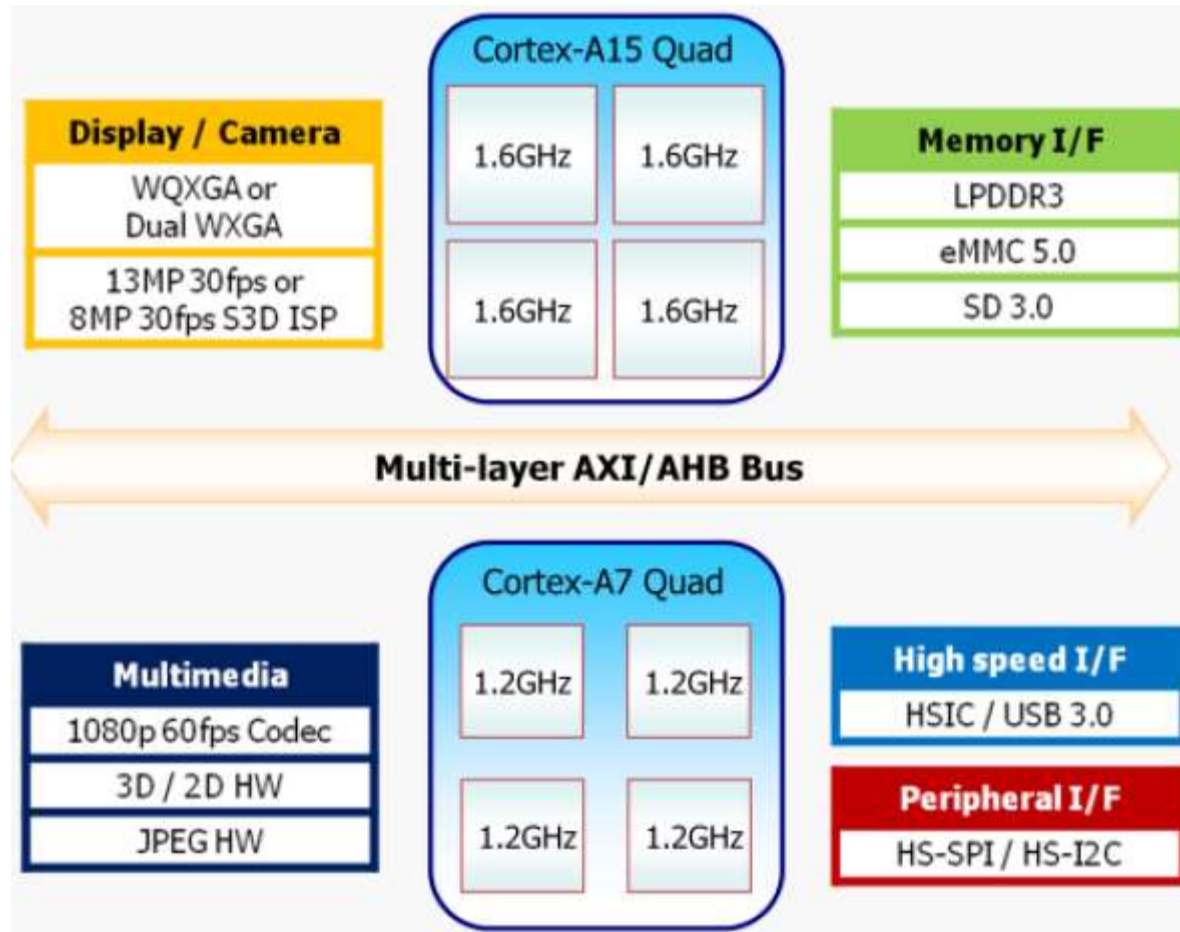
4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (3)

Main features of Samsung's Exynos 5410 octa core big.LITTLE mobile SOC with exclusive cluster allocation

SoC		CPU				GPU	Memory technology	Availability	Utilized in the devices (examples)
Model number	fab.	Instr. set	Microarch.	cores	fc (GHz)				
Exynos 5 Octa (<i>Exynos 5410</i>)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8 1.2	IT PowerVR SGX544MP3 @ 480 MHz 49 GFLOPS	32-bit DCh LPDDR3-1600 (12.8 GB/sec)	Q2 2013	Samsung Galaxy S4 I9500, ZTE Grand S II TD,

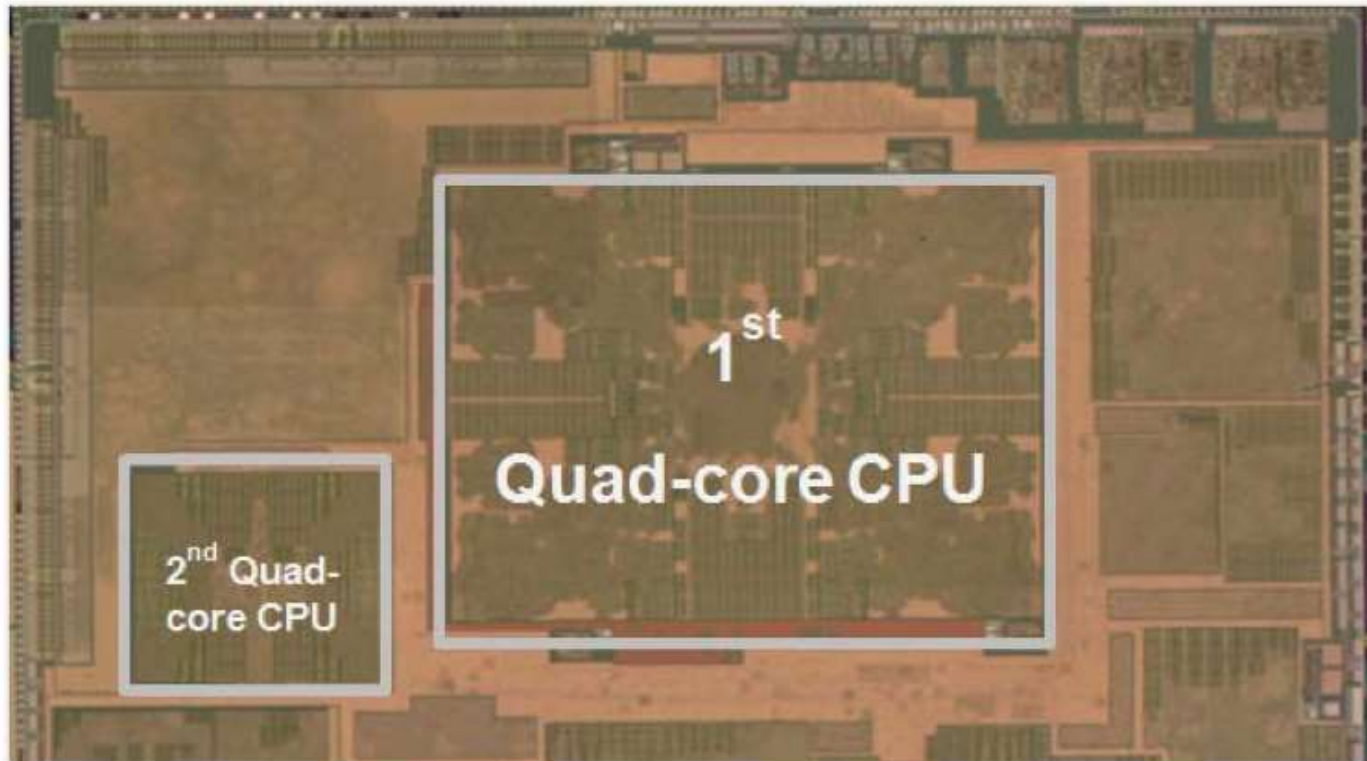
4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (4)

Block diagram of Samsung's Exynos 5 Octa 5410 [2]



4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (5)

Assumed die photo of Samsung's Exynos 5 Octa 5410 [3]

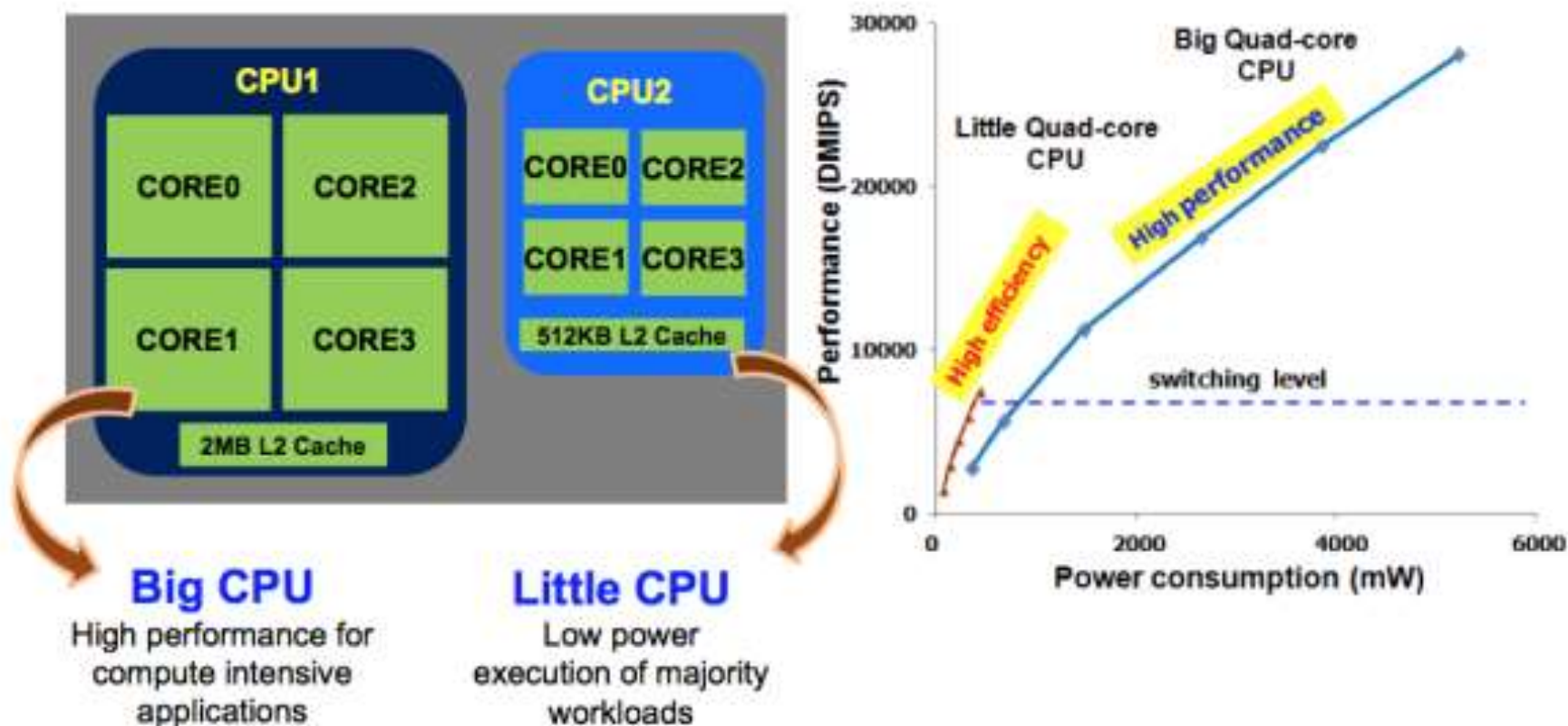


Revealed at the International Solid-State Circuit Conference (ISSCC) in 2/2013 without specifying the chip designation [3].

4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (6)

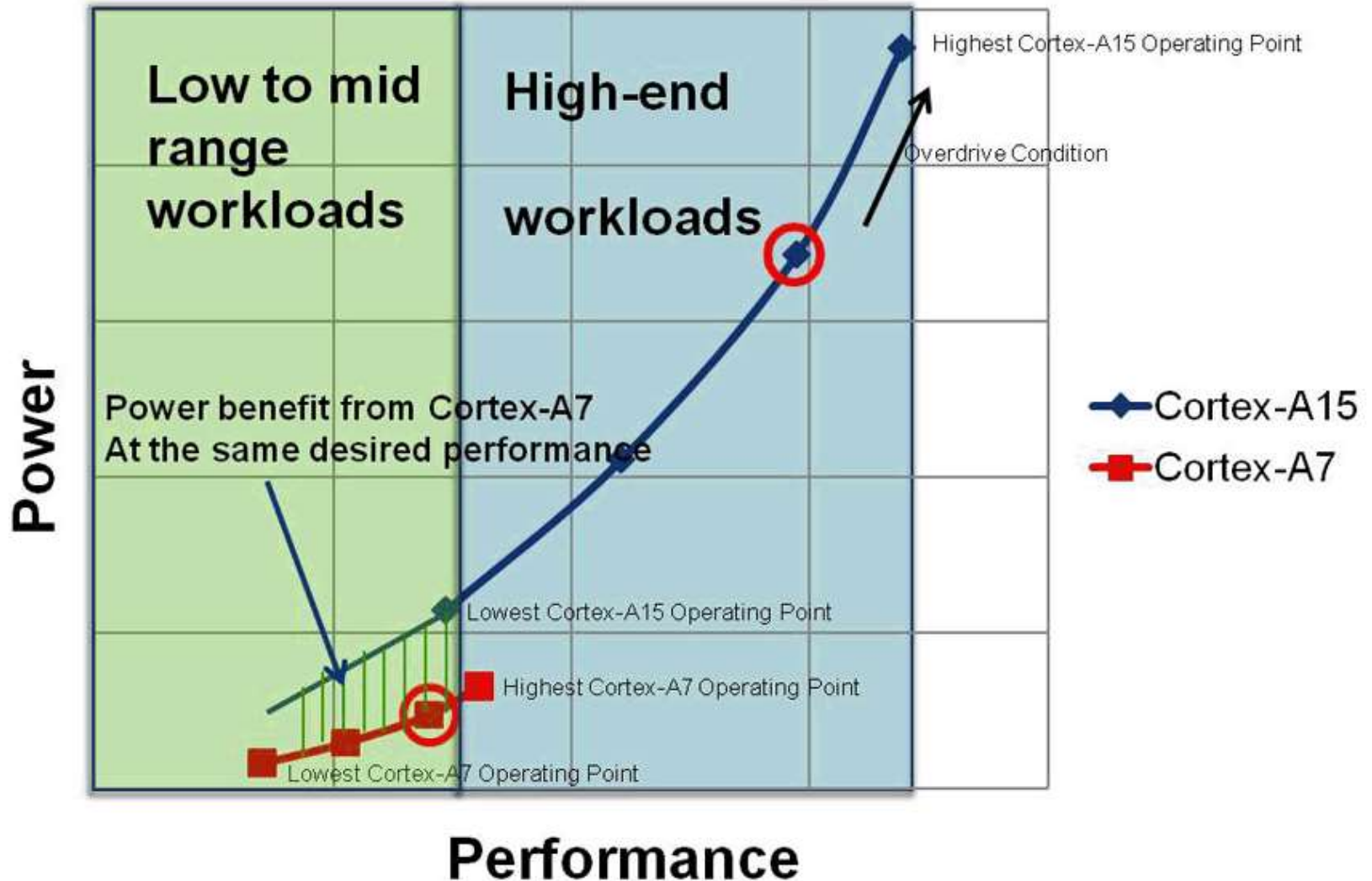
Principle of operation of the Exynos 5 Octa 5410 [3]

- For low performance demand the "Little CPU" and for higher performance demand the "Big CPU" is used.
- At a given switching level the scheduler performs a **cluster switch**.



4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (7)

Performance points of operation of the LITTLE and big clusters [4]



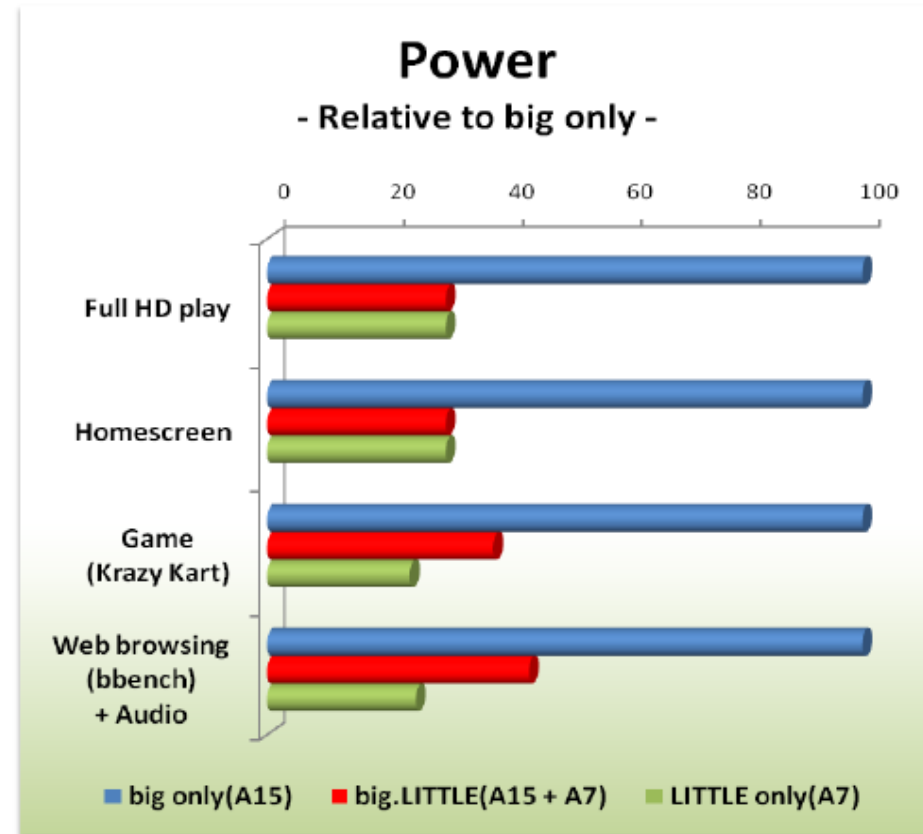
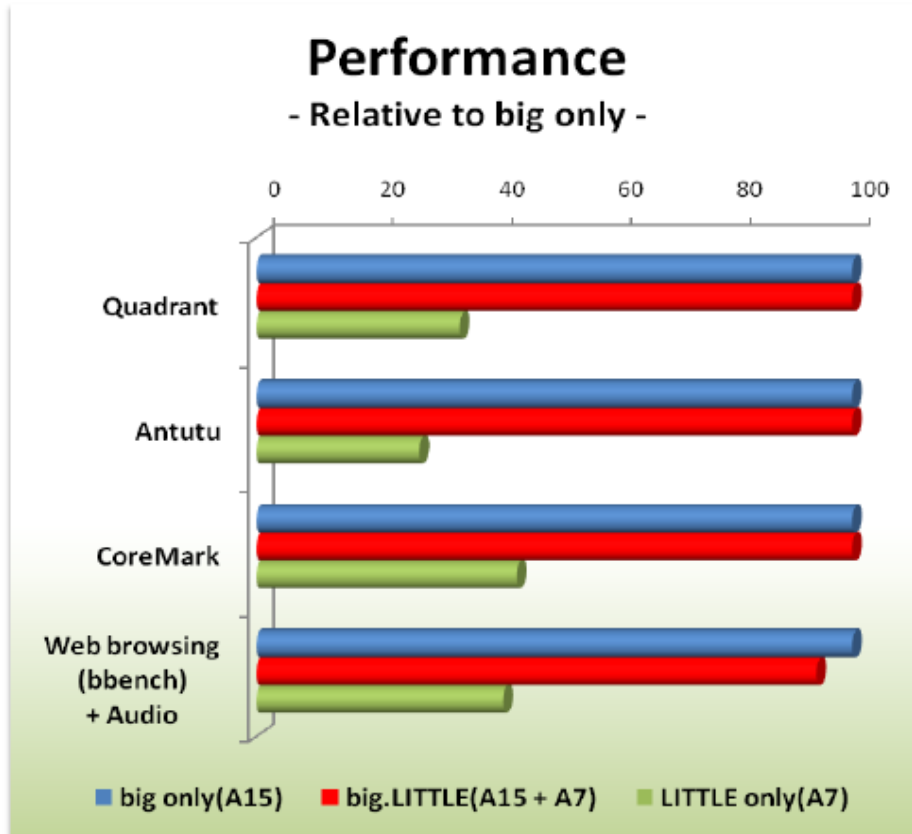
4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (8)

Remark

- In a White Paper [66] Samsung's engineers compare asynchronous architectures with per-core DVFS and synchronous big.LITTLE architectures concerning their energy efficiency.
- Their conclusion is that **concerning the energy efficiency** (e.g. power consumption performance) **or net energy consumption the big.LITTLE architecture is superior vs. the per-core DVFS** for the majority of commonly used applications, such as e-mail messaging, web browsing or multimedia playback.
- The reason is performance degradations due to cache misses or transferring data between different cores.
- Based on this finding **Samsung implemented the big.LITTLE technology in the Exynos 5410 with synchronous DVFS** (meaning that all cores within a cluster run at the same voltage and frequency).

4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (9)

Performance and power results of the Exynos 5 Octa 5410 [2]



4. Octa core big.LITTLE mobile SOC with exclusive cluster allocation (10)

Remark

- According to sources there was a troublesome [bug in the CCI-400 coherent bus interface](#) [3].
- Thus, [Samsung disabled the coherency between the two clusters](#), and as a consequence after cluster switches they need to invalidate all caches.
- Obviously, this has impeded performance and battery life.

5. Octa core big.LITTLE mobile SOCs supporting GTS

- 5.1: Octa core big.LITTLE mobile SOCs supporting GTS - Overview
- 5.2: The world's first octa core big.LITTLE mobile SOC supporting GTS: The Exynos 5 Octa 5420 (2013)
- 5.3: Samsung's first 64-bit octa core big.LITTLE SOC supporting GTS and operating in the ARMv8 AArch32 mode: the Exynos 7 Octa 5433 (2014)
- 5.4: Samsung's first 64-bit octa core big.LITTLE SOC operating in the ARMv8 AArch64 mode: the Exynos 7 Octa 7420 (2015)
- 5.5: Samsung's first SoC including an in-house designed CPU core (the M1): the Exynos 8 Octa 8890 (2016)
- 5.6: Samsung's first 10 nm SOC: the Exynos 9 8895 (2017)
- 5.7: Samsung's first SOC supporting the DynamIQ cluster technology: the Exynos 9 9810 (2018)

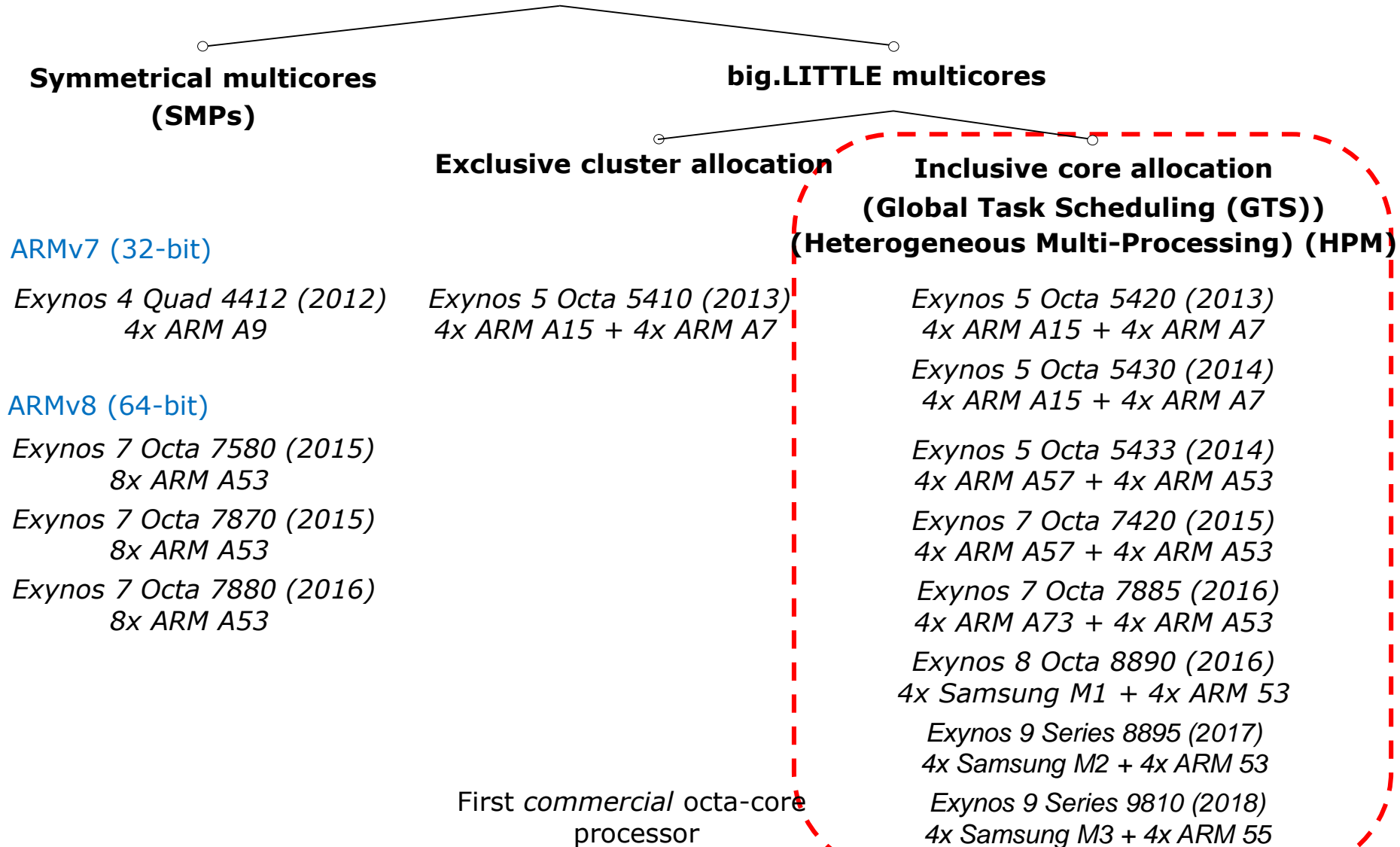
5.1 Octa core big.LITTLE mobile SOCs supporting GTS

- Overview

5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (1)

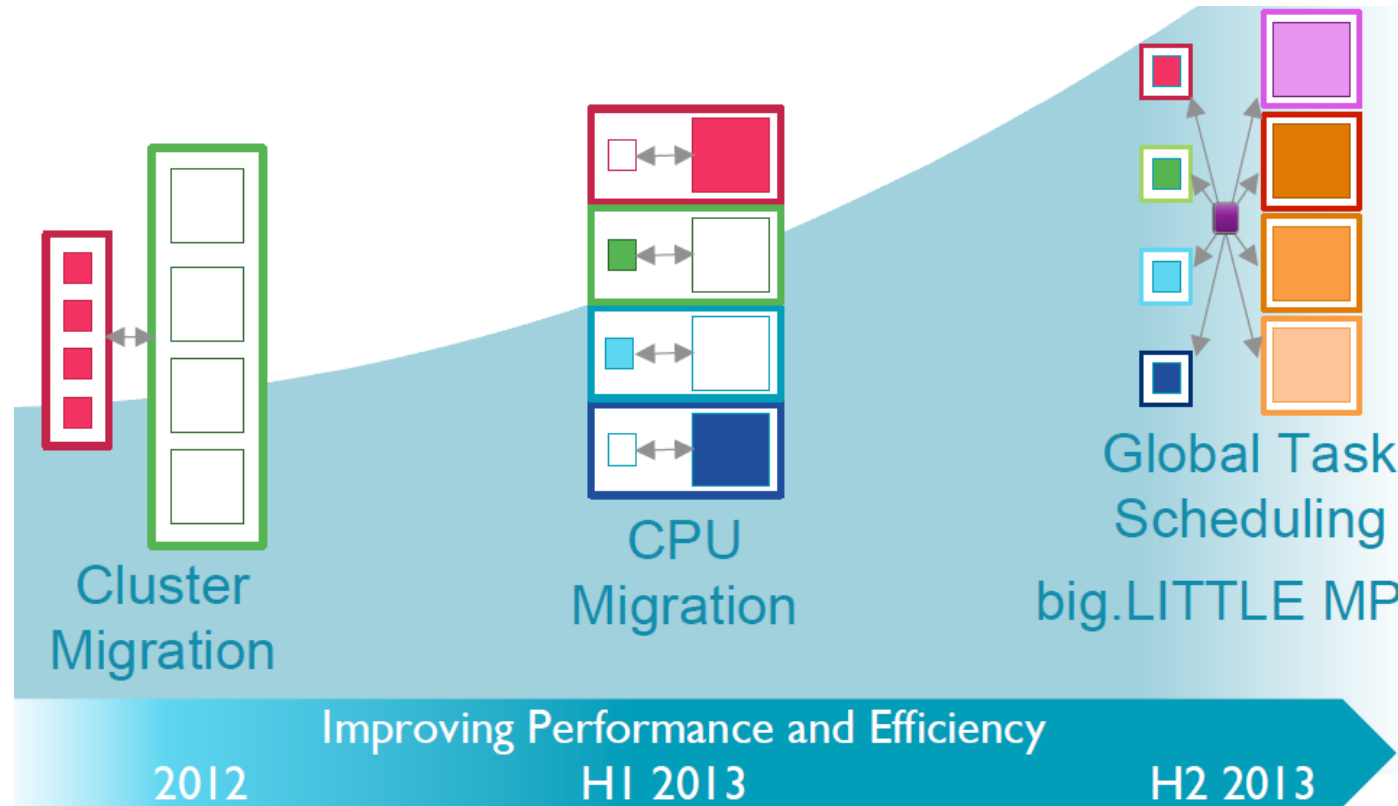
5.1 Octa core big.LITTLE mobile SOCs supporting GTS – Overview

Samsung's quad- and octa core mobile SOCs



5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (2)

Evolution of the scheduling techniques [5]

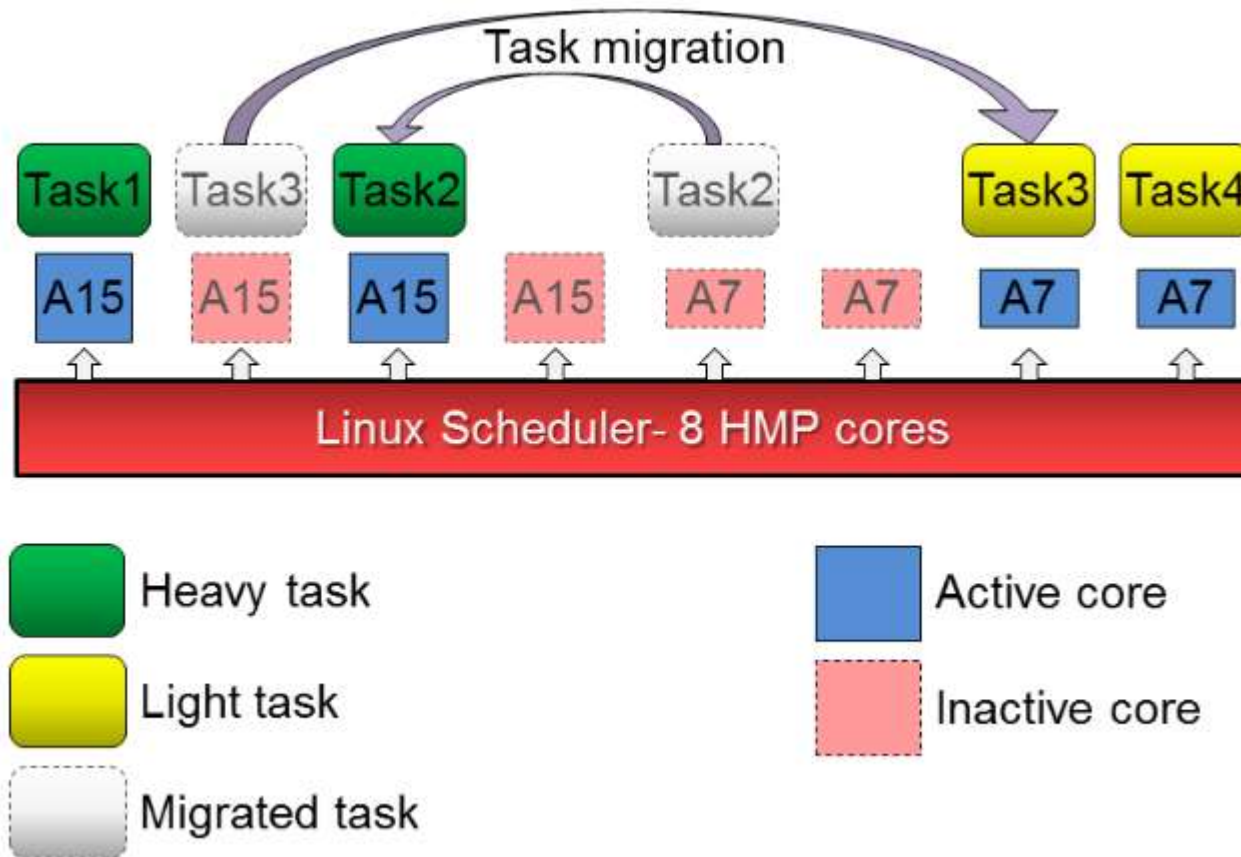


Remark

- In **CPU allocation** there are **big.LITTLE core pairs** and the scheduler can activate either the **big** or the **LITTLE** core from each core pair.
- **No commercial implementation** is known using the CPU allocation.

5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (3)

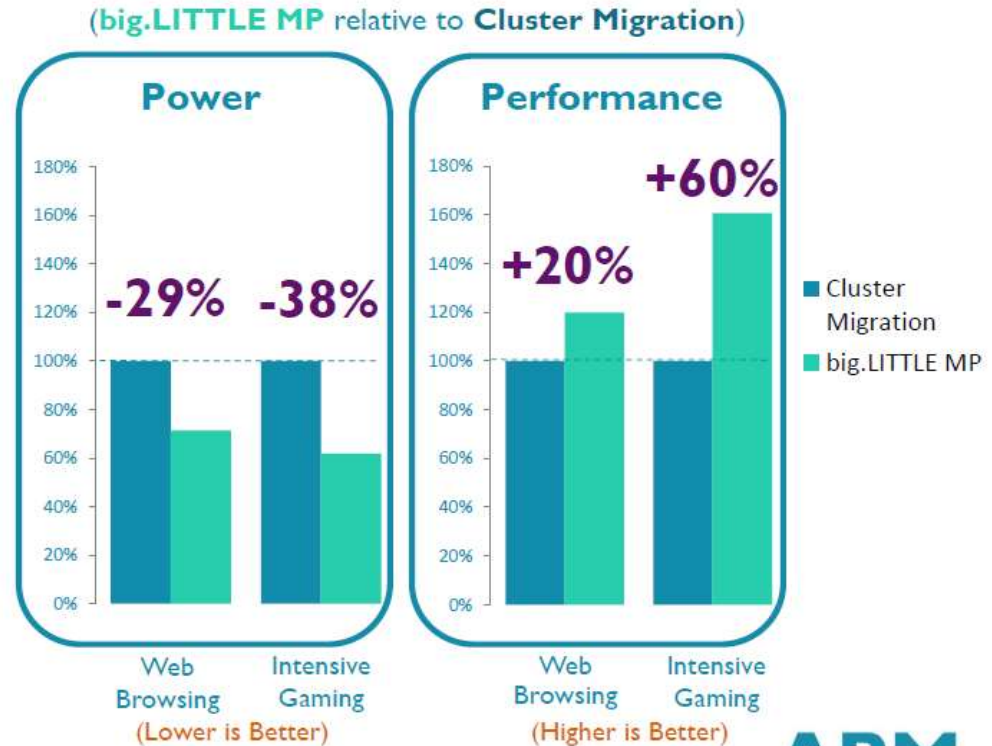
Example for Global Task Scheduling (GTS) [72]



5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (4)

Benefits of GTS scheduling (designated as big.LITTLE MP in the Figure below) vs. exclusive cluster allocation [6]

- Delivers higher power efficiency
- Extends battery life
- Improves user experience



5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (5)

Overview of early big.LITTLE implementations supporting GTS

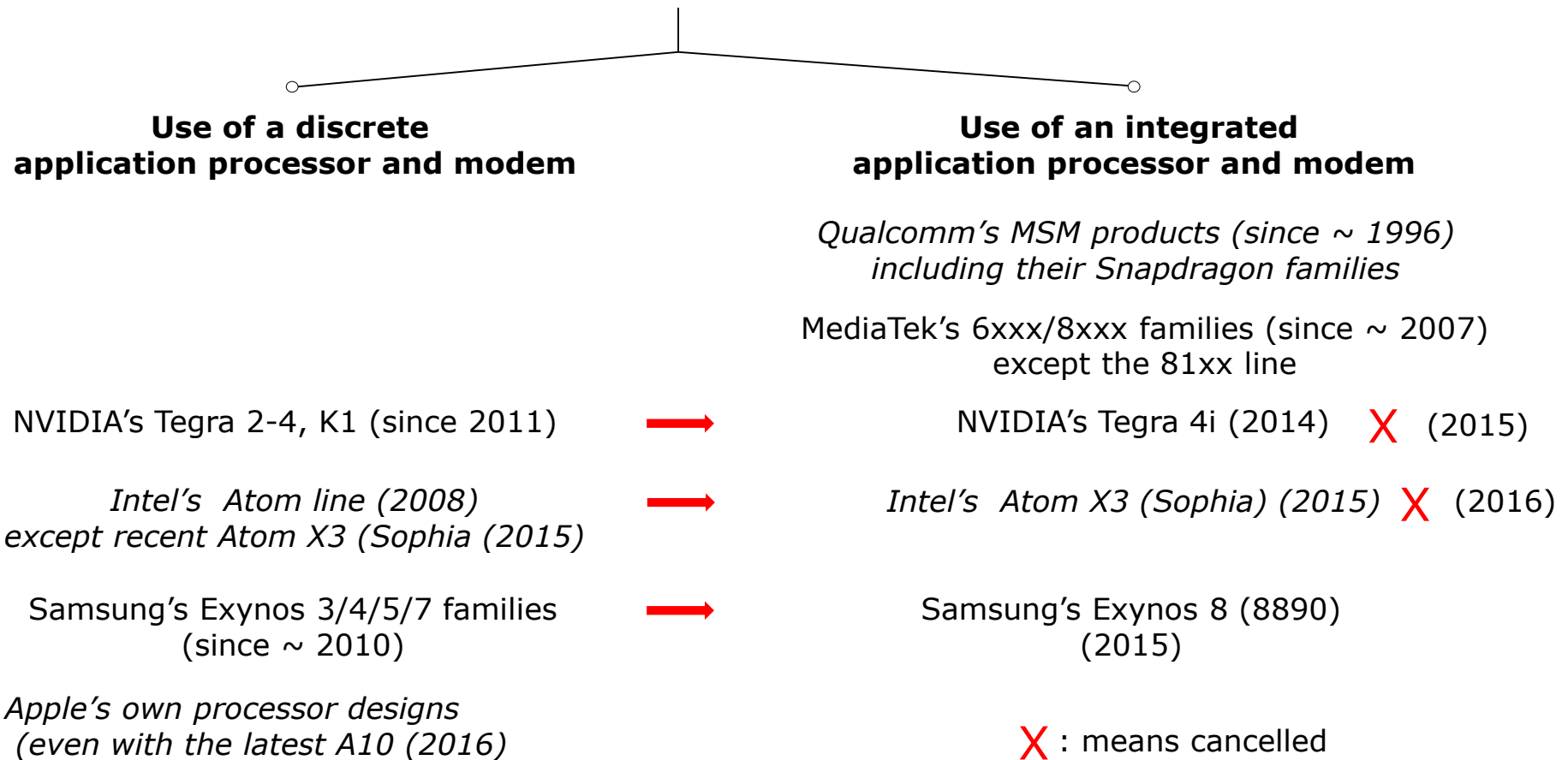
Model	Year	Cores	Techn.	Integrated modem
Samsung Exynos 5 Octa 5420	2013	4x A7 + 4x A15	28 nm	no
Samsung Exynos 5 Octa 5422	2014	4x A7 + 4x A15	28 nm	no
Samsung Exynos 5 Hexa 5260	2014	4x A7 + 2x A15	28 nm	no
Samsung Exynos 5 Octa 5430	2014	4x A7 + 4x A15	20 nm	no
Samsung Exynos 7 Octa 5433	2014	4x A53 + 4x A57	20 nm	no
Samsung Exynos 7 Octa 7420	2015	4x A53 + 4x A57	14 nm	no
Samsung Exynos 8 Octa 8890	2015	4x A53 + 4x M1	14 nm	yes
Qualcomm Snapdragon S 808	2014	4x A53 + 2x A57	20 nm	no
Qualcomm Snapdragon S 810	2015	4x A53 + 4x A57	20 nm	no
Qualcomm Snapdragon S 820	2016	2x Kryo 1.7 GHz + 2x Kryo 2.2 GHz	14 nm FnFET	no
MediaTek MT8135	2013	2x A7 + 2x A15	28 nm	no
MediaTek MT6595	2014	4x A7 + 4x A17	28 nm	yes
MediaTek MT6797	2015	8x A53+ 2x A57	20 nm	yes
Renesas MP 6530	2013	2x A7 + 2x A15	28 nm	yes
Allwinner UltraOcta A80	2014	4x A7 + 4x A15	28 nm	no

5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (6)

Integration of the application processor and the modem

- **Integrating the modem** into the chip results in **less costs** and **shorter time to market**.
- **Qualcomm pioneered** this move by designing integrated parts already about 1996.

Integration of the application processor and the modem



5.7.1 The Exynos 9 Series 9810 – Overview (2)

Main features of the Exynos 9810 vs. the Exynos 8995 [68]

Samsung Exynos SoCs Specifications		
SoC	Exynos 9810	Exynos 8895
CPU	4x Exynos M3 @ 2.9 GHz 4x 512KB L2 ?? 4x Cortex A55 @ 1.9 GHz 4x 128KB L2 4096KB L3 DSU ??	4x Exynos M2 @ 2.314 GHz 2048KB L2 4x Cortex A53 @ 1.690GHz 512KB L2
GPU	Mali G72MP18	Mali G71MP20 @ 546MHz
Memory Controller	4x 16-bit CH LPDDR4x @ 1794MHz	4x 16-bit CH LPDDR4x @ 1794MHz
Media	10bit 4K120 encode & decode H.265/HEVC, H.264, VP9	28.7GB/s B/W 4K120 encode & decode H.265/HEVC, H.264, VP9
Modem	Shannon Integrated LTE (Category 18/13) DL = 1200 Mbps 6x20MHz CA, 256-QAM UL = 200 Mbps 2x20MHz CA, 256-QAM	Shannon 355 Integrated LTE (Category 16/13) DL = 1050 Mbps 5x20MHz CA, 256-QAM UL = 150 Mbps 2x20MHz CA, 64-QAM
ISP	Rear: 24MP Front: 24MP Dual: 16MP+16MP	Rear: 28MP Front: 28MP
Mfc. Process	Samsung 10nm LPP	Samsung 10nm LPE

5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (7)

Main features of Samsung's octa core big.LITTLE SOCs supporting GTS

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (Exynos 5420)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (Exynos 5422)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (Exynos 5800)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (Exynos 5430)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (Exynos 5433)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
Exynos 7 Octa (Exynos 7420)	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
Exynos 7 Octa (Exynos 7885)	14 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
Exynos 8 Octa (Exynos 8890)	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
Exynos 9 Series (Exynos 8895)	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
Exynos 9 Series (Exynos 9810)	10 nm FinFET		Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus

OS support for GTS

- big.LITTLE technology needs suitable OS support for scheduling tasks to the right computing resources to achieve the least possible power consumption.
- ARM and Linaro jointly develop OS support for GTS, these become available first as Linux or Android patch sets, later also they will be included into the mainstream Linux or - Android kernels.
- As an example, ARM/Linaro's IPA (Intelligent Power Management) became first available as a Linaro patch set in 09/2014 and then it was included into Linux 4.10 in 8/2015.
- It is stated that "software represents the Achilles' heel of the technology and severely limits its potential [57].
- In the Chapter on big-LITTLE processing we give an overview of the OS support of GTS.

Remark

Linaro is a non-profit foundation of interested firms to foster open source Linux packages that are optimized for ARM architectures.

5.1 Octa core big.LITTLE mobile SOCs supporting GTS - Overview (9)

Overview of OS kernels supporting GTS (announced or used)

ARM/Linaro	ARM big.LITTLE MP (Global Task Scheduling) (~06/2013)	ARM IPA (Intelligent Power Allocation) (on Exynos Octa models) (10/2014)	ARM/Linaro EAS (Energy Aware Scheduling) (on Google Pixel Phone), (10/2016)
MediaTek	MediaTek CorePilot 1.0 (on MT8135) (07/2013)	MediaTek CorePilot 2.0 (on Helio X10 (MT6595)) (03/2015)	MediaTek CorePilot 3.0 (on Helio X20 (MT6797)) (05/2015)
			MediaTek CorePilot 4.0 (on Helio X30 (MT6799)) (02/2017)
Qualcomm	Qualcomm's Energy Aware Scheduling (on Snapdragon 610/615) (02/2014))	Qualcomm Symphony System Manager (on Snapdragon 820) (11/2015)	
Samsung	Samsung's big.LITTLE HMP (~ARM's big.LITTLE MP) (on Exynos 5 models) (09/2013)		
	2013	2014	2015
			2016
			2017

5.2: The world's first octa core big.LITTLE mobile SOC
supporting GTS: Samsung's Exynos 5 Octa 5420 (2013)

5.2 The Exynos 5 Octa 5420 (1)

5.2: The world's first octa core big.LITTLE mobile SOC supporting GTS: Samsung's Exynos 5 Octa 5420 (2013) [2]

- It is a 32-bit ARMv7 mobile processor.
- Announced in 03/2013, launched in Galaxy S4 models in 4/2013.
- Task scheduling supports GTS, called HMP (Heterogeneous Multi-Processing) by Samsung.

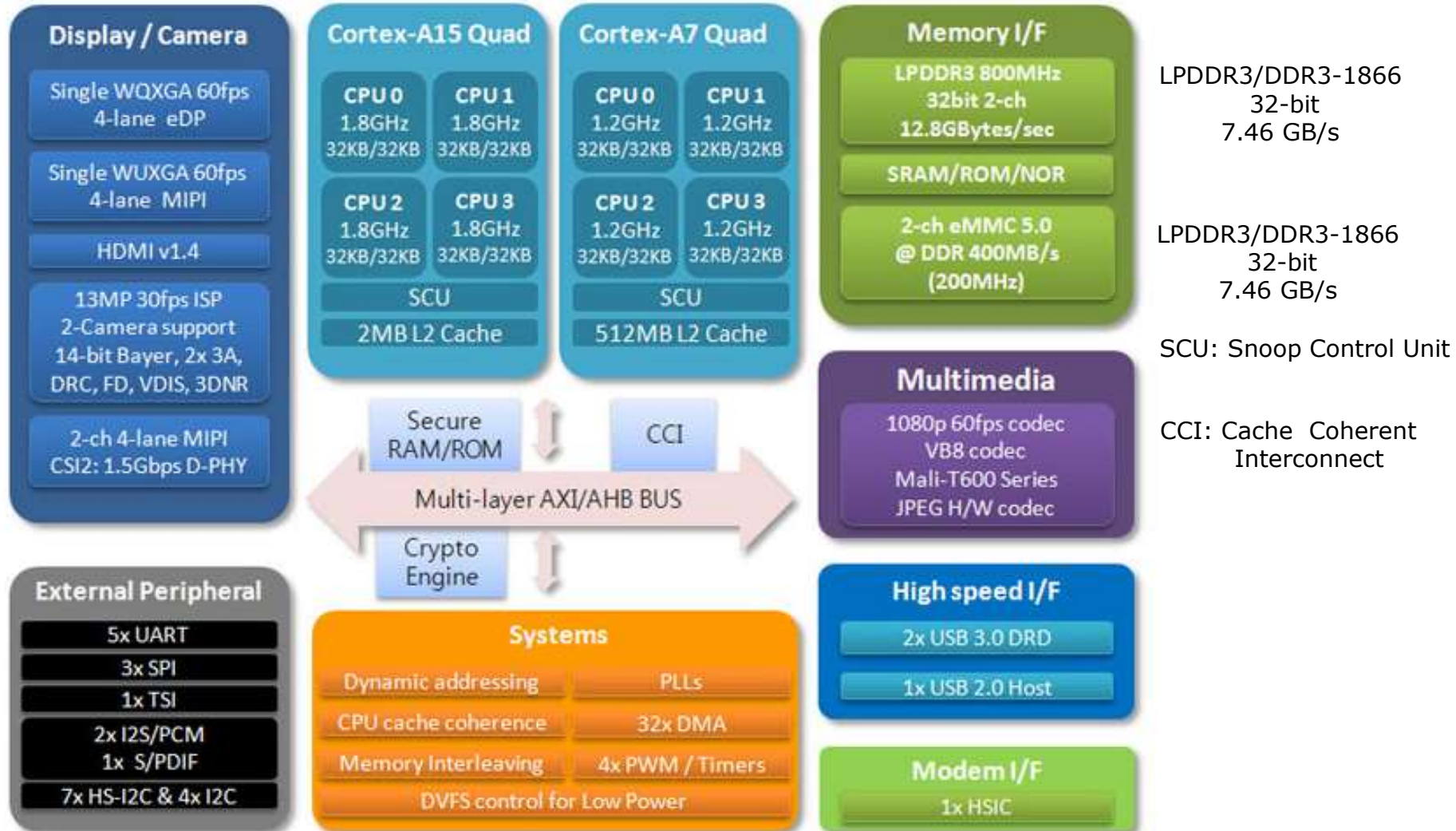
5.2 The Exynos 5 Octa 5420 (2)

Main features of Samsung's Exynos 5 Octa 5420 (2013)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (<i>Exynos 5420</i>)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (<i>Exynos 5422</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (<i>Exynos 5800</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (<i>Exynos 5430</i>)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (<i>Exynos 5433</i>)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
<i>Exynos 7 Octa (Exynos 7420)</i>	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
<i>Exynos 7 Octa (Exynos 7885)</i>	14 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
<i>Exynos 8 Octa (Exynos 8890)</i>	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
<i>Exynos 9 Series (Exynos 8895)</i>	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
<i>Exynos 9 Series (Exynos 9810)</i>	10 nm FinFET		Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus

5.2 The Exynos 5 Octa 5420 (3)

Block diagram of Samsung's Exynos 5 Octa 5420 [7]



5.2 The Exynos 5 Octa 5420 (4)

Contrasting GTS with exclusive cluster switching (Based on [8])

Cluster allocation

Linux Scheduler picks any **One Cluster** at a time,
But, no combinations

High Cluster is picked if at-least one High Core is needed,
else **Low Cluster**

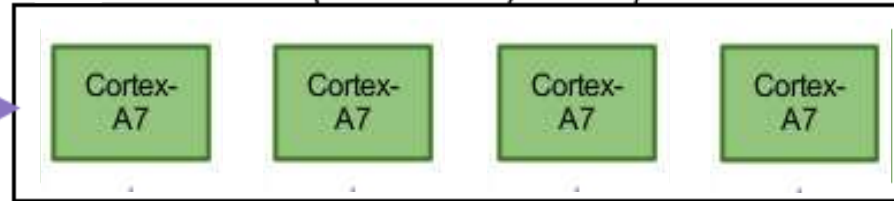
Either

A15 **High Cluster** (High in Perf, Power) of 4 Cores



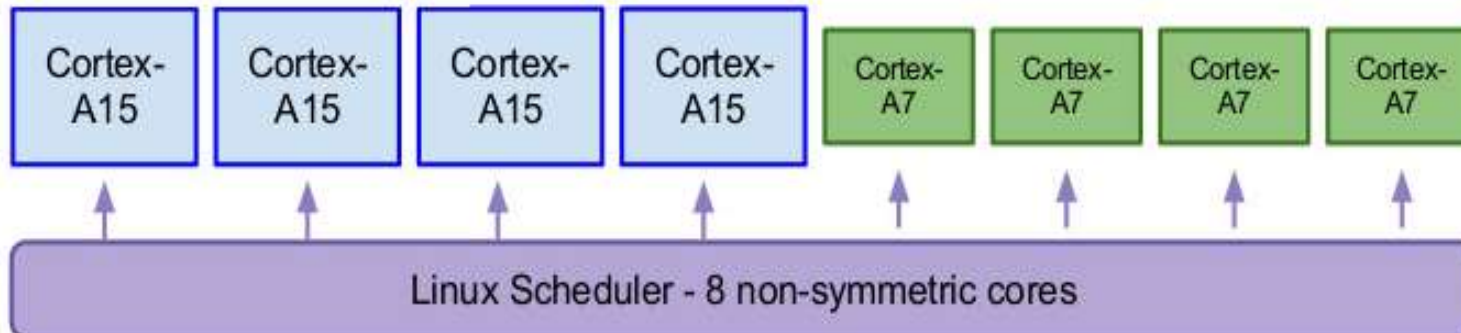
Or

A7 **Low Cluster** (Low in Perf, Power) of 4 Cores



Exynos 5 Octa 5410

Heterogeneous Multi-Processing (HMP)



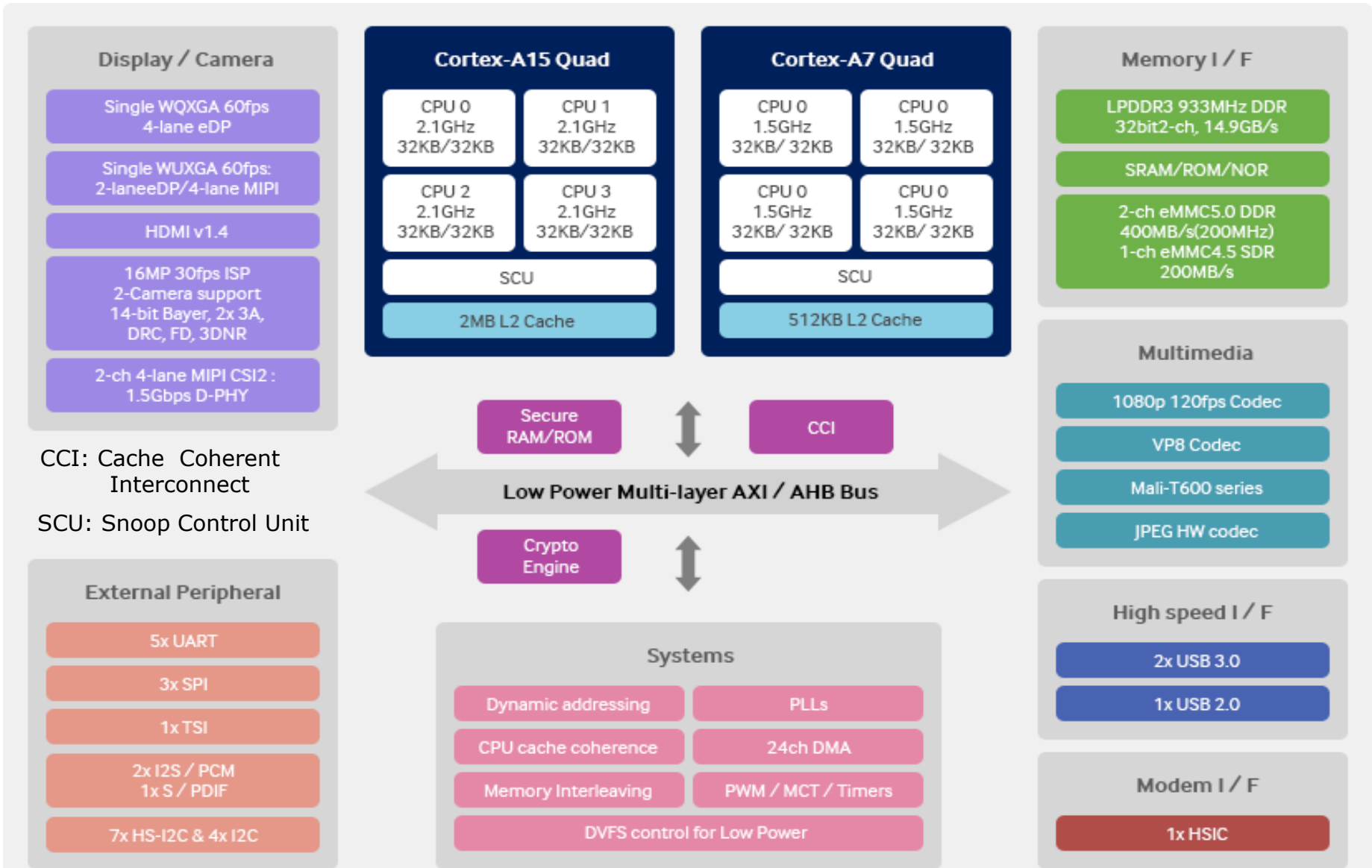
Exynos 5 Octa 5420

Remark [9]

- The 5420 was followed by the 5422, it is Samsung's second octa core big.LITTLE processor operating under GTS, announced in 2/2014, available in Q1 2014.
- It is basically a faster variant of the 5420, as seen in the next Figure.

5.2 The Exynos 5 Octa 5420 (6)

Block diagram of the Samsung Exynos 5 Octa 5422 [10]



5.2 The Exynos 5 Octa 5420 (7)

Samsung's subsequent big.LITTLE models supporting GTS

- In 8/2014 Samsung announced their first 20 nm octa core big.LITTLE processor, **the Exynos 5 Octa 5430** with **4 Cortex-A15 and 4 Cortex-A7 cores**.

Due to the new low-power High-K Metal Gate (HKMG) process technology **power consumption of this processor could be lowered by 28 %** compared to the previous 28 nm technology [11].

- About the same time Samsung unveiled also the **Exynos 5 Octa 5433**, that included **4 Cortex-A57 and 4 Cortex-A53 64-bit cores but runs in 32-bit mode** (called AArch32 mode).

The Exynos 5 Octa 5433 incorporates the Mali T760 GPU that is claimed to offer 76 % more performance than the previous Mali T628

- Later (**in 10/2014**) Samsung introduced the **Exynos 7 Octa 7420** that included **the same cores as the Exynos 5 Octa 5433** but runs already **in 64-bit mode** (AArch64).

Samsung announced about 57 % performance increase over the Exynos 5 Octa 5433 implementation [12].

- Subsequently, the Examples 2 and 3 give some more details about the Exynos 5 Octa 5433 and the Exynos 7 Octa 7420.

5.3: Samsung's first 64-bit octa core big.LITTLE mobile processor supporting GTS and operating in the ARMv8 Aarch32 mode: the Exynos 5 Octa 5433 (2014)

5.3: Samsung's first 64-bit octa core big.LITTLE SOC supporting GTS and operating in the ARMv8 Aarch32 mode: the Exynos 5 Octa 5433 (2014)

- The Exynos 5 Octa 5433 is Samsung's **first mobile** processor **built up of ARMv8 cores**, but **it operates in the AArch32 execution mode** [13].
This is the reason why the model designation starts with 5 instead of 7.
- Nevertheless, **it takes advantage in the architectural improvements of the AArchv8 cores** (actually the Cortex-A57 and Cortex-A53 cores), as indicated next in the performance ranking of this processor.

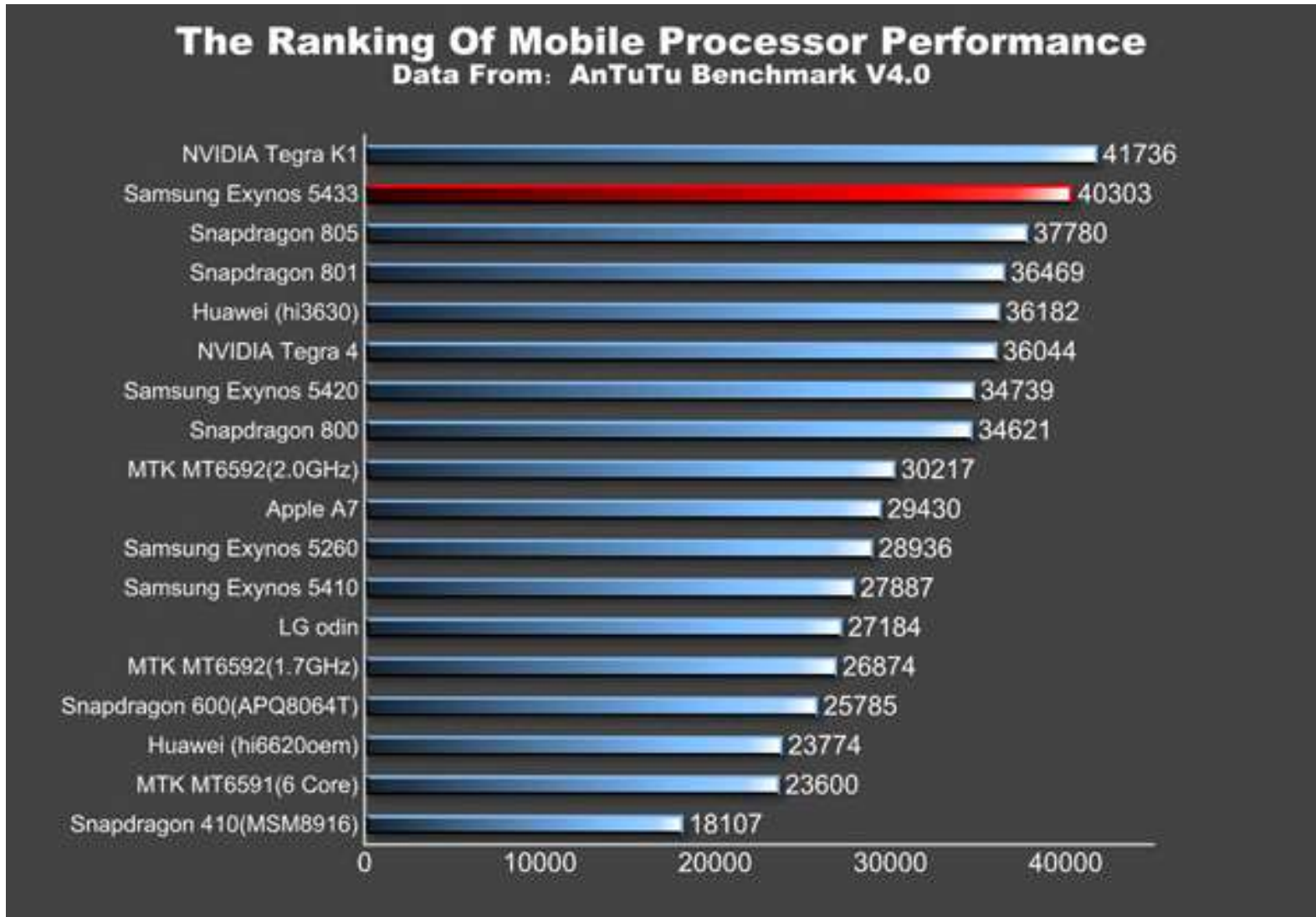
5.3 The Exynos 5 Octa 5433 (2)

Main features of Samsung's Exynos 5 Octa 5433 (2014)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (<i>Exynos 5420</i>)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (<i>Exynos 5422</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (<i>Exynos 5800</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (<i>Exynos 5430</i>)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (<i>Exynos 5433</i>)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
<i>Exynos 7 Octa (Exynos 7420)</i>	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
<i>Exynos 7 Octa (Exynos 7885)</i>	148 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
<i>Exynos 8 Octa (Exynos 8890)</i>	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
<i>Exynos 9 Series (Exynos 8895)</i>	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
<i>Exynos 9 Series (Exynos 9810)</i>	10 nm FinFET	Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus	

5.3 The Exynos 5 Octa 5433 (4)

Performance ranking of the Exynos 7 Octa 5433 based on the AnTuTu v4.0 benchmark [14]



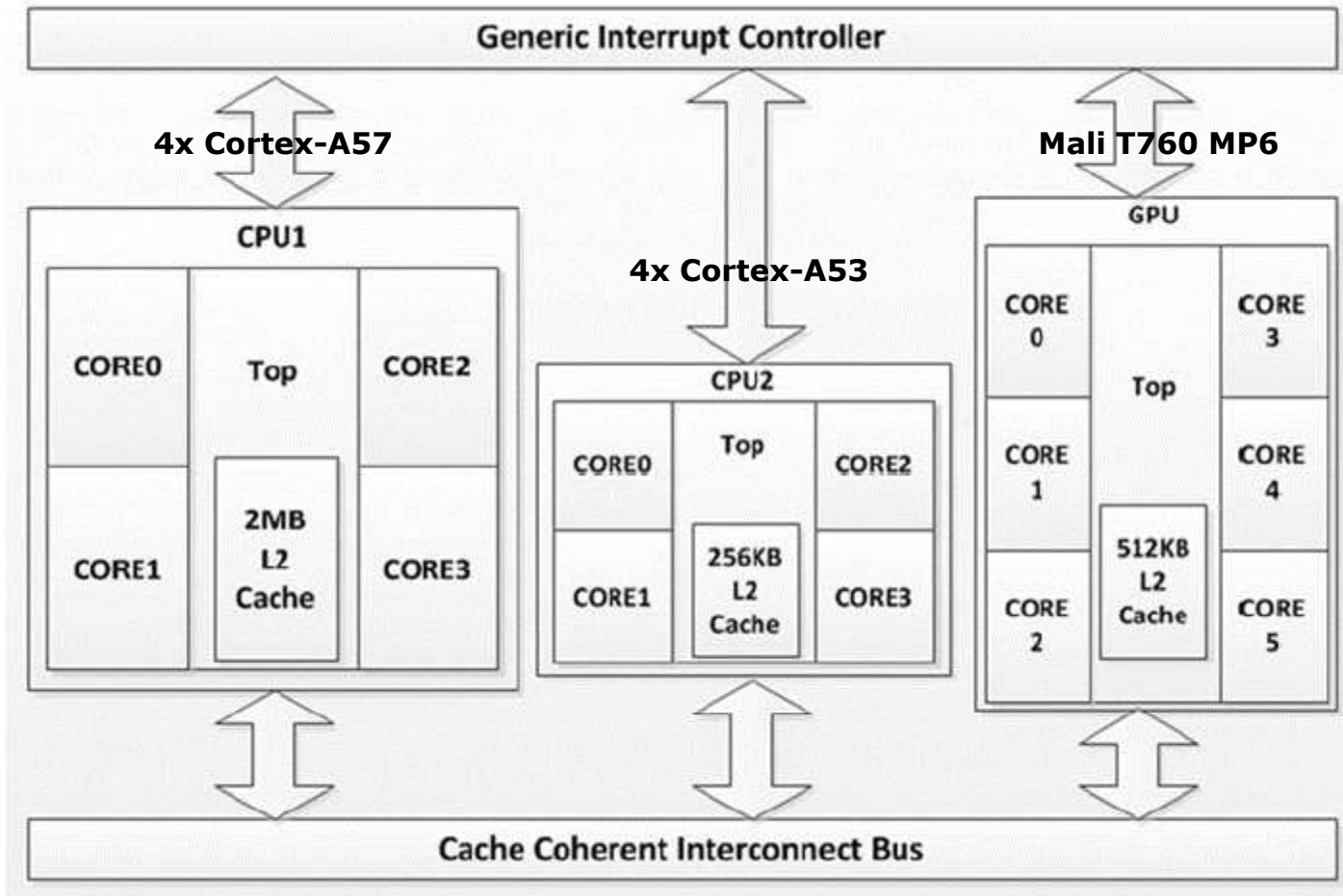
5.3 The Exynos 5 Octa 5433 (5)

Main features of Samsung's Exynos 5 Octa SOC [13]

Samsung Exynos 5 Octa 2014 lineup			
SoC	Samsung Exynos 5422	Samsung Exynos 5430	Samsung Exynos 5433
CPU	4x Cortex A7 r0p5 @ 1.3GHz	4x Cortex A7 r0p5 @ 1.3GHz	4x Cortex A53 @ 1.3GHz
	4x Cortex A15 r2p4 @ 1.9GHz	4x Cortex A15 r3p3 @ 1.8GHz	4x Cortex A57 r1p0 @ 1.9GHz
Memory Controller	2x 32-bit @ 933MHz	2x 32-bit @ 1066MHz	2x 32-bit @ 825MHz
	14.9GB/s b/w	17.0GB/s b/w	13.2GB/s b/w
GPU	Mali T628MP6 @ 533MHz	Mali T628MP6 @ 600MHz	Mali T760MP6 @ 700MHz
Mfc. Process	Samsung 28nm HKMG	Samsung 20nm HKMG	Samsung 20nm HKMG

5.3 The Exynos 5 Octa 5433 (6)

Block diagram of Samsung's Exynos 5 Octa 5433 mobile processor [15]



5.4: Samsung's first 64-bit octa core big.LITTLE SOC operating in the ARMv8 AArch64 mode: the Exynos 7 Octa 7420 (2015)

- 5.4.1 The Exynos 7 Octa 7420 - Overview
- 5.4.2 Introducing binning in form of ASV groups
- 5.4.3 Introducing AVS, called ASV
(Adaptive Scaling Voltage) by Samsung
- 5.4.4 Introducing LPDDR4
- 5.4.5 Implementing a hardware memory compressor

5.4.1: The Exynos 7 Octa 7420 - Overview

5.4.1: The Exynos 7 Octa 7420 - Overview

- The **Exynos 7 Octa 7420** is the world's first 64-bit octa core big.LITTLE SOC operating in the ARMv8 AArch64 mode-
- It is the world's first application processor built on 14 nm FinFET.
- It is the core part of the Samsung Galaxy S6.
- The Exynos 7 Octa 7420 is the 14 nm shrink of the Exynos 5 5433 with major enhancements, such as **ASV (Adaptive Scaling Voltage)**.

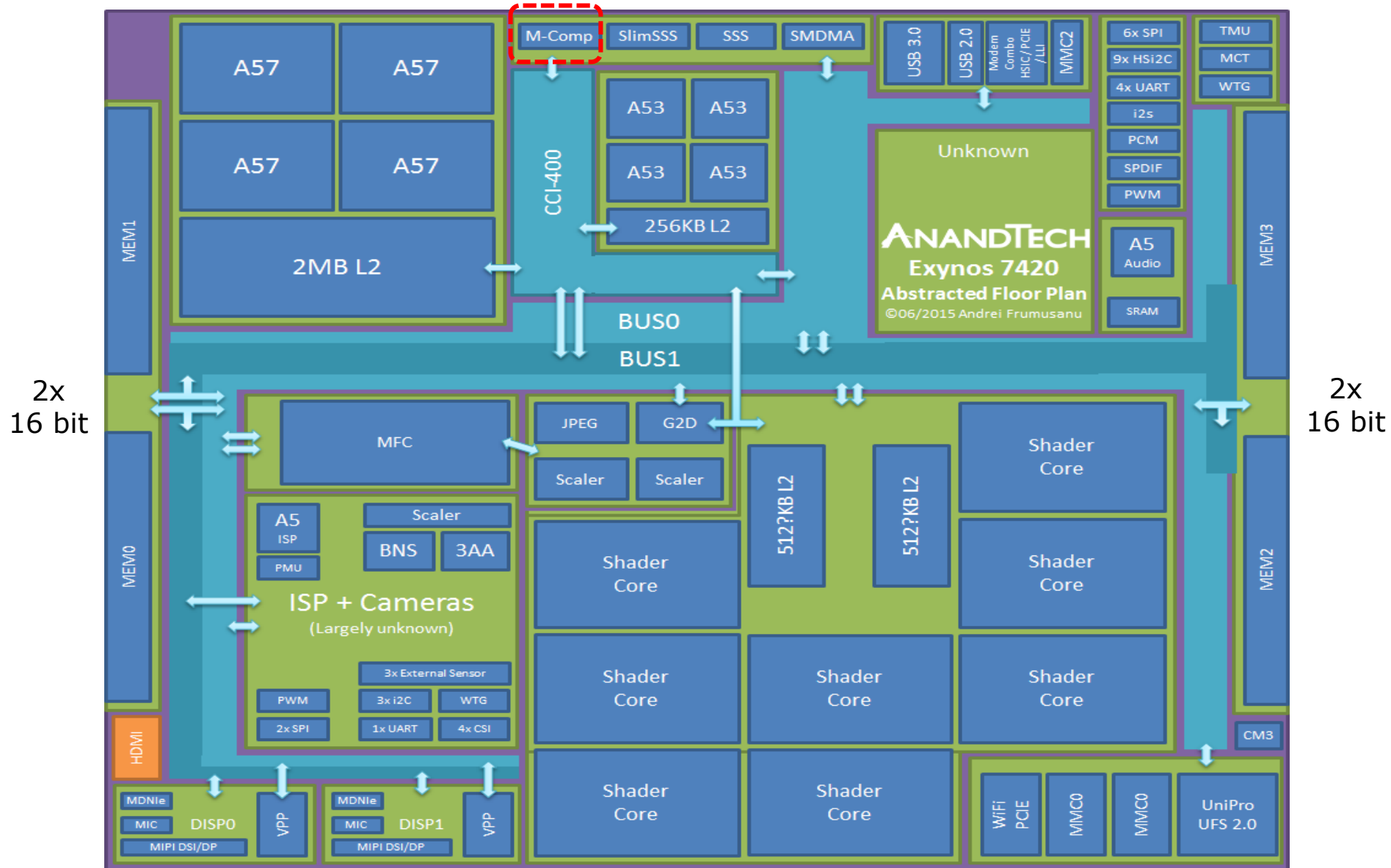
5.4.1 The Exynos 7 Octa 7420 - Overview (2)

Main features of Samsung's Exynos 7 Octa 7420 (2015)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (<i>Exynos 5420</i>)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (<i>Exynos 5422</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (<i>Exynos 5800</i>)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (<i>Exynos 5430</i>)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (<i>Exynos 5433</i>)	20 nm HKMG		Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
<i>Exynos 7 Octa (Exynos 7420)</i>	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
<i>Exynos 7 Octa (Exynos 7885)</i>	14 nm HKMG	ARM v8-A	Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
<i>Exynos 8 Octa (Exynos 8890)</i>	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
<i>Exynos 9 Series (Exynos 8895)</i>	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
<i>Exynos 9 Series (Exynos 9810)</i>	10 nm FinFET		Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus

5.4.1 The Exynos 7 Octa 7420 - Overview (3)

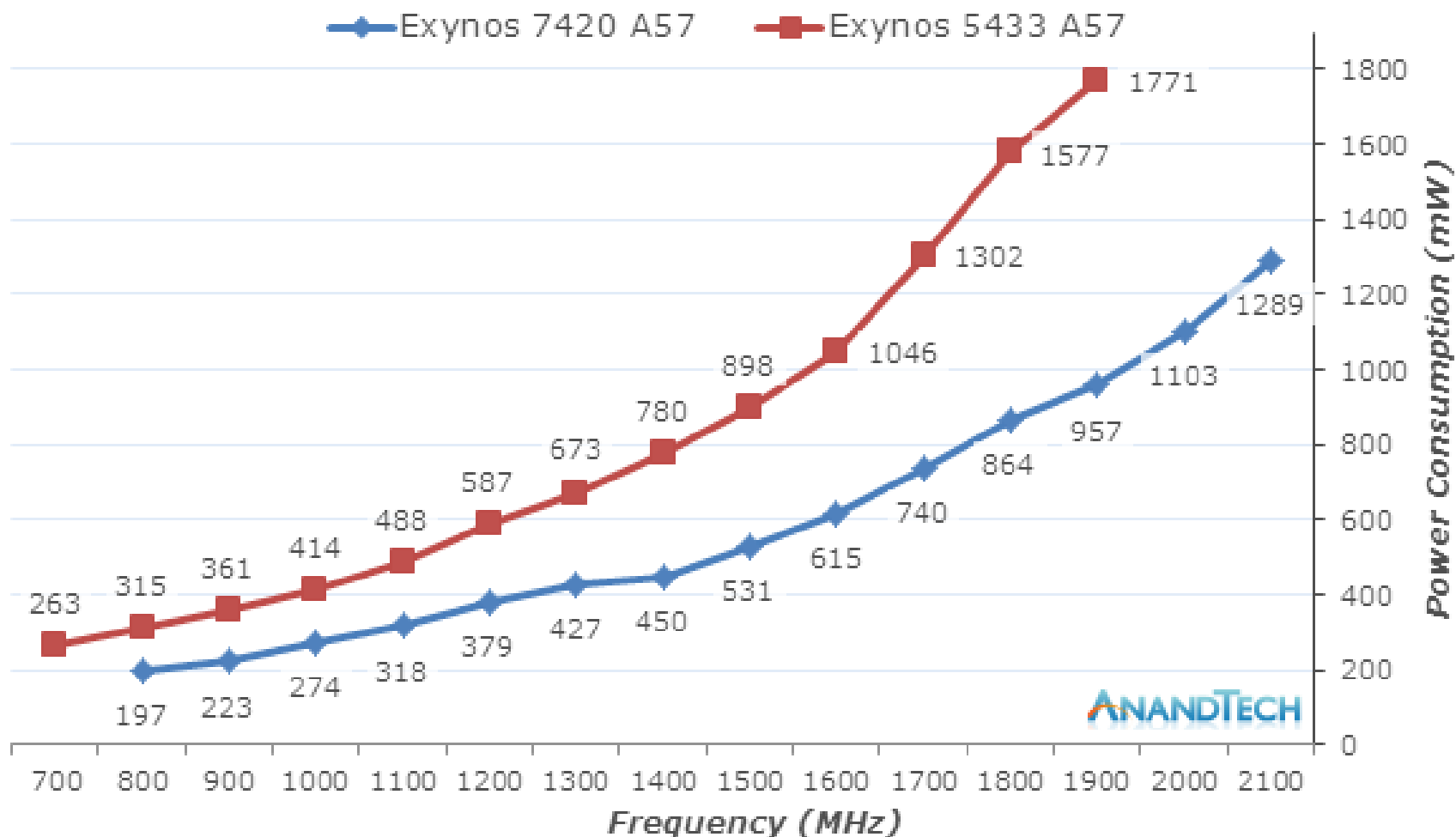
Assumed block diagram of Samsung's Exynos 7 Octa 7420 processor [16]



5.4.1 The Exynos 7 Octa 7420 - Overview (4)

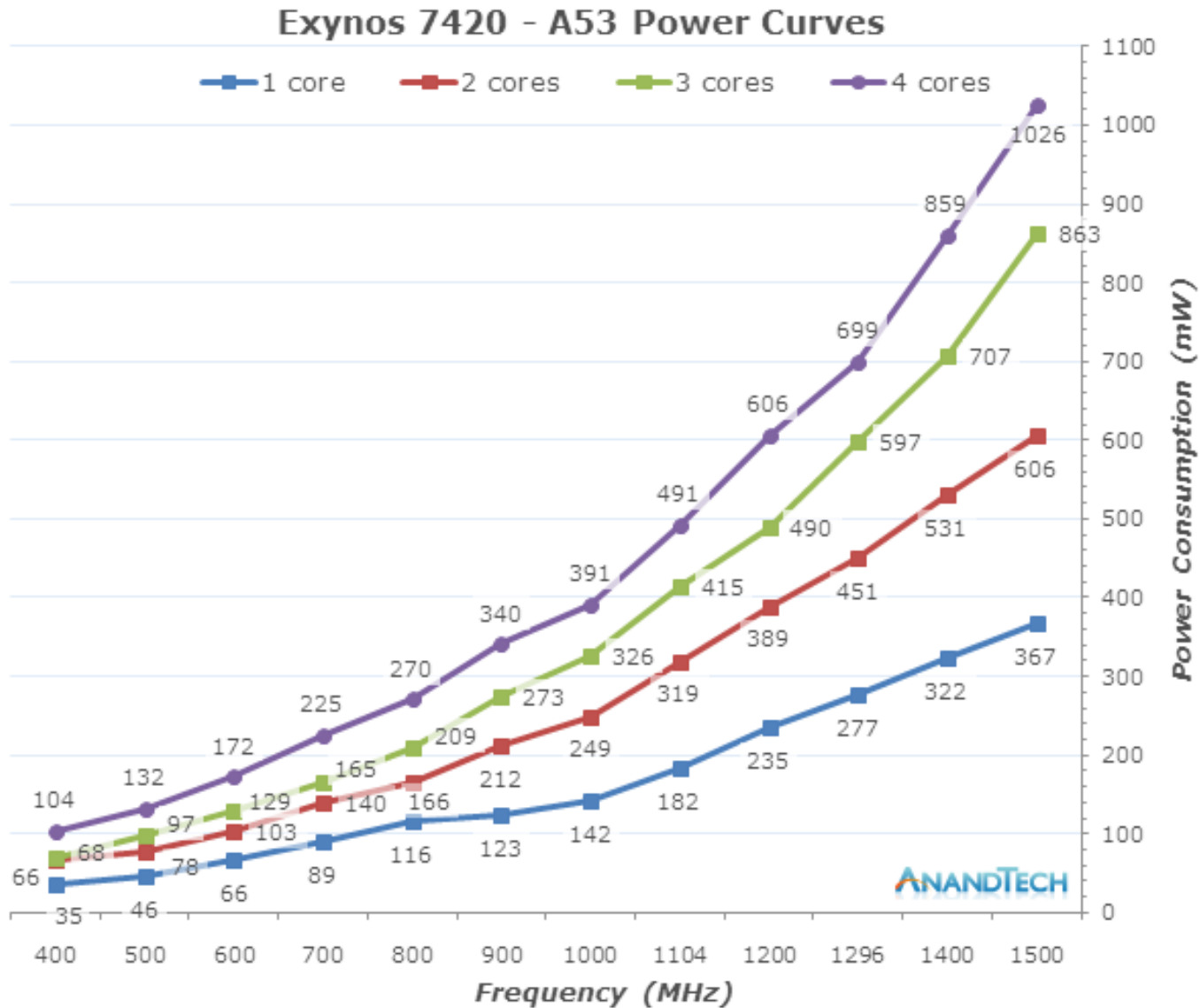
A57 power curves of the Exynos 7 7420 vs. the Exynos 5 5433 [16]

Core Average Maximum Power Consumption



5.4.1 The Exynos 7 Octa 7420 - Overview (5)

A53 power curves of the Exynos 7 7420 [16]



Main innovations of the Samsung Exynos 7 7420

- a) Introducing binning in form of ASV groups
- b) Introducing AVS, called ASV (Adaptive Scaling Voltage) by Samsung
- c) Using LPDDR4 memory technology
- d) Implementing a hardware memory compressor

5.4.2 Introducing binning in form of ASV groups

5.4.2 Introducing binning in form of ASV groups (1)

5.4.2 Introducing binning in form of ASV groups

Before discussing Samsung's approach for binning let's recall **traditional binning**.

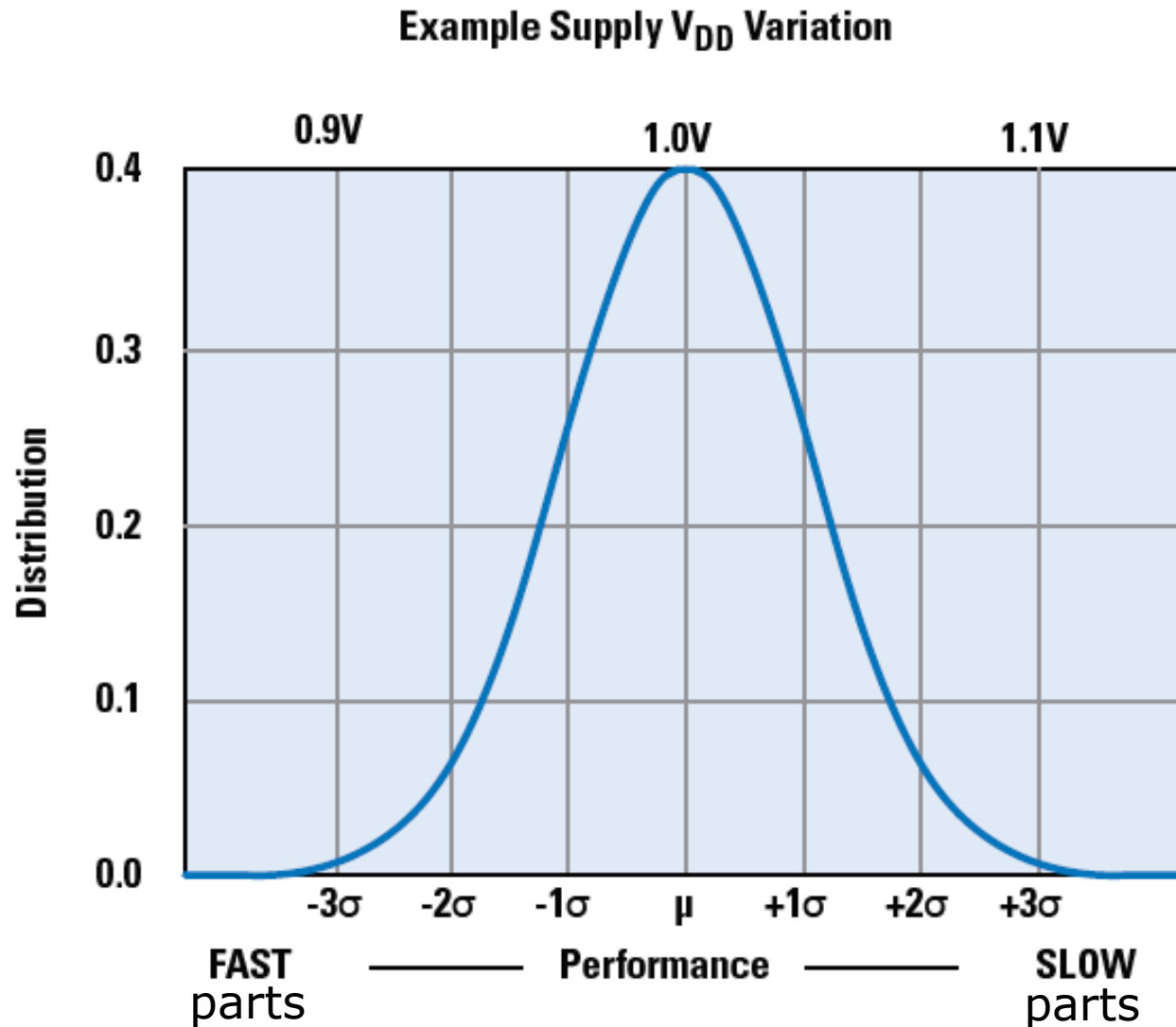
5.4.2 Introducing binning in form of ASV groups (2)

Traditional binning of processors-1

Electrical parameters of fabricated processor chips vary in a wide range, as illustrated for the distribution of minimum core voltages needed to sustain a given clock frequency, as measured post fabrication on the chips.

5.4.2 Introducing binning in form of ASV groups (3)

Example: Distribution of the minimum core voltage needed to sustain a given clock frequency measured on fabricated chips [17]



5.4.2 Introducing binning in form of ASV groups (4)

Traditional binning of processors-2

- The distribution of electrical parameters on the fabricated chips will traditionally be addressed by the manufacturers by testing all chips at the factory and classifying them into a few number of groups, often called bins.
- These groups are considered then as different models of a processor line (termed also as SKUs (Stock Keeping Units)) with given sets of electrical parameters, first of all with different max. clock frequencies and will be sold typically at different sales prices.
- As an example, the next Table shows different models (SKUs) of a given processor line.

5.4.2 Introducing binning in form of ASV groups (5)

Frequency bins of a given model (Intel's Core 2 Duo (2006))

Product Name	Intel Core2 Duo E6400	Intel Core2 Duo E6300	Intel Core2 Duo E4300
Code Name	Conroe	Conroe	Conroe
Essentials			
Processor Number	E6400	E6300	E4300
Launch Date	Q3'06	Q3'06	Q3'06
Lithography	65 nm	65 nm	65 nm
Recommended Customer Price	\$128.00	N/A	\$106.00
Performance			
# of Cores	2	2	2
Base Frequency	2.13 GHz	1.86 GHz	1.80 GHz
Cache	2 MB L2	2 MB L2	2 MB L2
Bus Speed	1066 MHz FSB	1066 MHz FSB	800 MHz FSB
TDP	65 W	65 W	65 W
VID Voltage Range	0.8500V-1.5V	0.8500V-1.5V	0.8500V-1.5V

5.4.2 Introducing binning in form of ASV groups (6)

Samsung's approach to meet variations of electrical parameters of chips [16]

- In the traditional way of binning fabricated chips are classified according to their max. clock frequency into different groups and each group is sold as a different model of the same line, by contrast Samsung also tests their chips post manufacturing and assigns each chip to a group with similar characteristics, called an **ASV group**, but Samsung sells their chips of a given design only as a single model while marking each chip permanently with an ASV group identifier.
- For the Exynos Octa 7 7420 Samsung marks their chips with the ASV group identifiers **ASV0 to ASV 15** by burning them into on-chip fuses.

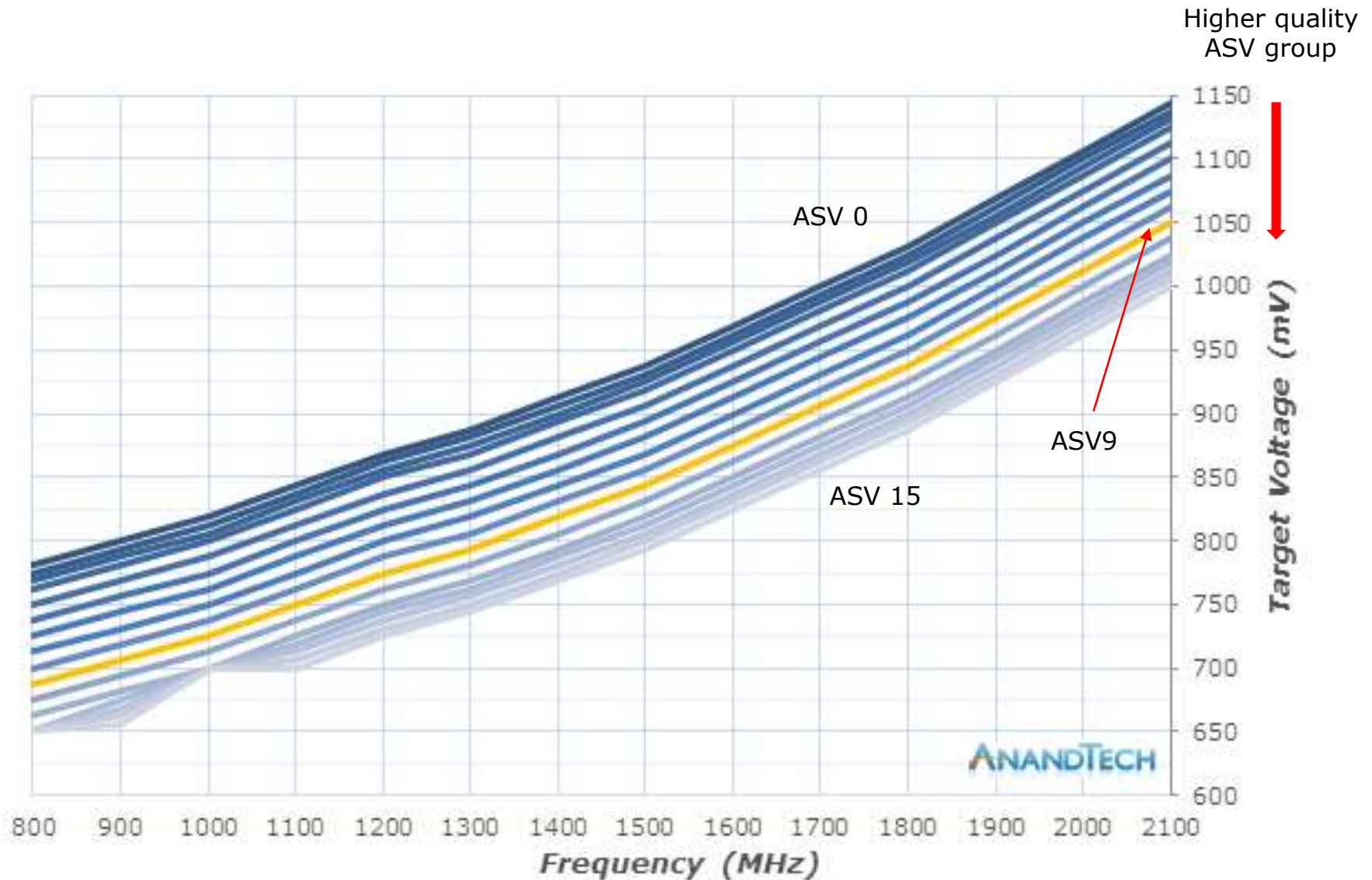
A lower ASV value identifies a worse quality bin whereas a higher one a better quality bin.

Accordingly, **ASV0 is the worst and ASV15 the best quality bin** whereas bin 9 represents the median group.

- As an example, the next Figure shows the target voltage vs. core frequency characteristics of the ASV groups.

5.4.2 Introducing binning in form of ASV groups (7)

Core voltage - core frequency curves of the Exynos 7 Octa 7420 [16]



5.4.3 Introducing AVS, called ASV (Adaptive Scaling Voltage) by Samsung

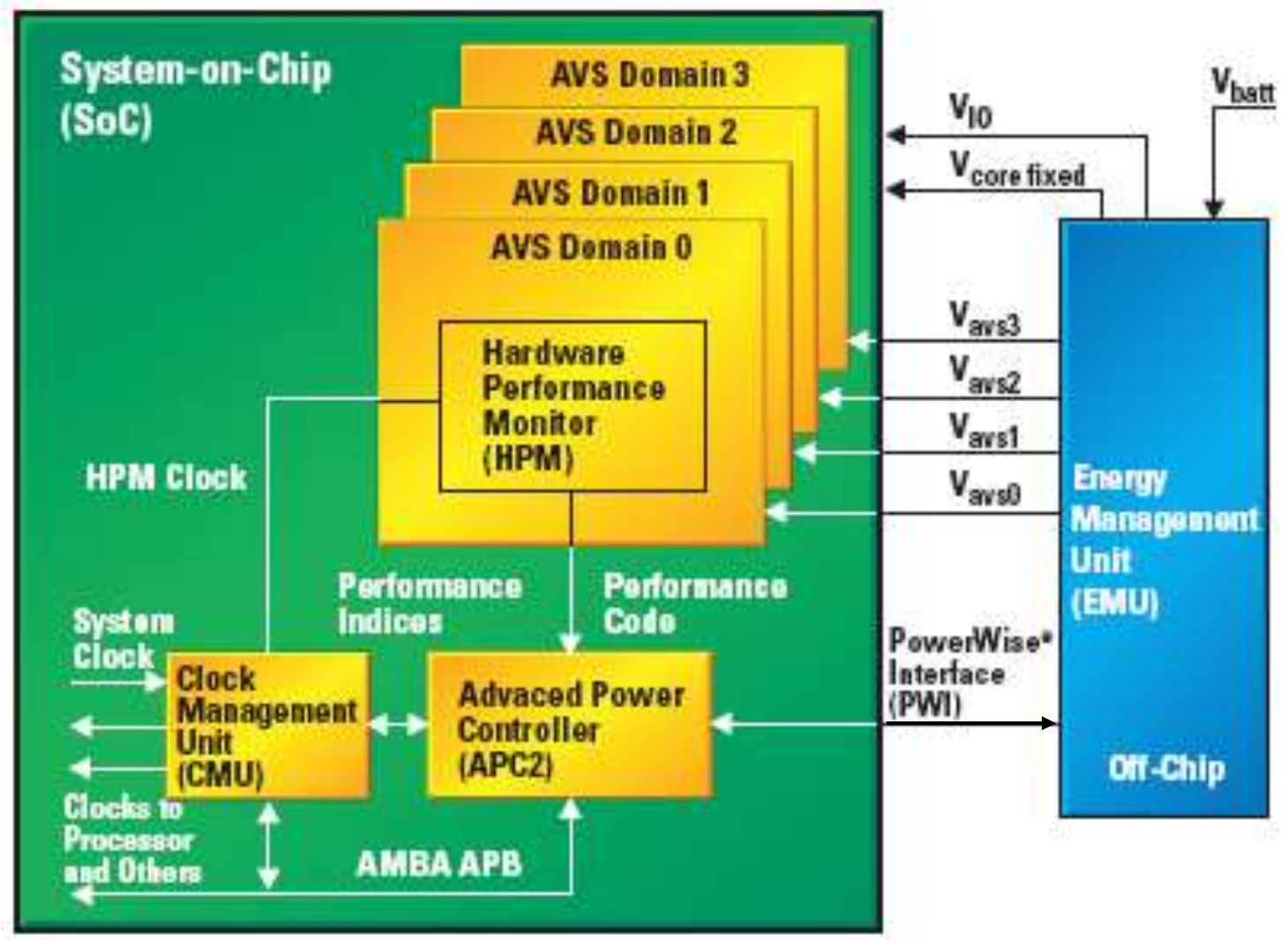
5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (1)

5.4.3 Introducing AVS, called ASV (Adaptive Scaling Voltage) by Samsung

- Samsung's AVS technology is based on licensing National's PowerWise patent that is owned now by Texas Instrument (TI), as TI acquired National Semiconductor in 2011.
- Main components of National's PowerWise technology are seen in the next Figure.

5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (2)

Main components of National's PowerWise technology [16]



5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (3)

Principle of operation of National's PowerWise technology (simplified)-1 [18]

- Based on the current activity of the considered core the OS forwards a **Target performance index** to the **Clock Management Unit (CMU)**.
- The CMU forwards the Target performance index to the **Advanced Power Controller (APC)** and also sets the clock frequency of the **Hardware Performance Monitor (HPM)** to the value corresponding to the Target performance index (this is needed for measuring the actual speed of the core).
- The next step is **voltage adjustment in a closed loop**.
- The HPM measures the propagation delay of the delay line (critical path) and sends a **Performance code (PC)** to the APC.
- The APC compares the received PC with the **Reference Calibration Code (RCC)** that is burnt to on chip fuses and directs the **Energy Management Unit (EMU)** via the **PowerWise Interface (PWI)** accordingly.
- If the PC indicates that the propagation delay is longer than required, APM will let the EMU to increase the **core voltage (V_{avs})** to speed up the core and vice versa.

Nevertheless, depending on whether the gate delays on the chip are too long or too short there are two different avenues to follow subsequently.

Remark

The **RCC** is determined at the factory in a stress test, as the smallest Performance Code (PC) that allows a correct operation of the processor in the given frequency range, and it is burnt to on-chip fuses beyond the ASV identifier.

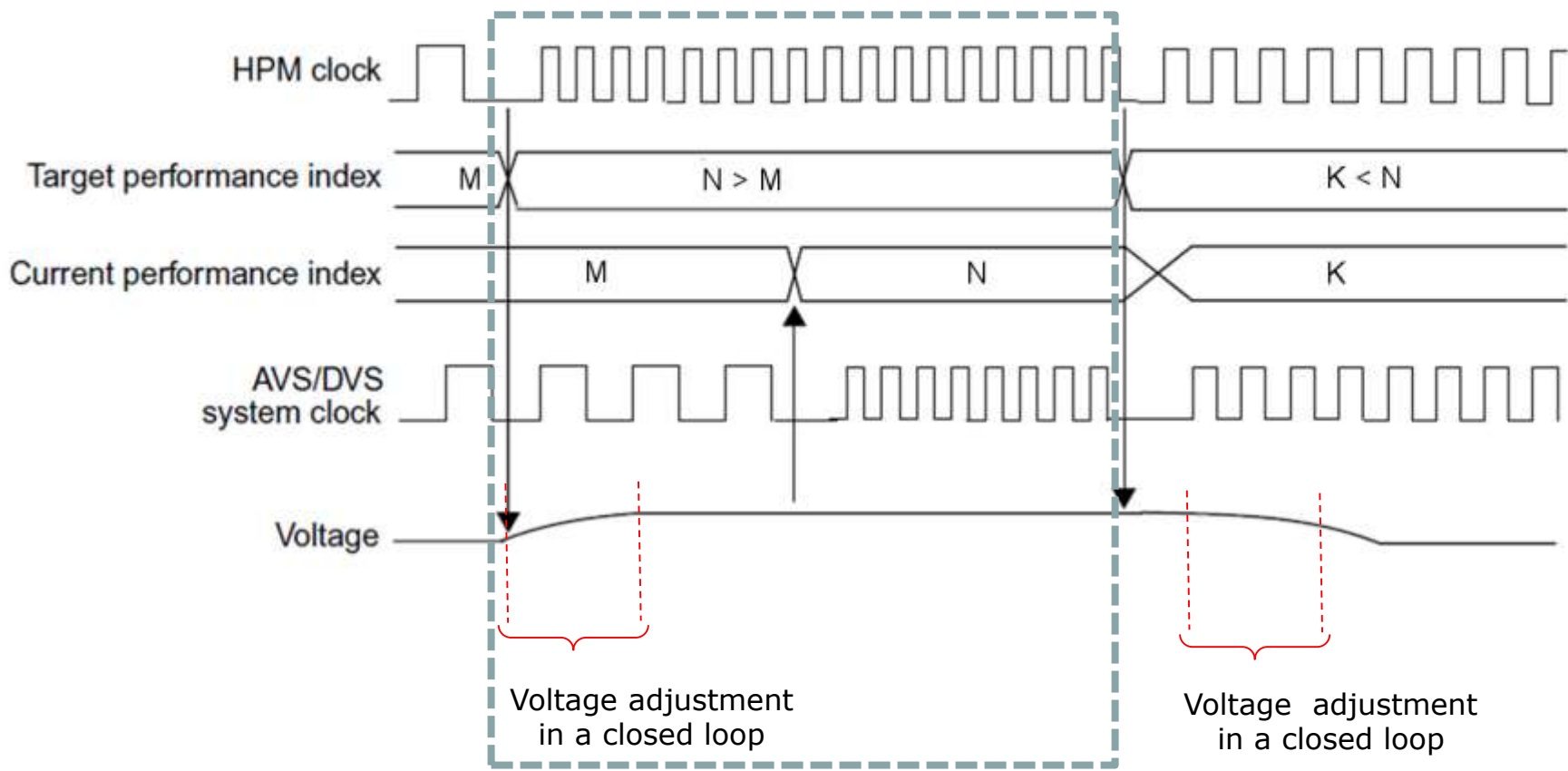
5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (4)

Principle of operation of National's PowerWise technology (simplified)-2 [18]

- When the current performance needs to be increased by raising the current clock frequency, first the supply voltage will be increased adaptively in the closed loop and only subsequently, after the core voltage has already been adjusted will the clock rate be raised.
- To achieve this the APC informs the CMU about finishing the voltage adjustment and then the CMU will change the core clock to the desired value.
- By contrast, when a lower performance is requested than recently existing, the ASP immediately instructs the CMU to reduce the clock frequency to the requested value.
- In parallel the core voltage (V_{avs}) will be adjusted in the closed loop to the appropriate value.

5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (5)

Changing P-states in National's PowerWise technology [18]



5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (6)

Differences between Samsung's and National's AVS implementations

- aa) Implementing AVS in conjunction with ASV binning.
- ab) Use of an ARM M3 microcontroller as the APC unit that communicates with other units by mailbox messages.

5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (7)

aa) Implementing AVS in conjunction with ASV binning

- Selecting the manufactured chips into up to 16 ASV groups has the benefit that the measured Reference Calibration Code (RCC) will fit with a much less tolerance to a particular chip than would in case when the manufactured chips would be selected into a much less number of models.
- This has a further benefit since then the voltage adjusting process becomes shorter.

5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (8)

ab) Use of an ARM M3 microcontroller as the APC unit that communicates with other units by mailbox messages [16]

- Instead of making use of the Advanced Power Controller (APC) unit offered by National as part of the PowerWise technology Samsung implements the APC in form of an **ARM M3 microcontroller**.
- The M3 communicates with the other system components by **mailbox messages**.

Mailboxes represent a kind of **interprocess communication** that is used typically between different architectures such that **each unit can only write messages to its own mailbox** (being in the RAM space) **but is able to read all other mailboxes**.

5.4.3 Introducing AVS called ASV (Adaptive Scaling Voltage) by Samsung (9)

Reducing core voltage and power consumption by using AVS in the Exynos 7 7420 vs. the 5 5433 for different clock rates and ASV groups [19] (Note: Data are not corresponding to the previous figure).

	Exynos 5433	Exynos 7420	Differenz
A57 1,9 GHz bei ASV9	1200,00 mV	975,00 mV	-225,00 mV
A57 1,9 GHz bei ASV15	1125,00 mV	912,50 mV	-212,50 mV
A57 800 MHz bei ASV9	900,00 mV	687,50 mV	-224,50 mV
A57 800 MHz bei ASV15	900,00 mV	625,00 mV	-275,00 mV
A53 1,3 GHz bei ASV9	1112,50 mV	950,00 mV	-162,50 mV
A53 1,3 GHz bei ASV15	1062,50 mV	900,00 mV	-162,50 mV
A53 400 MHz bei ASV9	787,50 mV	656,25 mV	-131,25 mV
A53 400 MHz bei ASV15	750,00 mV	606,25 mV	-143,75 mV
GPU 700 MHz bei ASV9	1050,00 mV	800,00 mV	-250,00 mV
GPU 700 MHz bei ASV15	1012,50 mV	750,00 mV	-262,50 mV
GPU 266 MHz bei ASV9	800,00 mV	668,75 mV	-131,25 mV
GPU 266 MHz bei ASV15	762,50 mV	606,25 mV	-156,25 mV

5.4.4 Introducing LPDDR4

5.4.4 Introducing LPDDR4

Using LPDDR4 almost doubles the memory transfer rate of the 32-bit dual channel memory compared to the LPDDR3's implemented in the Exynos 7 5433 model, actually from 1650 MT/s to 3104 MT/s.

5.4.5 Implementing a hardware memory compressor

5.4.5 Implementing a hardware memory compressor (1)

5.4.5 Implementing a hardware memory compressor [16]

- This is a **hardware unit**, called **M-Comp** on the block diagram of the processor that is designed especially **for Android**.
- We note that **beginning with the Android 4.4 kernel DRAM compression has already become a validated part of the OS and all devices support this feature**.
- Most vendors support it by the **"zram" mechanism**, which is a **ramdisk** with compression support.

The kernel makes use of it **as a swapping device to store rarely used memory pages**.

Also Samsung had implemented this compression mechanism in their Galaxy devices as far back as Android 4.1.

- The **Galaxy S6 implements a more advanced compressor scheme called "zswap"** which is able to **compress memory pages before they need to get swapped out to a swap device**, so it's a more optimized mechanism that sits closer to the kernel's memory management part.

As an example "zswap" may compress 1.21GB of pages into 341MB of physical memory.

This is however **yet a software implementation running on the CPU cores**.

- The available dedicated **hardware compressor (M-Comp)** is **currently not yet activated** and its use needs OS support to be provided in a future OS update.

5.5: Samsung's first SoC including an in-house designed CPU core (the M1): the Exynos 8 Octa 8890 (2016)

- 5.5.1 The Exynos 8 Octa 8890 - Overview
- 5.5.2 The M1 (Mongoose) core

5.5.1 The Exynos 8 Octa 8890 - Overview

5.5.1 The Exynos 8 Octa 8890 - Overview

- It is fabricated based on Samsung's 2. gen. 14 nm (Low-Power Plus (LPP) FinFET process).
- It is the kernel piece of Samsung's Galaxy S7, S7 Edge and Galaxy Note 7.
- Introduced in Q1/2016.

The Exynos 8 Octa 8890 - Overview (2)

Main features of Samsung's Exynos 8 Octa 8890 (2016)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (Exynos 5420)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (Exynos 5422)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (Exynos 5800)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (Exynos 5430)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (Exynos 5433)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
Exynos 7 Octa (Exynos 7420)	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
Exynos 7 Octa (Exynos 7885)	14 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
Exynos 8 Octa (Exynos 8890)	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
Exynos 9 Series (Exynos 8895)	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
Exynos 9 Series (Exynos 9810)	10 nm FinFET		Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus

The Exynos 8 Octa 8890 - Overview (3)

Main innovations of the Exynos 8 Octa 8890 processor

- It is built up as a **big.LITTLE architecture** while as **big cores** Samsung makes use of their first in-house core design, designated as the **M1 (Mongoose) core**.
- It is based on Samsung's custom **SCI (Samsung Coherent Interconnect) bus**.
- It is Samsung's first application processor with an **integrated modem**.

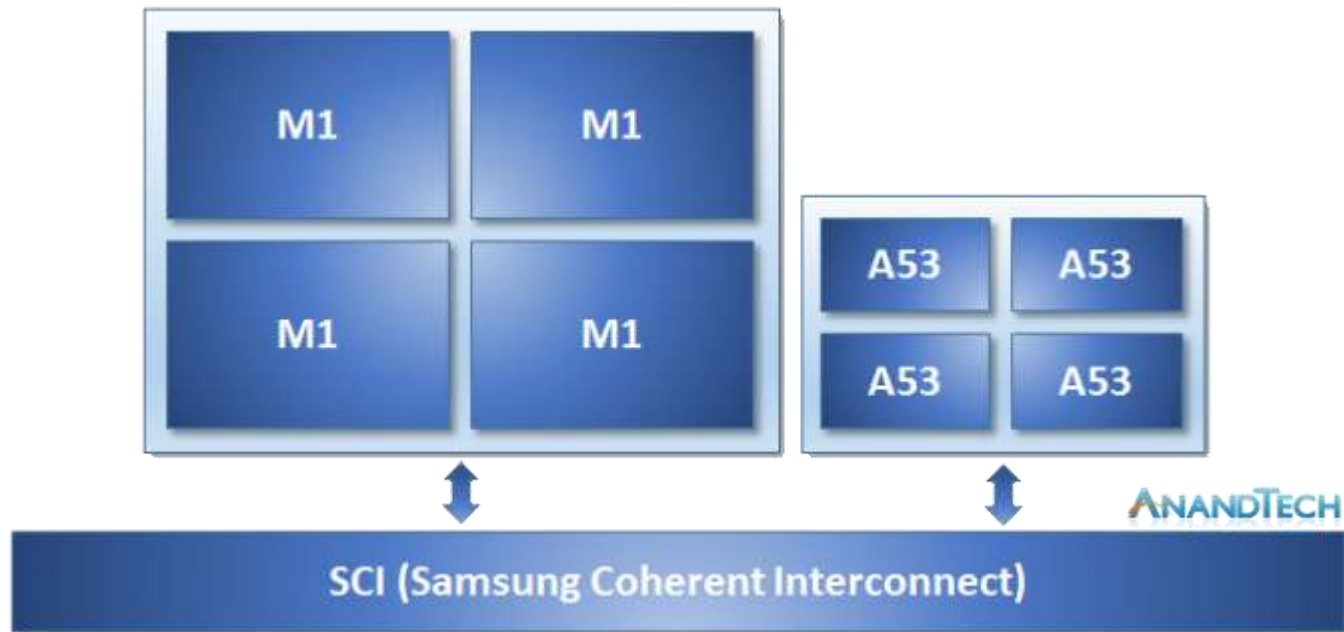


Figure: Basic structure of Samsung's Exynos 8 Octa 8890 [20]

The Exynos 8 Octa 8890 - Overview (4)

Contrasting key features of high-end 14/16 nm mobile processors [21]

High-End SoCs Specifications				
SoC	Qualcomm Snapdragon 820	Huawei Kirin 950	Samsung Exynos 8 Octa 8890	Samsung Exynos 7 Octa 7420
CPU	2x Kryo@1.593GHz 2x Kryo@2.150GHz	4x Cortex A72 @2.3 Ghz 4x Cortex A53 @1.8Ghz	4x A53@1.586GHz 4x Exynos M1 @ 2.60GHz (1-2 core load) 2.29GHz (3-4 core load)	4x A53@1.50GHz 4x A57@2.1GHz
Memory Controller	2x 32-bit LPDDR4 @ 1803 MT/s 28.8GB/s b/w	2x 32-bit LPDDR3 or LPDDR4 @ 1333 MT/s 21.32 GB/s	2x 32-bit LPDDR4 @ 1794 MT/s 28.7GB/s b/w	2x 32-bit LPDDR4 @ 1555 MT/s 24.8GB/s b/w
GPU	Adreno 530 @ 624 MHz	ARM Mali T860 , @ 900 MHz	Mali T880MP12 @ 650 MHz	Mali T770MP8 @ 770 MHz
Mfc. Process	Samsung 14nm LPP	TSMC 16 nm FinFET+	Samsung 14nm LPP	Samsung 14nm LPE

5.5.2 The M1 (Mongoose) core

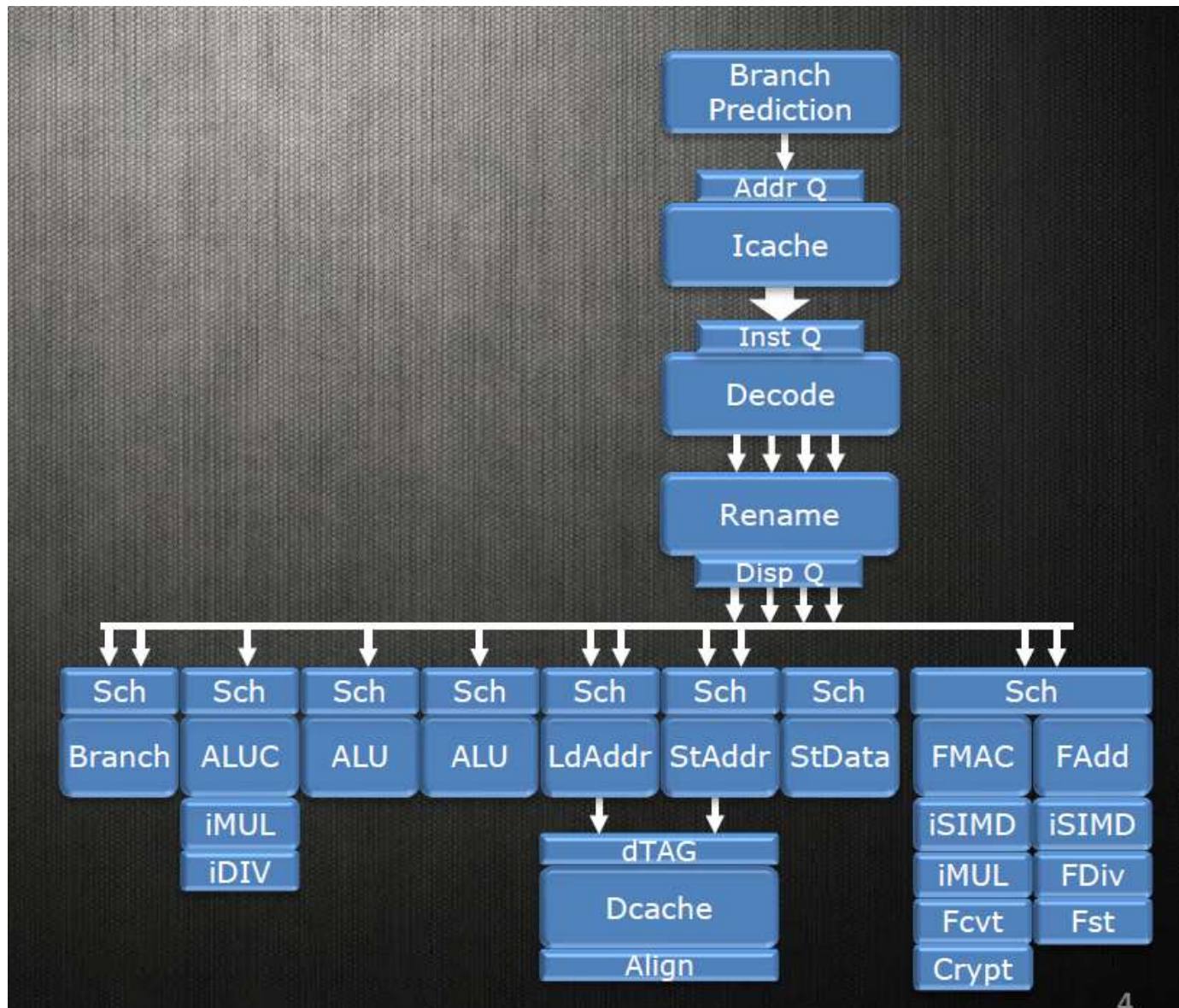
5.5.2 The M1 (Mongoose) core

Main features of the M1 (Mongoose) core

- It implements the [ARMv8 ISA](#).
- It has a 4-wide front-end.
- It has neural net (perceptron) based branch prediction.
- The core is an [out-of-order 2.6 GHz design](#).
- The core design lasted 3 years.
- Subsequently, we point out main features of the Mongoose core design.

5.5.2 The M1 (Mongoose) core (2)

The overall microarchitecture of the M1 core [22]

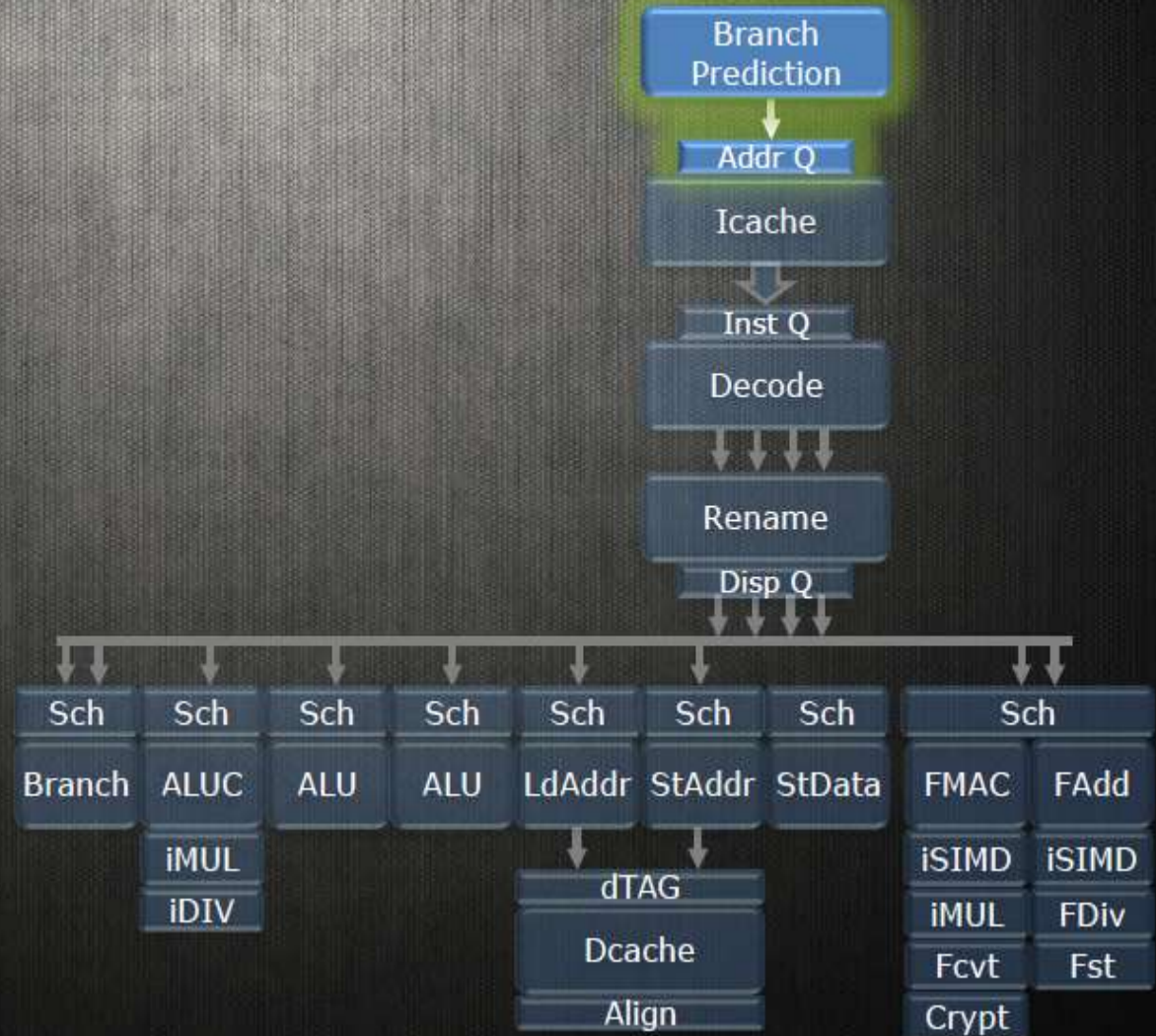


Neural net based branch prediction [22]

Samsung M1 Micro-Architecture

Branch Prediction:

- Neural Net based predictor
- Two branches/cycle
- Fetch up to 24-bytes/cycle
- 64-entry microBTB
- 4k-entry mainBTB
- 64-entry Call/Return Stack
- Indirect Predictor
- Loop Predictor
- Decoupled AddrQ



Remarks on the use of perceptrons for branch prediction

- Each miss prediction causes a number of wasted cycles in instruction processing, the more the longer the instruction pipeline is.
- The efficiency of branch prediction and prefetching are decisive for the achievable ILP and thus for the processor performance.
- Accordingly, the evolution of processors was accompanied by the evolution of branch prediction.
- Recent branch predictors consists of a number of dedicated predictors addressing different types of branches, like direct or indirect branches, loops etc.
- **Perceptron based (called also neural) branch prediction** was first suggested by Vintan (U. Sibiu) [23] in 1999 and then by Jimenez and Lin (U. Texas) [24] in (2001).

5.5.2 The M1 (Mongoose) core (5)

The perceptron model: a single layer perceptron [24]

- The **perceptron**, introduced in 1962, is in fact an **artificial neuron**.
- It **receives a number of inputs** (x_i) that are bipolar (-1 or 1) and **calculates an output value** (y) that is **the sum of the product of the input values** (x_i) and **given weights** (w_i), as shown below.
- A perceptron **can be trained** to provide a prediction.

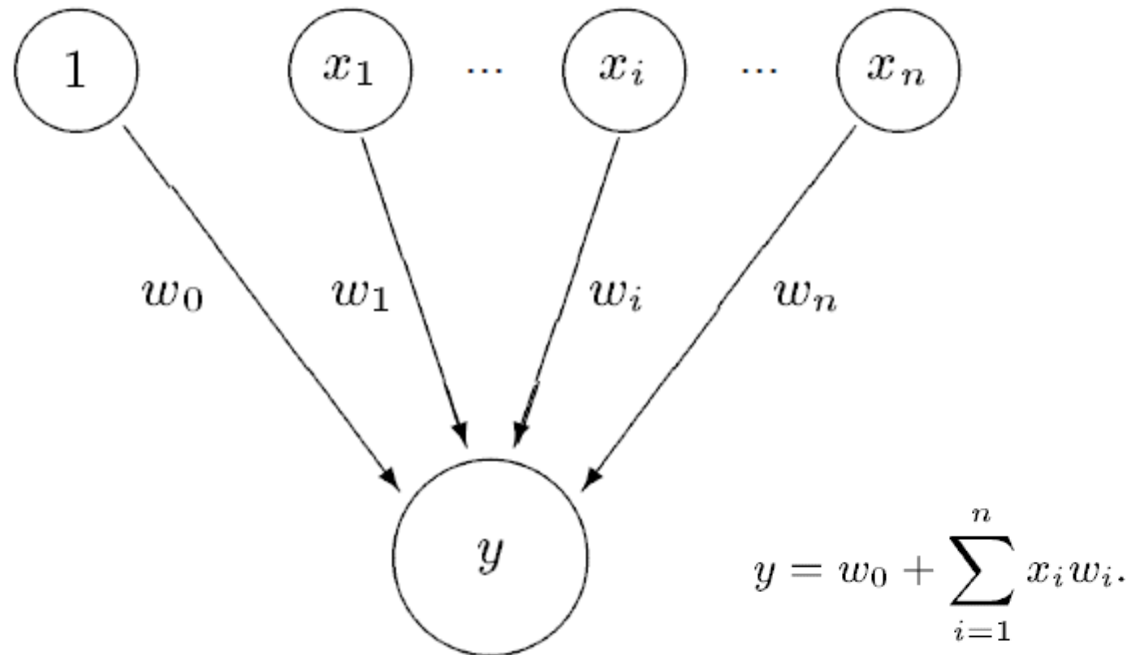


Figure: A single layer perceptron [24]

5.5.2 The M1 (Mongoose) core (6)

Principle of using perceptrons for branch prediction [24]

- Inputs (x_i) are taken from branch history and are -1 or +1.
- The weights (w_i) are small integer values that are learned by on-line training,
- Training finds correlation between history and outcome.
- The output (y) is the dot product of x_i 's and w_i 's, as shown below.
- The output (y) is interpreted as prediction is taken if $y \geq 0$.
- Once the outcome of the prediction (y) becomes known the **training algorithm** uses this value to update the weights (w_i).

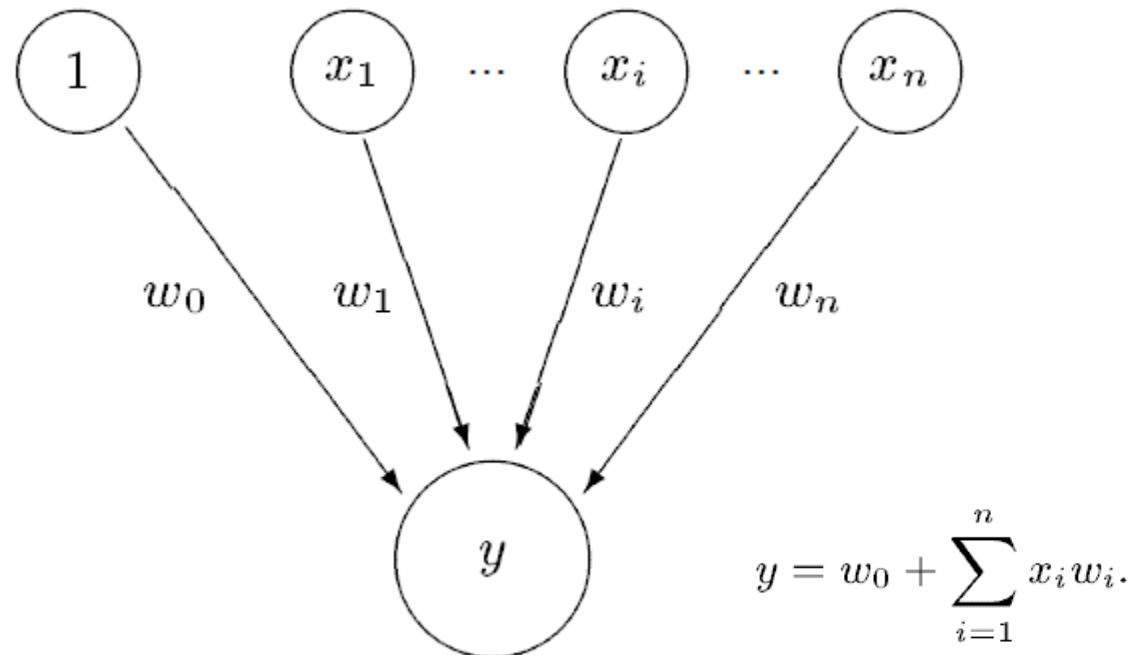


Figure: A single layer perceptron [24]

5.5.2 The M1 (Mongoose) core (7)

Published use of perceptrons (neural networks) for branch prediction

- AMD Bobcat (2011)
 Jaguar (2013)
 Piledriver (2012)
 Zen (2017)
- Oracle SPARCT4 (2011)
- Samsung Exynos Octa 8 8890 M1 (Mongoose) core 2016

Implementation of perceptron based (neural) branch prediction [24]

- Actually there are **no published details** about the perceptron based (neural) branch predictors used in the processors enlisted above.
- As an example below we show AMD's related slide "revealing the use of neural branch prediction in their Zen processor (2017).

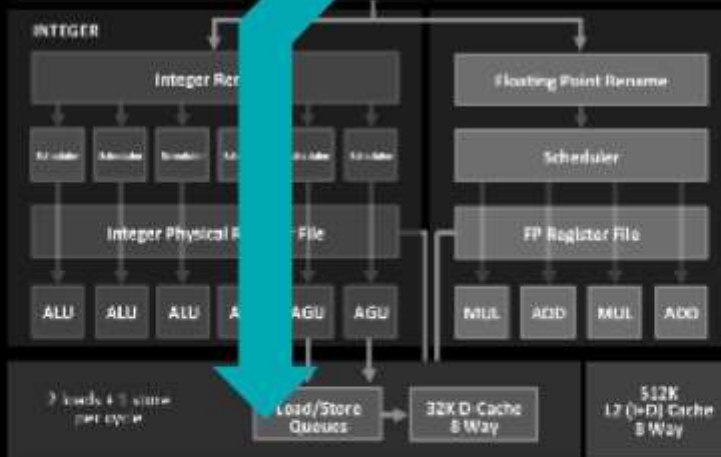
5.5.2 The M1 (Mongoose) core (9)

Example: Perceptron based branch prediction in AMD's Zen microarch. [58]



Scary Smart Prediction

- ▶ A true artificial network inside every “Zen” processor
- ▶ Builds a model of the decisions driven by software code execution
- ▶ Anticipates future decisions, pre-load instructions, choose the best path through the CPU



Shared L3 Cache

5.5.2 The M1 (Mongoose) core (10)

Achieved accuracy of recent sophisticated branch predictors

- Since 2004 the The Journal of Instruction-Level Parallelism organizes each third year a Championship Branch Prediction (CBP-1 to CPB5) [25].
- Presented predictors for conditional branches are evaluated on a given trace set by calculating the weighted average of **Mispredictions Per Thousand Instructions (MPTI)**.
- Predictors must be implemented within a fixed storage budget of 8 kB, 64 kB or unlimited.
- The **best results** reveal astonishingly low misprediction rates [26]:

Storage budget	MPTI
8 KB	5.3
64 KB	4.1
Unlimited	3.0

Remarks on Samsung's perceptron based branch predictor implementation in Zen

- The designer of AMD's first branch prediction logic for the first microprocessor with a neural network branch predictor (AMD Bobcat) (James Dundas) left AMD and joined Samsung in 2012 [27].
- Also the Chief Architect of the Bobcat processor (Brad Burgess) left AMD, joined Samsung and became the Chief CPU Architect in 2011 [28].

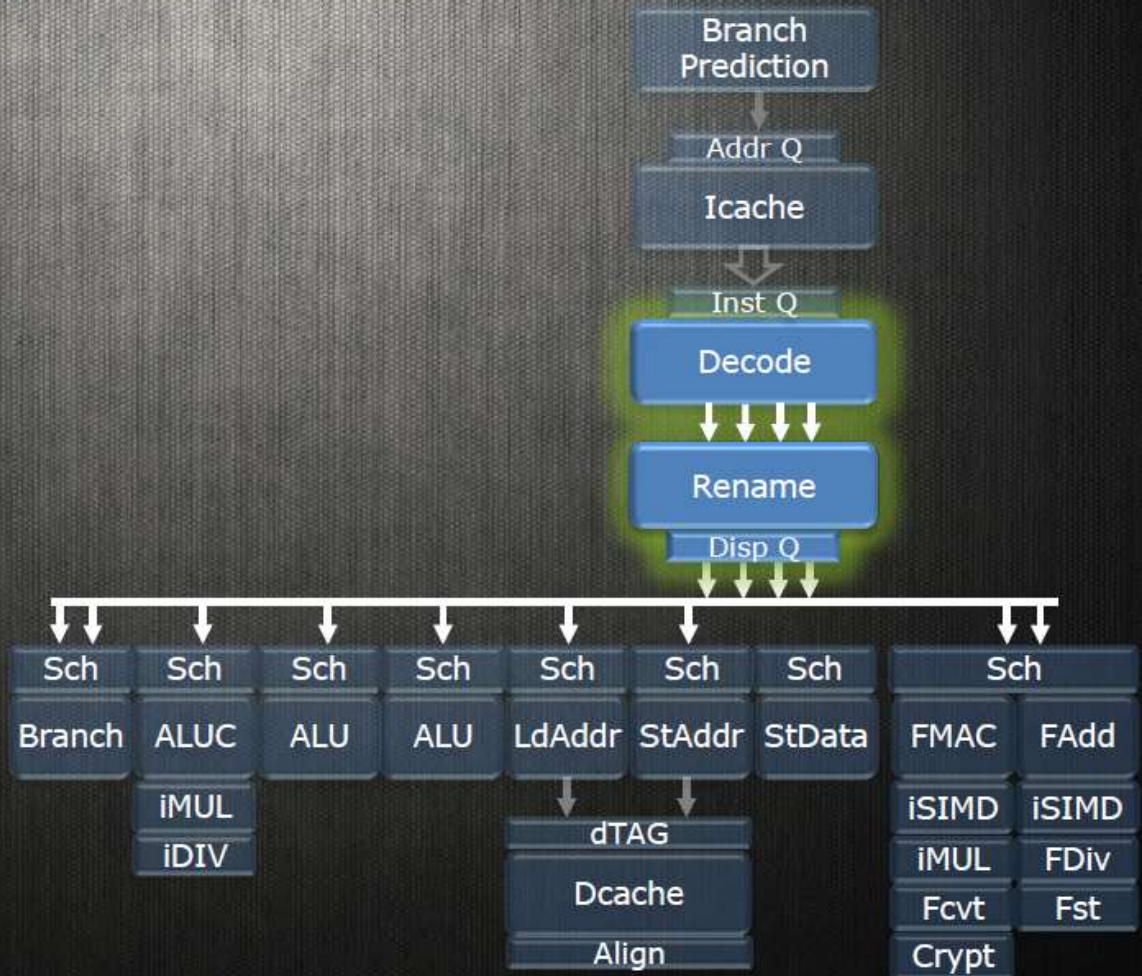
4-wide front end rather than 2 to 3 as in most mobiles [22]

Samsung M1

Micro-Architecture

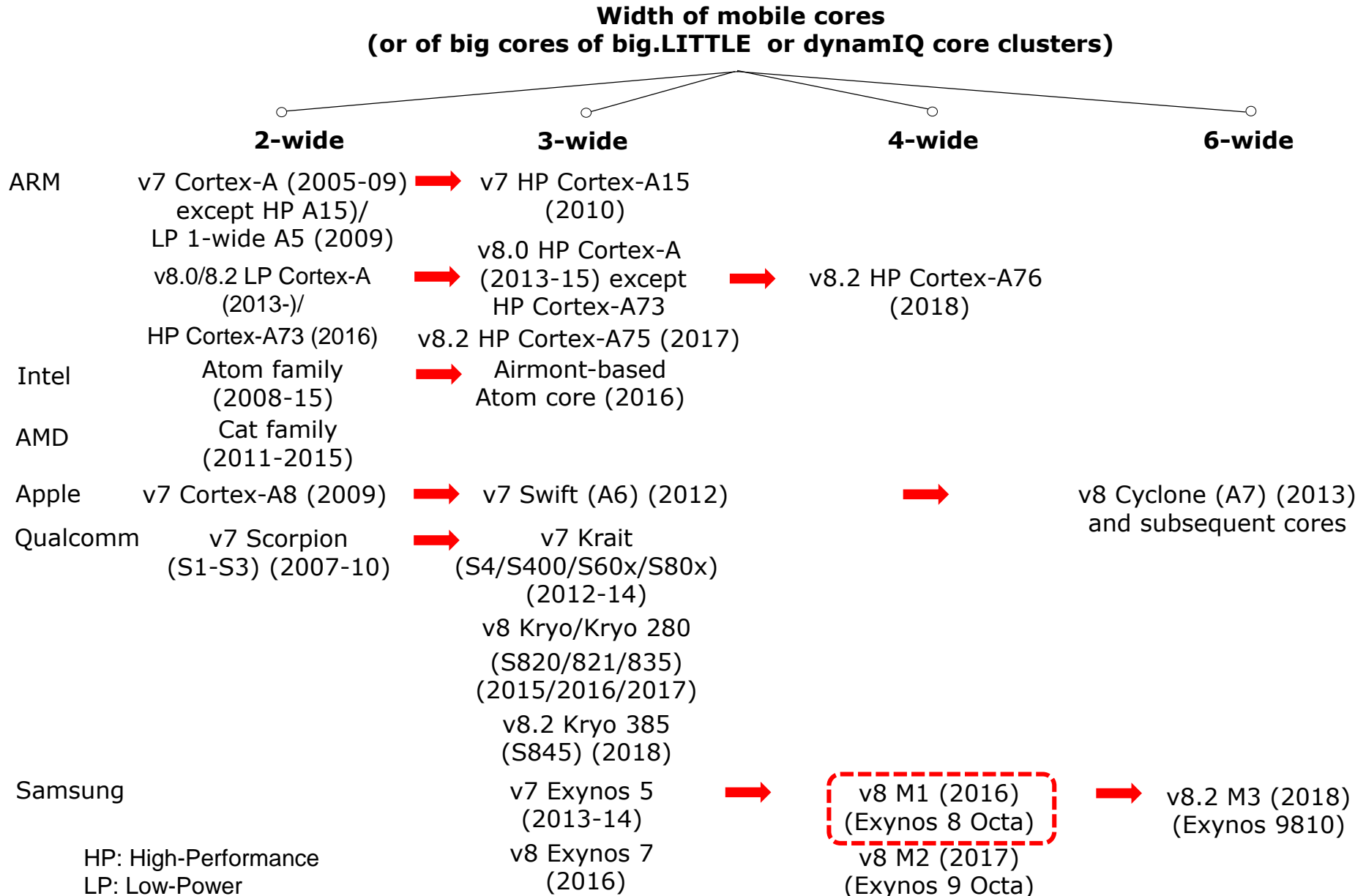
Decode / Rename / Retire:

- Decode 4 inst/cycle
- AArch64, AArch32
- Sequencer for multi-uop
- Rename 4-uops/cycle
- Special renaming for FP
- Fast map recovery
- Retire 4-uops/cycle
- 96-entry ROB
- Dispatch 4-uops/cycle



5.5.2 The M1 (Mongoose) core (12b)

Evolution of the width of mobile cores

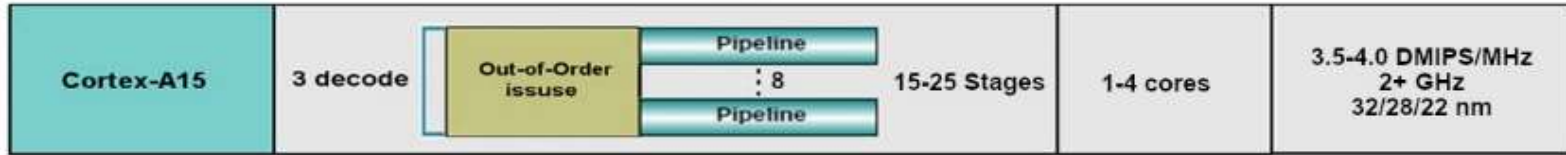


5.5.2 The M1 (Mongoose) core (13)

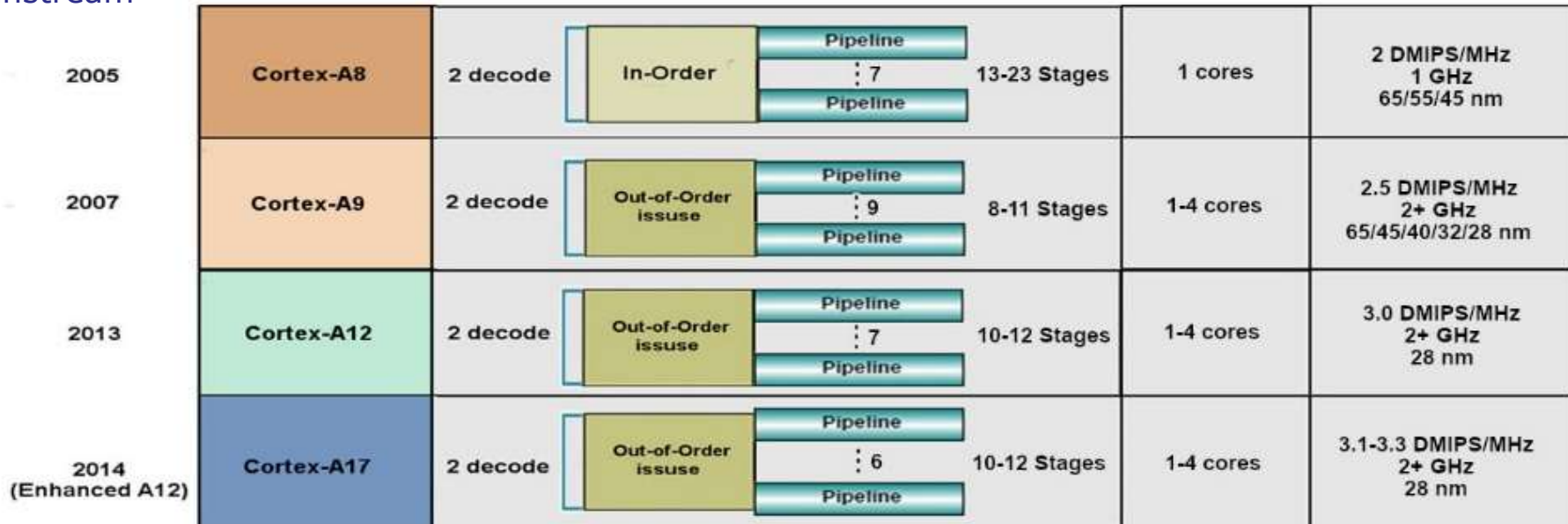
By contrast: Width of ARM v7 ISA based microarchitectures (based on [29])

High performance

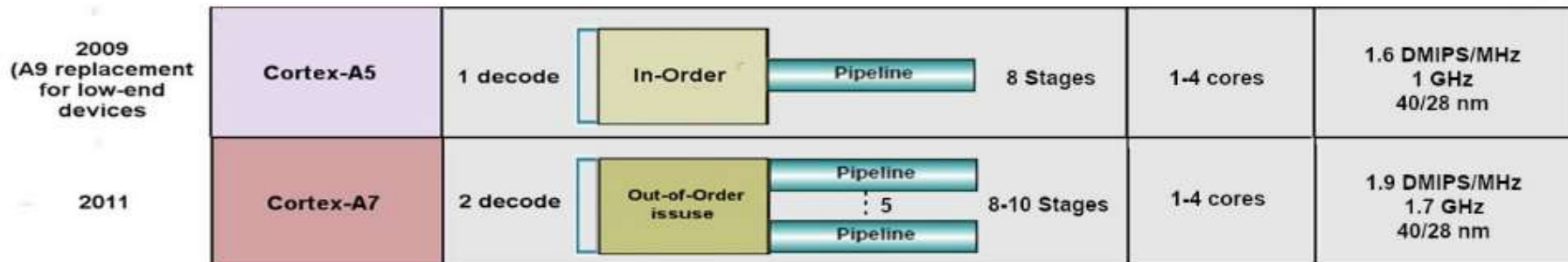
32-bit



Mainstream



Low power



5.5.2 The M1 (Mongoose) core (14)

By contrast: Width of ARM v8.0 ISA based microarchitectures (based on [29])

High performance

64-bit

2013	Cortex-A57	3 decode	Out-of-Order issue	 8 15+ Stages	1-4 cores	4.1-4.7 DMIPS/MHz Up to 2.0 GHz 28/20/16/14 nm
2015	Cortex-A72	3 decode	Out-of-Order issue	 8 15+ Stages	1-4 cores	6.3-7.35 DMIPS/MHz Up to 2.5 GHz 28/16 nm
2016	Cortex-A73	2 decode	Out-of-Order issue	 7 11 Stages	1-4 cores	7.4-8.5 DMIPS/MHz Up to 2.8 GHz 10 nm

Low power

64-bit


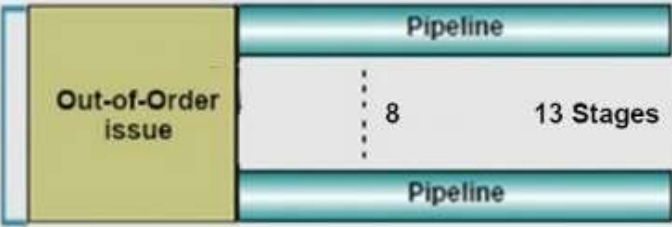
2013	Cortex-A53	2 decode	In-Order	 5 8 Stages	1-4 cores	2.3 DMIPS/MHz 1.2+ GHz 28/20/16/14/10 nm
2015	Cortex-A35	2 decode	In-Order	 6 8 Stages	1-4 cores	~2.1 DMIPS/MHz Up to 2.0 GHz 28/20/16/14/10 nm

5.5.2 The M1 (Mongoose) core (14b)

By contrast: Width of ARM v8.2 ISA based microarchitectures (based on [29])

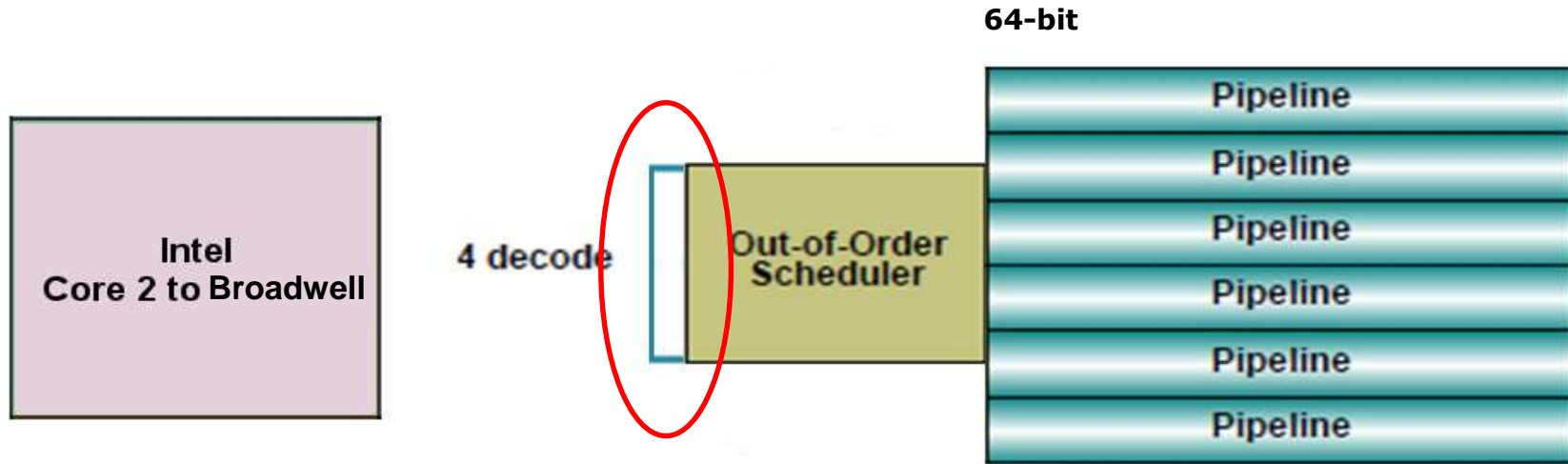
High performance

64-bit

2017	Cortex-A75	3 decode		1-4 cores	$\approx 8.2\text{-}9.5$ DIMPS/MHz Up to 3 GHz 10 nm
2018	Cortex-A76	4 decode		1-4 cores	$\approx 10.7\text{-}12.4$ DIMPS/MHz Up to 3 GHz 12/7/5 nm

5.5.2 The M1 (Mongoose) core (15)

By contrast: Front-end width of Intel's and AMD's recent microarchitectures



Remarks

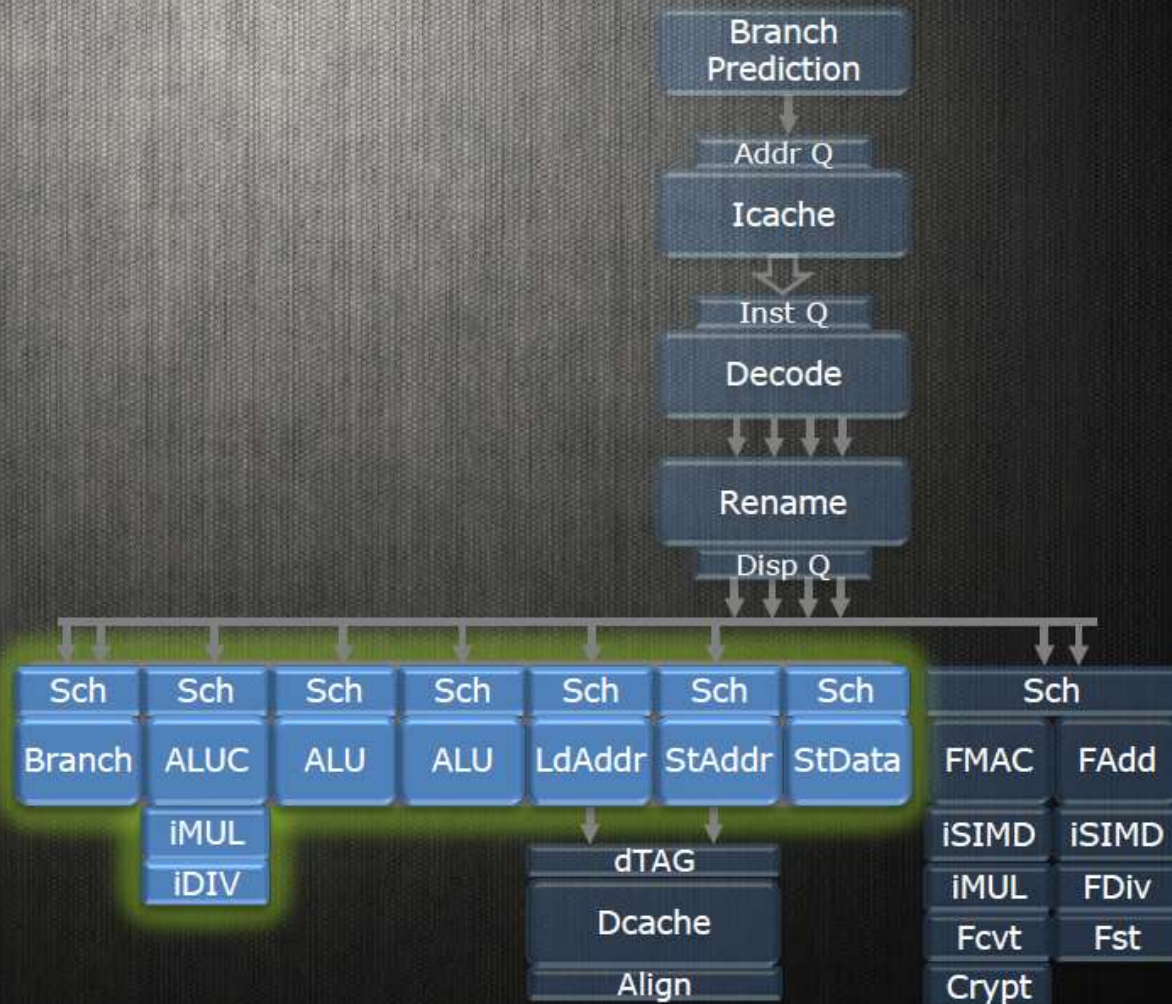
- Intel introduced 4-wide front ends beginning with their Core 2 (2006).
- Since Skylake Intel widened the front-end of its processor to 5.
- AMD introduced 4-wide microarchitectures only five years later, along with the Bulldozer line in 2011.

7-wide FX- and Load/Store scheduler [22]

Samsung M1 Micro-Architecture

Integer Execution:

- Issue up to 7 uops/cycle
- 96-entry integer PRF
- 58-entry distributed sched.
- Branch resolution
- ALUC – three source uops
- ALU – two source uops
- Load Address Adder
- Store Address Adder
- Store Data

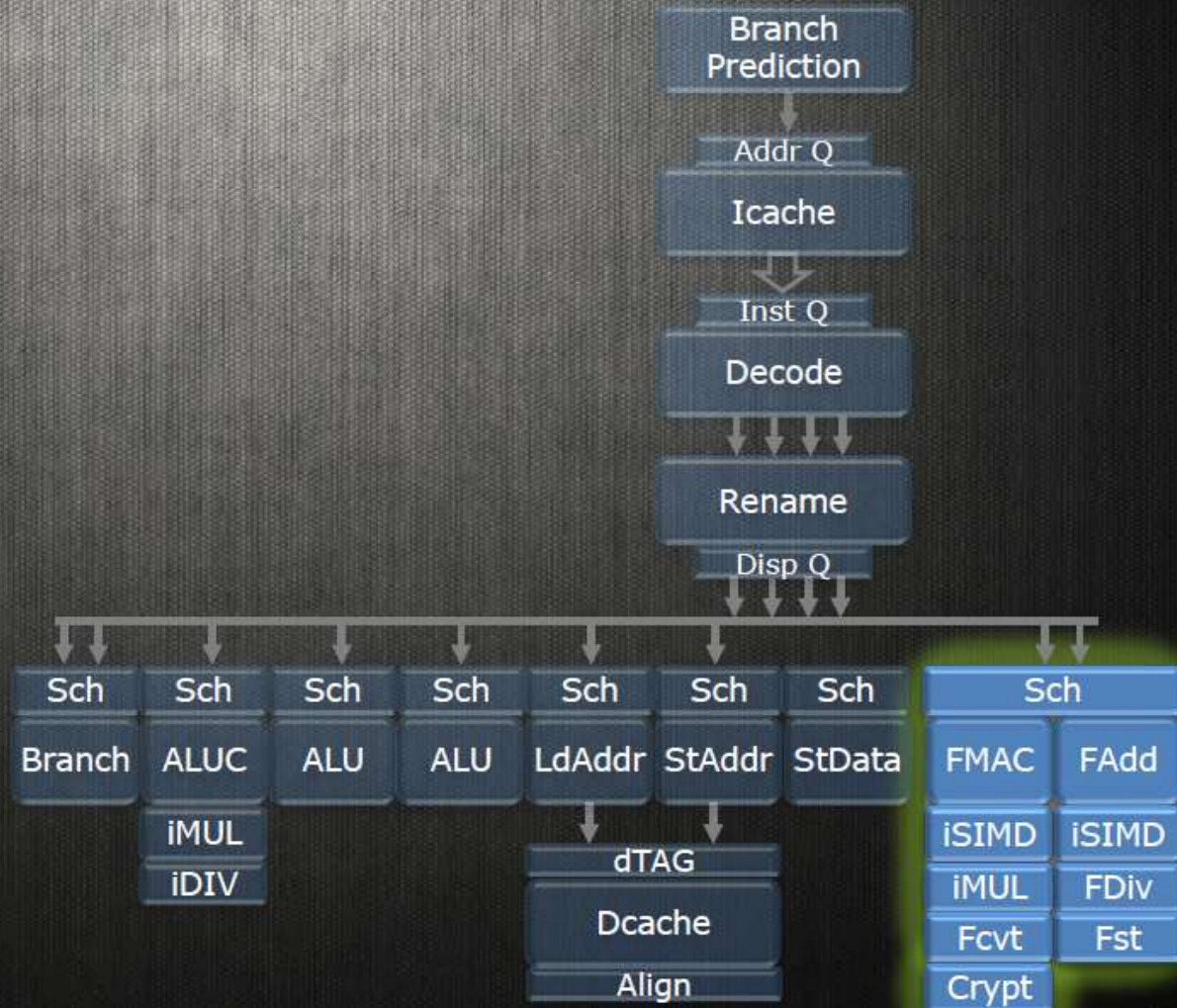


2-wide FP-issue, FMAC and FADD operations [22]

Samsung M1 Micro-Architecture

Floating Point Execution:

- 32-entry scheduler
- 96-entry FP PRF
- FMAC : 5-cycle MAC
4-cycle Mul
- FADD: 3-cycle



Pipeline structure of the M1 [22]

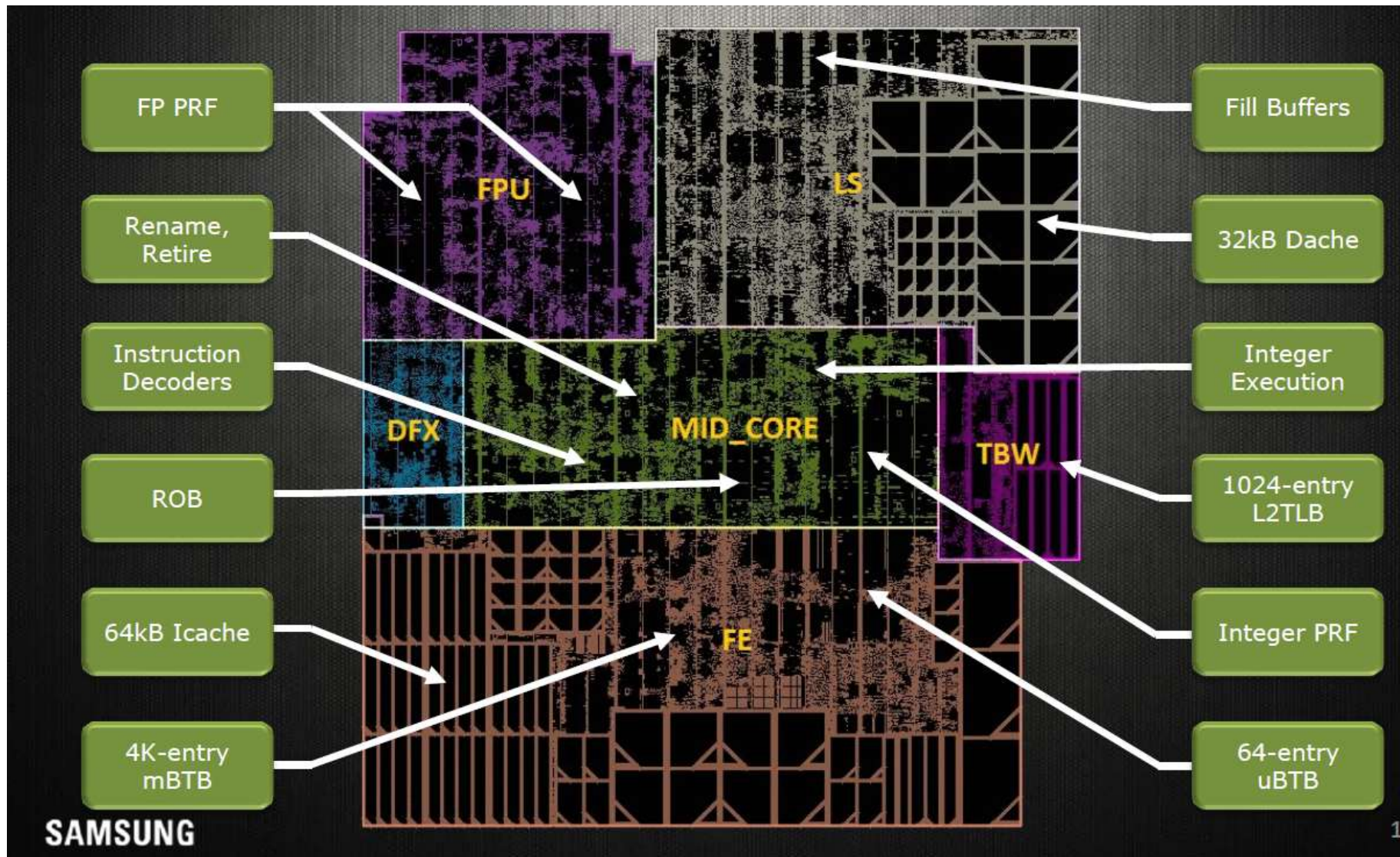
Samsung M1

Basic Pipeline



5.5.2 The M1 (Mongoose) core (19)

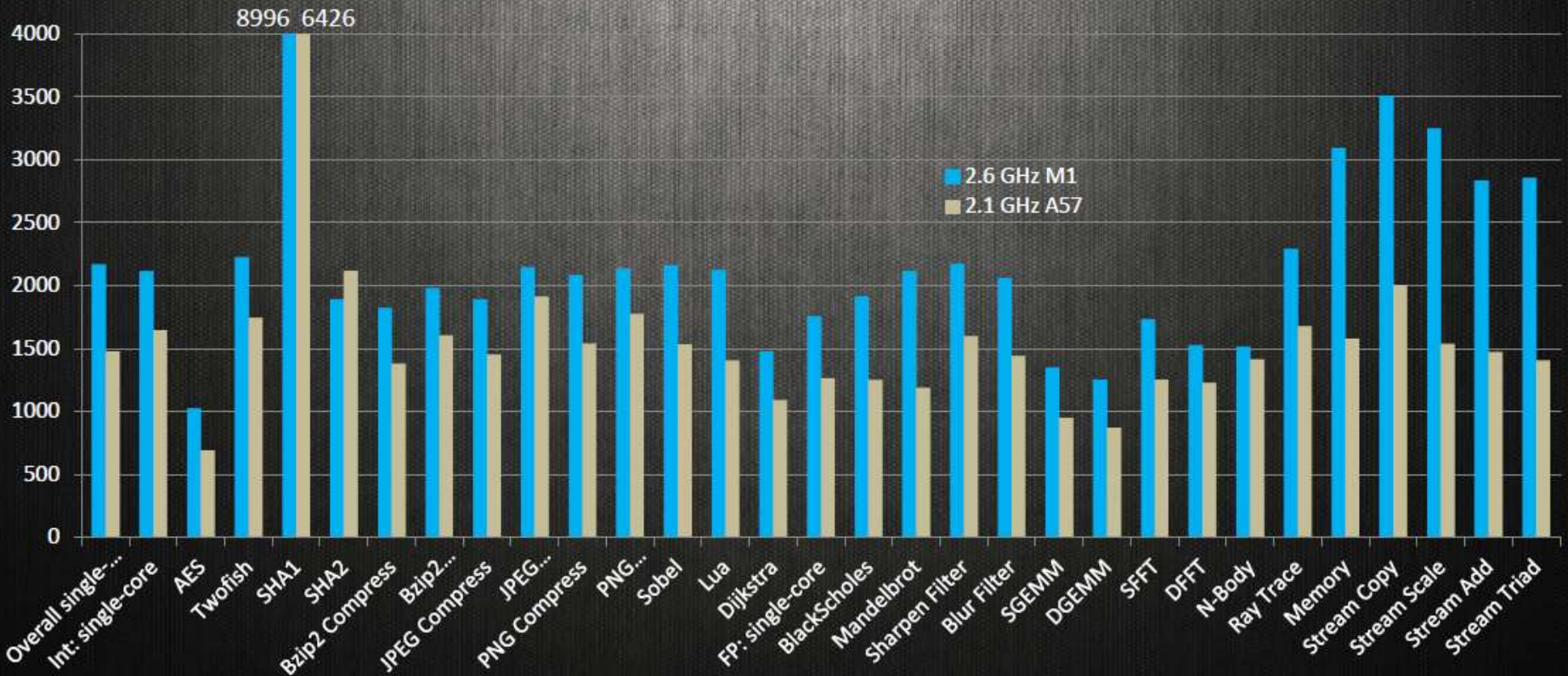
Die layout of the M1 core [22]



5.5.2 The M1 (Mongoose) core (20)

Single-core performance of the 2.6 GHz M1 vs. the 2.1 GHz A57 [22]

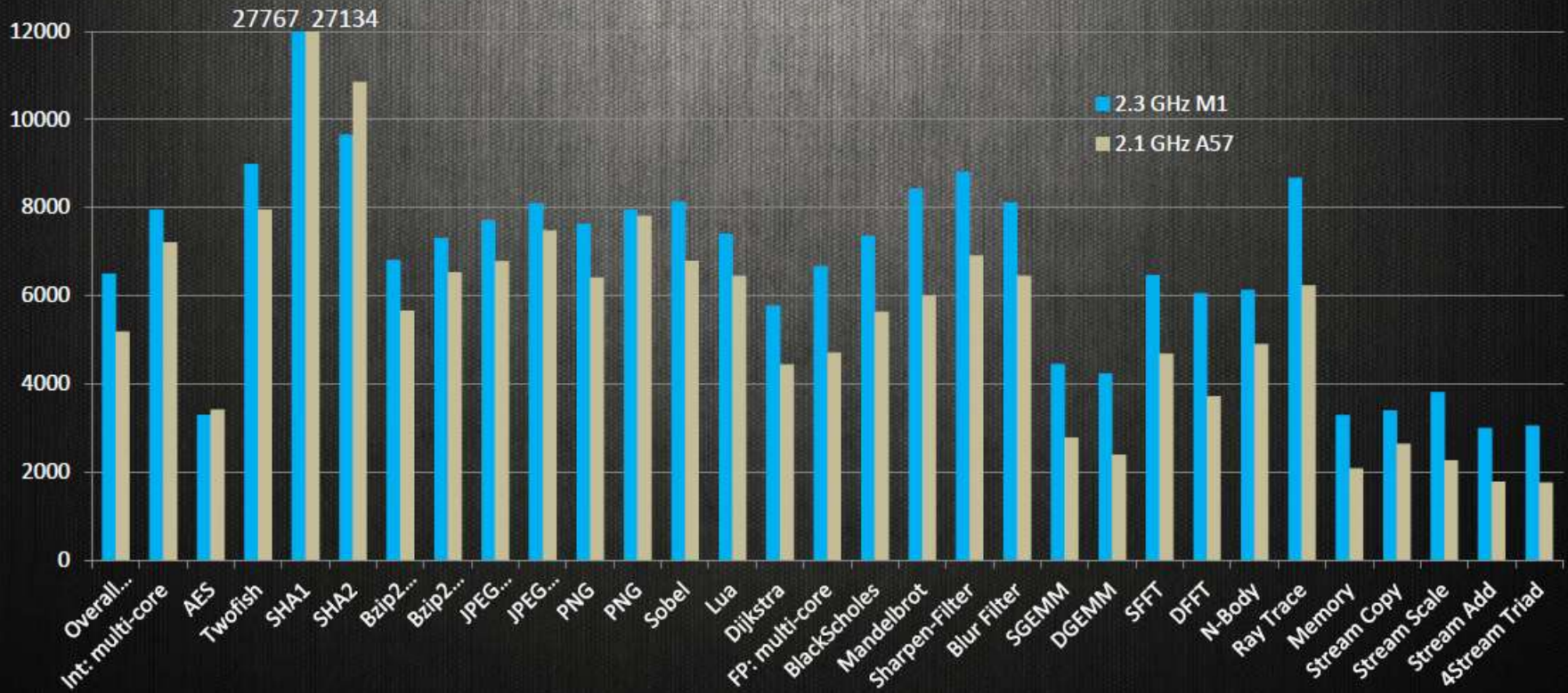
Samsung M1 Single-core Performance



5.5.2 The M1 (Mongoose) core (21)

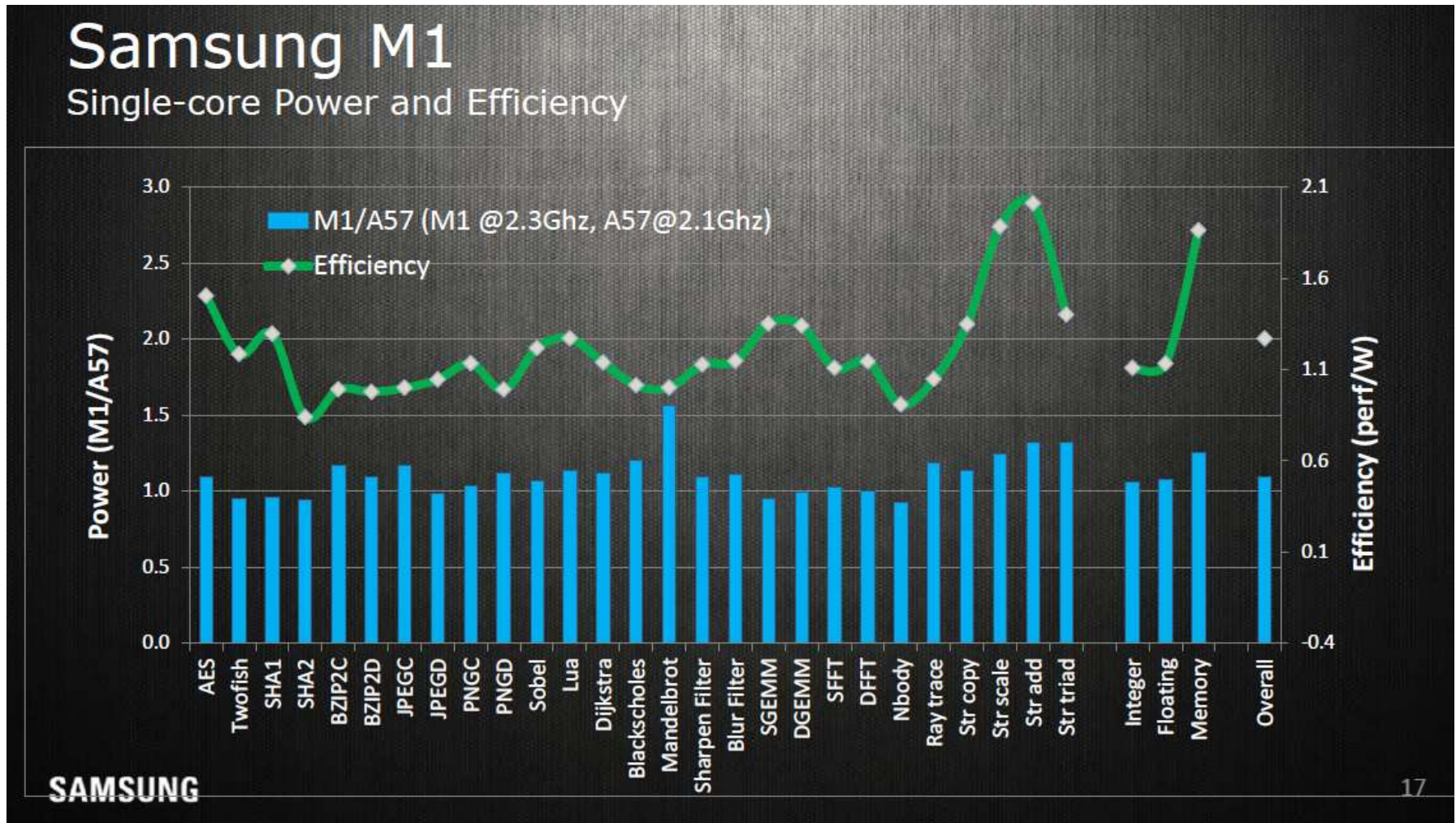
Multi-core performance of the 2.3 GHz M1 vs. the 2.1 GHz A57 [22]

Samsung M1 Multi-core Performance



5.5.2 The M1 (Mongoose) core (22)

Single core power and efficiency data of the M1 vs. the A57 [22]



5.6 Samsung's first 10 nm SOC: the Exynos 9 8895 (2017)

- 5.6.1 The Exynos 9 Series 8895 - Overview
- 5.6.2 Integrating ARM's next generation Mali-G71 GPU that is based on ARM's 3. gen. (Bifrost) GPU architecture
- 5.6.3 HSA (Heterogeneous System Architecture) compliance
- 5.6.4 Support for LPDDR4x memory
- 5.6.5 Separate security processing unit
- 5.6.6 Vision Processing Unit (VPU)

5.6.1 The Exynos 9 Series 8895 - Overview

5.6.1 The Exynos 9 Series 8895 - Overview

- It is Samsung's **first SoC** fabricated on their **10 nm FinFET process**.
The 10nm FinFET process allows **up to 27% higher performance or 40% lower power consumption** when compared to 14nm LPE FinFET [30].
- It is the kernel piece of one alternative of Samsung's **Galaxy S8, S8+**.
The other alternative is using Qualcomm's Snapdragon 835 for these mobiles (sold in the US).
- It was **announced in 02/2017 and shipped in 04/2017**.

5.6.1 The Exynos 9 Series 8895 - Overview (2)

Main features of Samsung's Exynos 9 Octa 8895 (2017)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (Exynos 5420)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (Exynos 5422)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (Exynos 5800)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (Exynos 5430)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (Exynos 5433)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
Exynos 7 Octa (Exynos 7420)	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
Exynos 7 Octa (Exynos 7885)	14 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
Exynos 8 Octa (Exynos 8890)	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
Exynos 9 Series (Exynos 8895)	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
Exynos 9 Series (Exynos 9810)	10 nm FinFET		Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus

5.6.1 The Exynos 9 Series 8895 - Overview (3)

Main enhancements of the Samsung Exynos 9 Series 8995 [30]

- It is built up further on a big.LITTLE architecture while as big cores Samsung employs their **second generation custom core**, designated as the **M2 (Mongoose)** core.
- It is based on Samsung's **upgraded custom SCI (Samsung Coherent Interconnect) bus** (the SCI was introduced in the Exynos 8 Octa 8890).

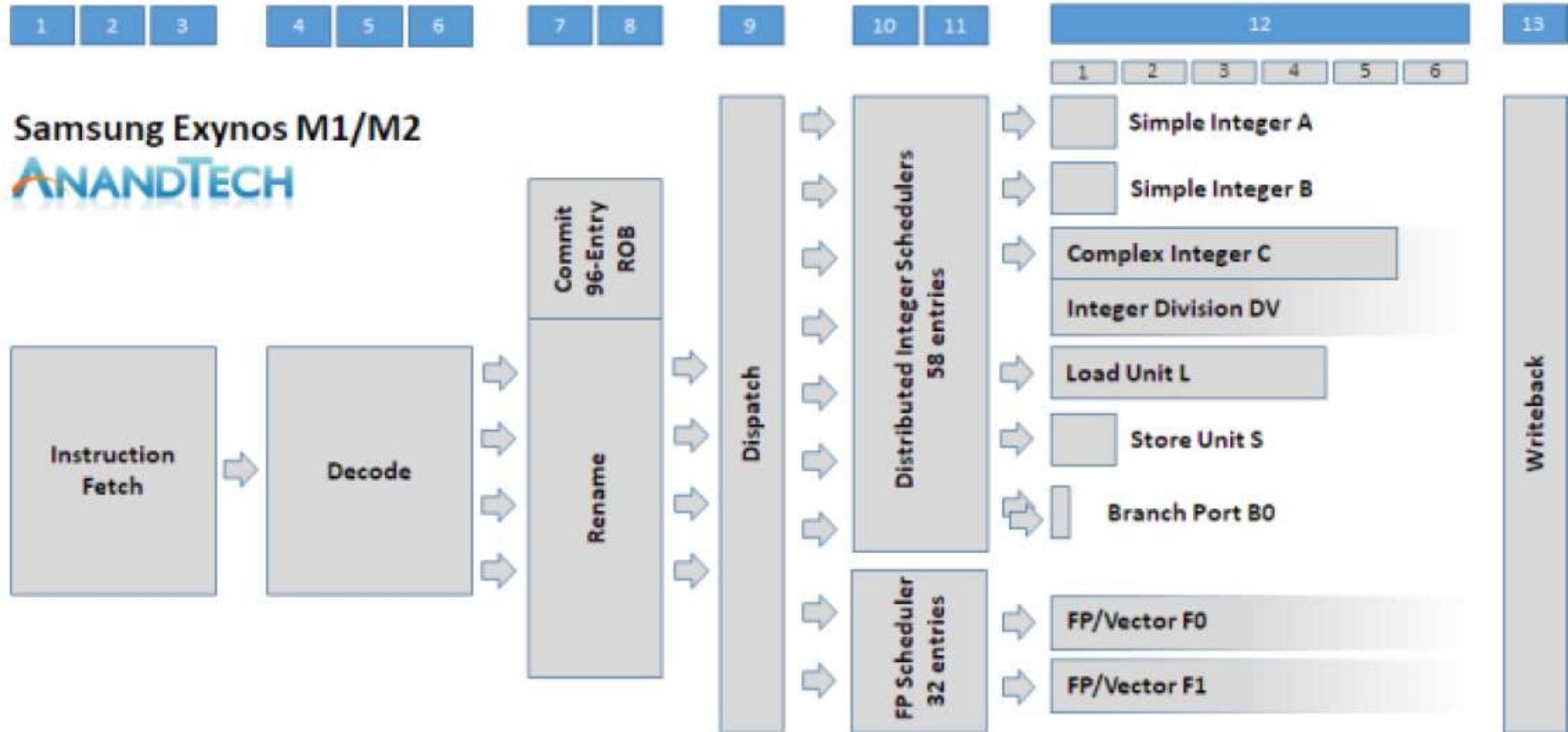
Upgrading the SCI bus for supporting **HSA (Heterogeneous System Architecture)**.

- **Upgraded modem** that implements
 - Cat 16 LTE for downloading at 1 Gbps by using 5x Carrier Aggregation and
 - Cat 13 LTE for uploading at 150 Mbps by using 2x Carrier Aggregation.



5.6.1 The Exynos 9 Series 8895 - Overview (4)

Microarchitecture of the M1/M2 cores [67]



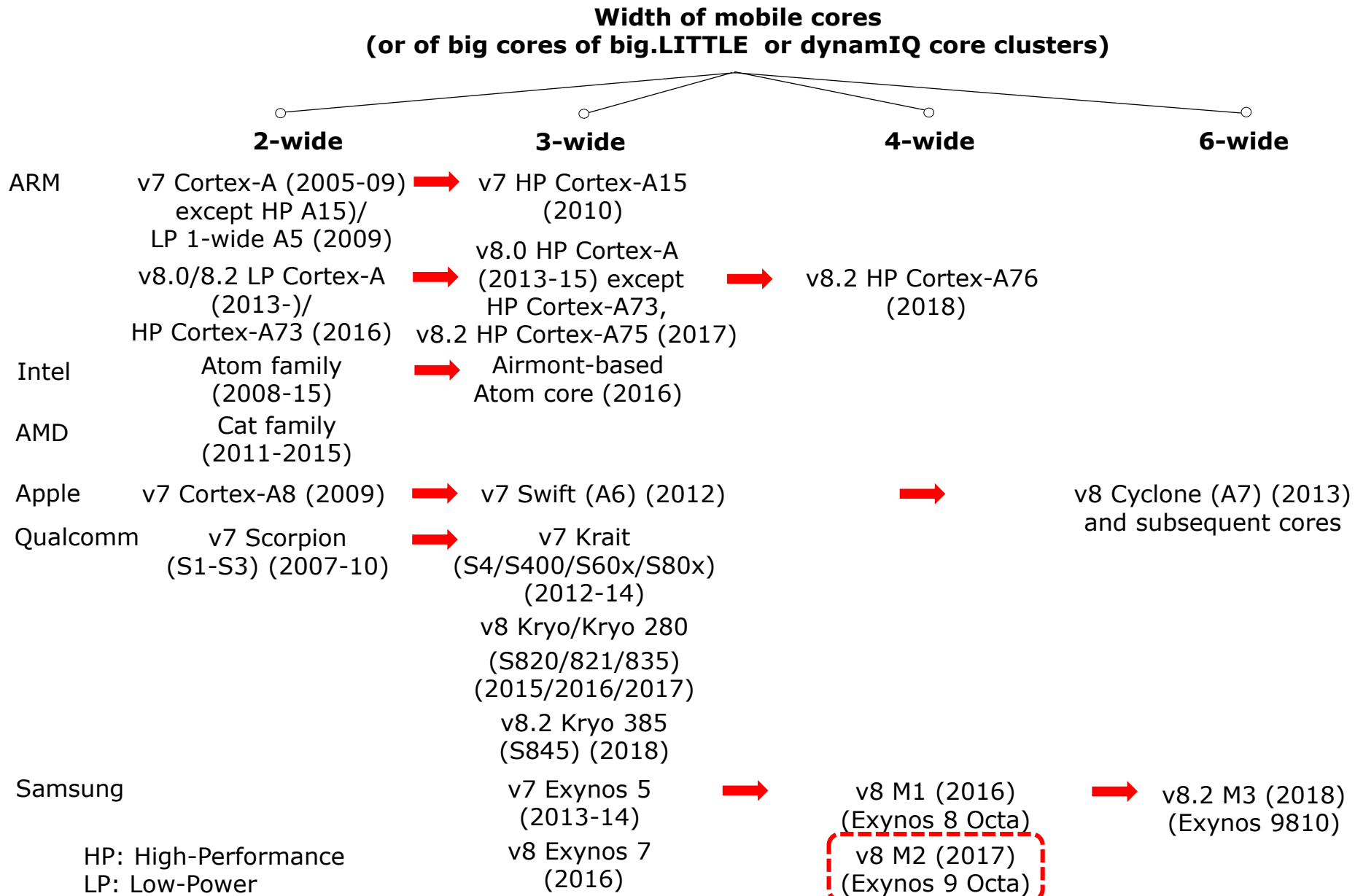
Key features of the microarchitecture of the M1/M2 cores (1)

The **front-end** part of the microarchitecture of the M1/M2 cores is **4-wide**, i.e. there are **4-wide in-order stages for decoding and dispatching**.

This is an unusual wide front-end considering mobile cores, as seen in the next slide.

5.6.1 The Exynos 9 Series 8895 - Overview (6)

Evolution of the width of mobile cores



Key features of the microarchitecture of the M1/M2 cores (2)

The **back-end** part of the M1/M2 microarchitecture includes **9 ports**, as follows [67]:

- Two simple ALU pipelines capable of integer additions.
- A complex ALU handling simple operations as well as integer multiplication and division.
- A load unit port
- A store unit port
- Two branch prediction ports
- Two floating point and vector operations ports leading to two mixed capability pipelines.

Main innovations of the Exynos 9 Series 8895 [31]

- a) Integrating ARM's next generation **Mali-G71** GPU that is based on ARM's new **Bifrost GPU architecture**.
- b) **HSA (Heterogeneous System Architecture)** compliant processor implementation.
- c) Support for **LPDDR4x** memory.
- d) Separate **security processing unit**.
- e) **Vision Processing Unit (VPU)**.

5.6.1 The Exynos 9 Series 8895 - Overview (9)

Comparing key features of Samsung's advanced Exynos models [31]

Samsung Exynos SoCs Specifications			
SoC	Exynos 8895	Exynos 8890	Exynos 7420
CPU	4x A53	4x A53@1.6GHz	4x A53@1.5GHz
	4x Exynos M2(?)	4x Exynos M1 @ 2.3GHz	4x A57@2.1GHz
GPU	Mali G71MP20	Mali T880MP12 @ 650MHz	Mali T760MP8 @ 770MHz
Memory Controller	2x 32-bit(?) LPDDR4x	2x 32-bit LPDDR4 @ 1794MHz	2x 32-bit LPDDR4 @ 1555MHz
		28.7GB/s b/w	24.8GB/s b/w
Storage	eMMC 5.1, UFS 2.1	eMMC 5.1, UFS 2.0	eMMC 5.1, UFS 2.0
Modem	Down: LTE Cat16 Up: LTE Cat13	Down: LTE Cat12 Up: LTE Cat13	N/A
ISP	Rear: 28MP Front: 28MP	Rear: 24MP Front: 13MP	Rear: 16MP Front: 5MP
Mfc. Process	Samsung 10nm LPE	Samsung 14nm LPP	Samsung 14nm LPE

5.6.1 The Exynos 9 Series 8895 - Overview (10)

Comparing key features of the 10 nm Qualcomm's Snapdragon 835 and Samsung's Exynos 8895 [32]

	Qualcomm Snapdragon 835	Samsung Exynos 8895
Manufacturing Process	10nm FINFET	10nm FINFET
CPU Config	Quad 2.45GHz Kryo 280 + Quad 1.9GHz Kryo 280	Quad 2.5GHz Samsung Mongoose + Quad 1.7GHz Cortex-A53
GPU	Adreno 540	Mali-G71 MP20
RAM	LPDDR4X	LPDDR4X
Camera support	Up to 32MP, or dual 16MP	Upto 28MP or 28MP+16MP
Flash	UFS 2.1 or eMMC 5.1	UFS 2.0 or eMMC 2.1
Video Shooting	UHD at 30fps	UHD 120fps
Video Playback	4K at 60fps, H.265 (HEVC), 10-bit H.264 (AVC), VP9 codecs	4K at 120fps, H.264, HEVC (H.265), VP9 codecs
Data Speeds	1Gbps down, 150 Mbps up	1 Gbps down, 150 Mbps up

5.6.2 Integrating ARM's next generation Mali-G71 GPU that is based on ARM's 3. gen. (Bifrost) GPU architecture

5.6.2 Integrating ARM's next generation Mali-G71 GPU (1)

5.6.2 Integrating ARM's next generation Mali-G71 GPU
that is based on ARM's 3. gen. (Bifrost) GPU architecture

Remark

Brief history of ARM's Mali GPU development

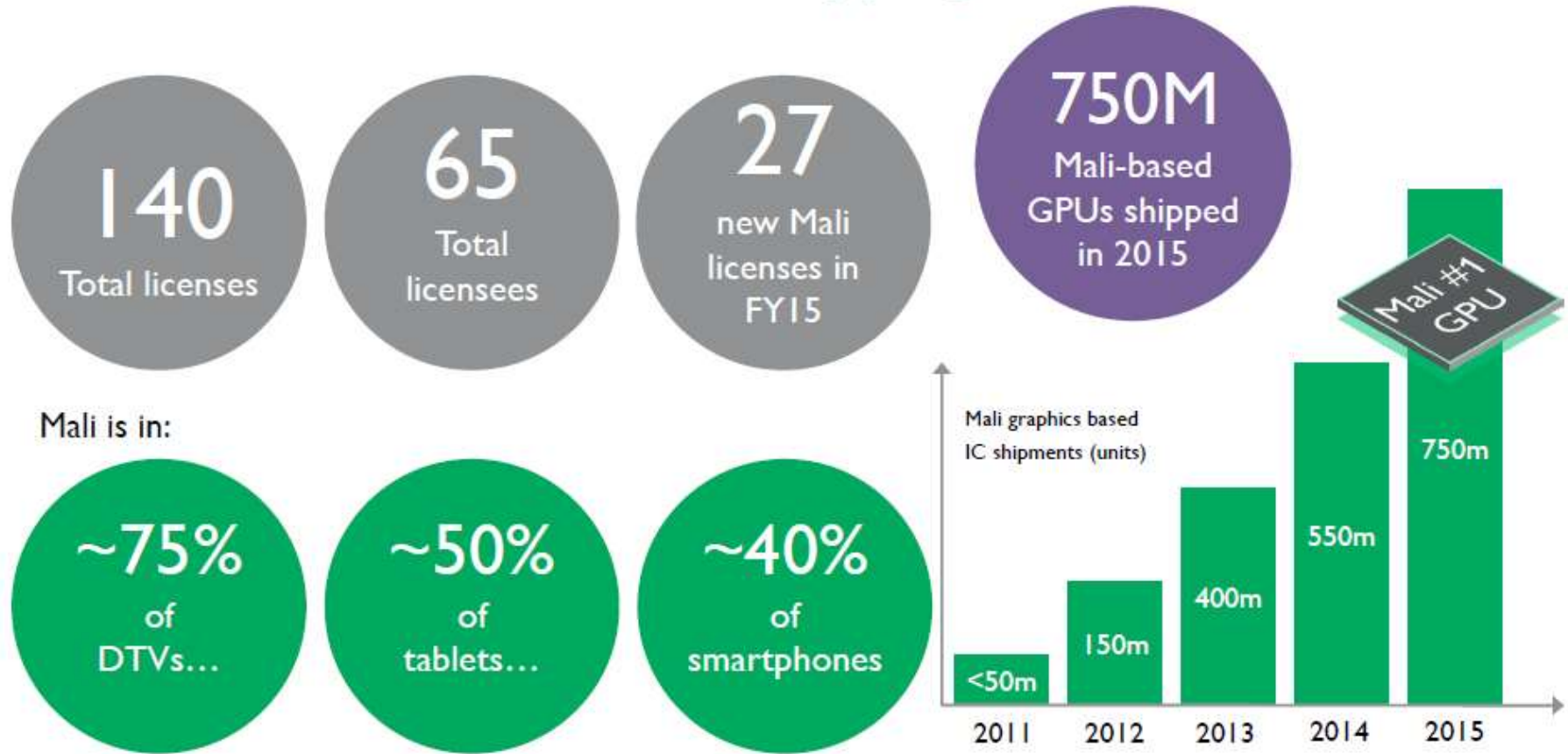
- The Mali graphics research group of the Norwegian University of Science and Technology was spun off and established the Falanx Microsystems in 2001.
- Originally, Falanx intended to break into the PC video card market but lack of adequate financing the firm changed its profile and started to design SoC-class GPUs and license those designs to SoC integrators.

Such an early design was the Mali-55.

- Later, when SoC industry began to flourish due to growing cell phone sales, ARM purchased Falanx in 2006, in the same year when AMD acquired ATI. Thus Falanx became ARM's GPU division.
- The division released their first OpenGL ES 2.0 design in 2007, the Mali-200 followed by the successors Mali-300, Mali-400, and Mali-450.
- All these designs were based on the team's Utgard architecture (see later).
- Recently the division has nearly 500 designers.
- To date the Mali family became the world's no. 1 shipping GPU, as indicated next.

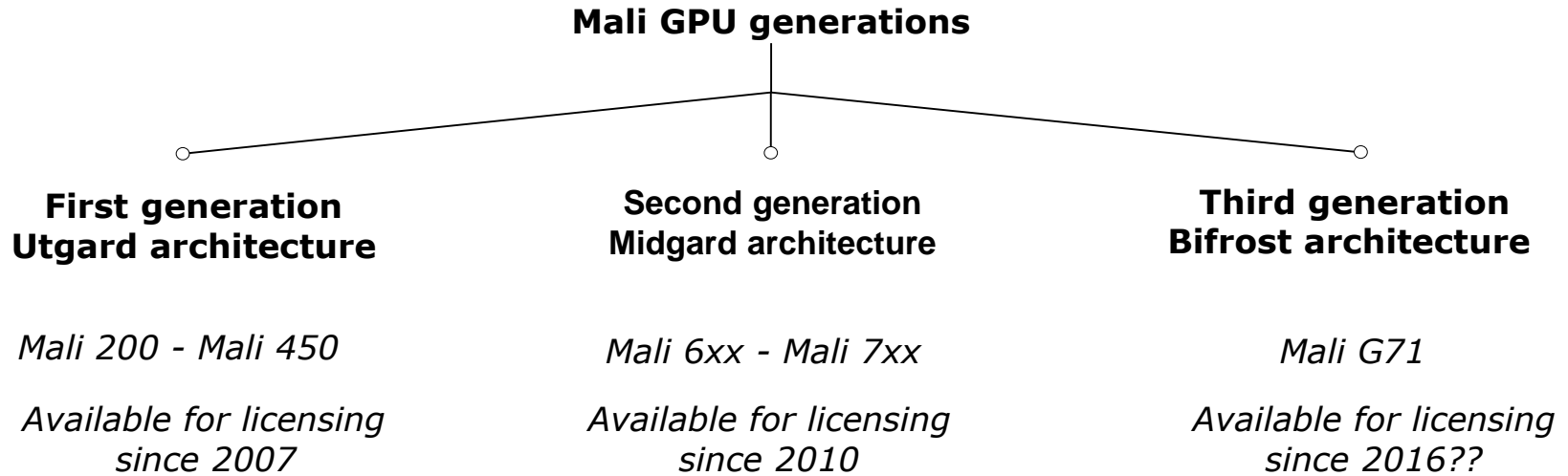
5.6.2 Integrating ARM's next generation Mali-G71 GPU (3)

Worldwide market share of the Mali GPUs [33]



5.6.2 Integrating ARM's next generation Mali-G71 GPU (4)

Evolution of the Mali GPU design



5.6.2 Integrating ARM's next generation Mali-G71 GPU (5)

Remark to the naming of the Mali architecture generations

- The Mali research team comes originally from the Norwegian University of Science and Technology.
- In connection with this Mali's GPU generations were named from the Norse (Scandinavian) mythology, as follows.

Utgard: is a stronghold surrounding the land of the giants.

Midgard: is the realm of humans that is surrounded by an ocean.

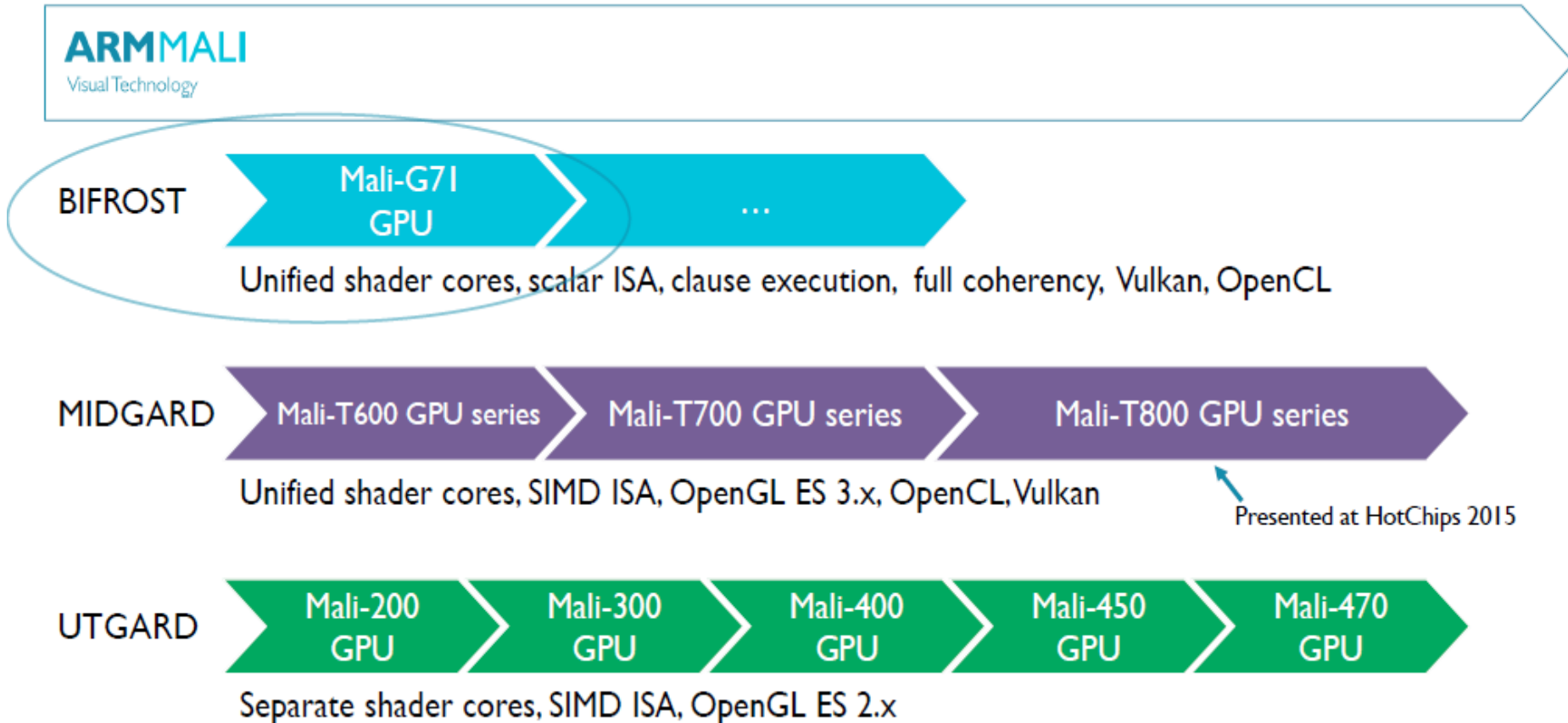
Biforce: is the rainbow bridge that connects Asgard, the world of the gods, with Midgard, the realm of humans.



Figure: Biforce, the rainbow bridge connecting the world of gods with the realm of humans [34]

5.6.2 Integrating ARM's next generation Mali-G71 GPU (6)

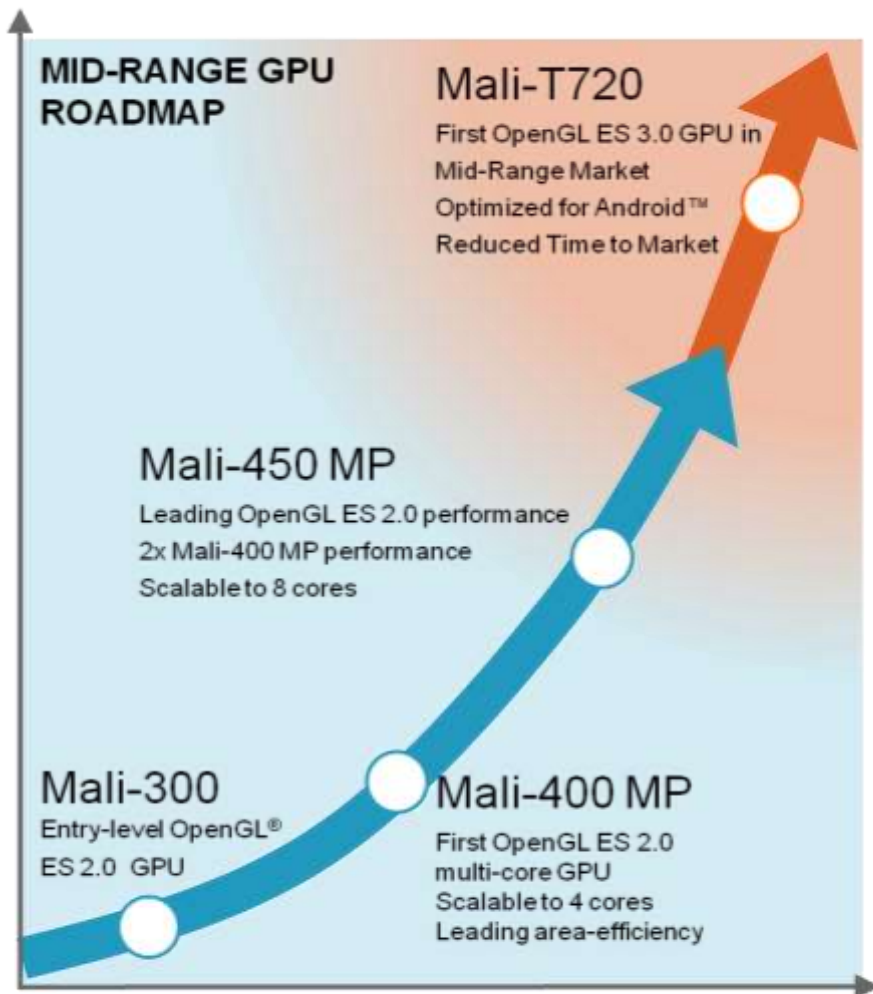
Key features of the Mali graphics processor generations [33]



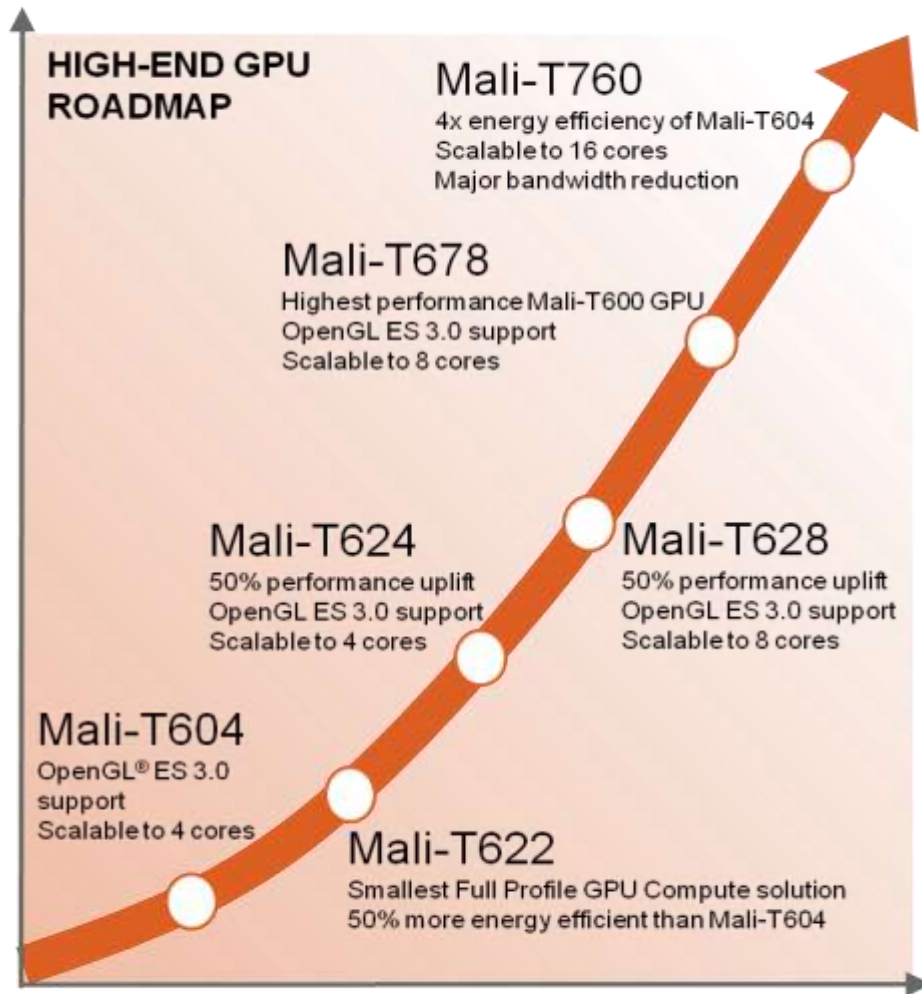
5.6.2 Integrating ARM's next generation Mali-G71 GPU (7)

Main Mali models based on the Utgard and Mitgard architecture [59]

Utgard architectures



Midgard architectures



➡ UTGARD ARCHITECTURE
➡ MIDGARD ARCHITECTURE

Specific issues of 2. and 3. generation Mali GPUs

Subsequently, we will discuss the following issues of 2. and 3. generation Mali GPUs:

- a) Arithmetic processing on 2. generation (Midgard) GPUs
- b) Arithmetic processing on 3. generation (Biforce) GPUs
- c) Vulkan graphics on 2. (Midgard) and 3. (Biforce) generation GPUs
- d) Clause execution on 3. generation (Biforce) GPUs

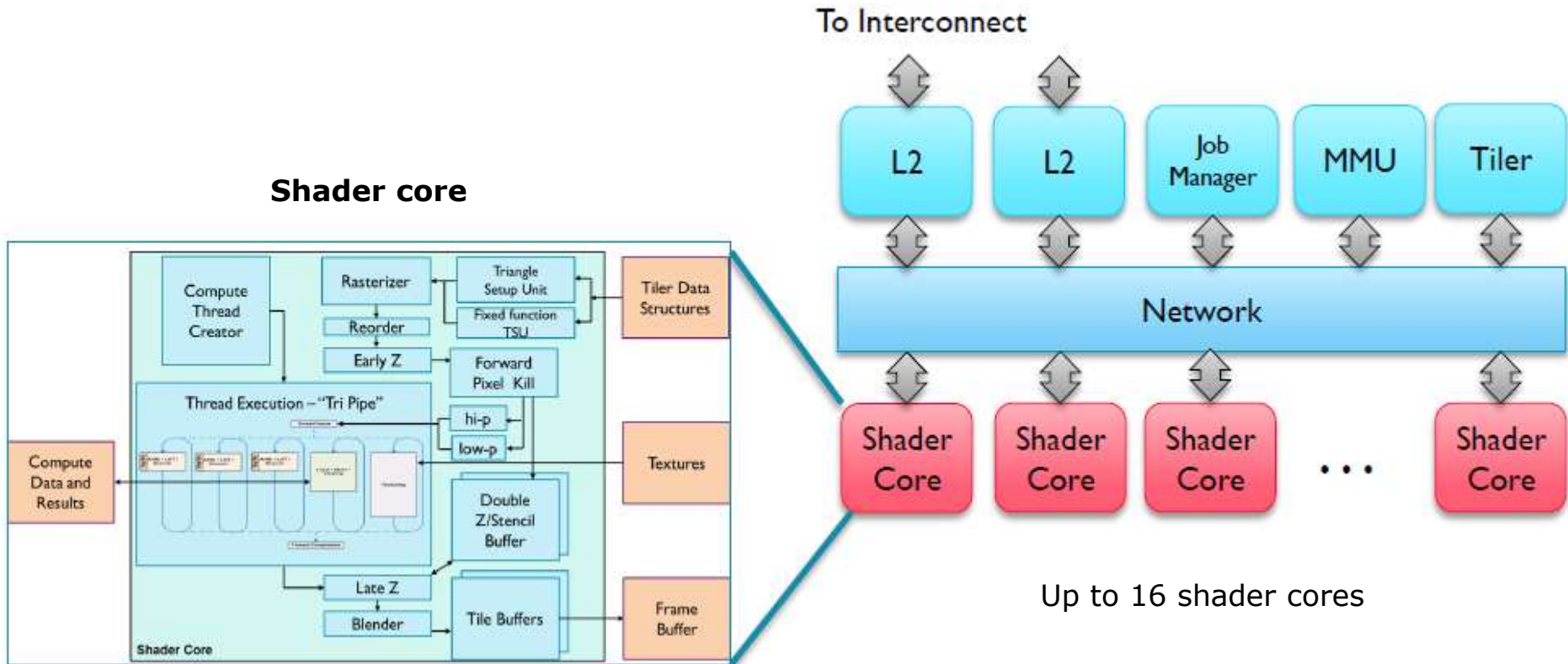
a) Arithmetic processing on 2. generation (Midgard) GPUs [35]

Beginning with the 2. gen. (Midgard) GPUs the Mali line supports running computing workloads as well as workloads written in OpenCL.

5.6.2 Integrating ARM's next generation Mali-G71 GPU (10)

Arithmetic processing on 2. generation (Midgard) GPUs [35]

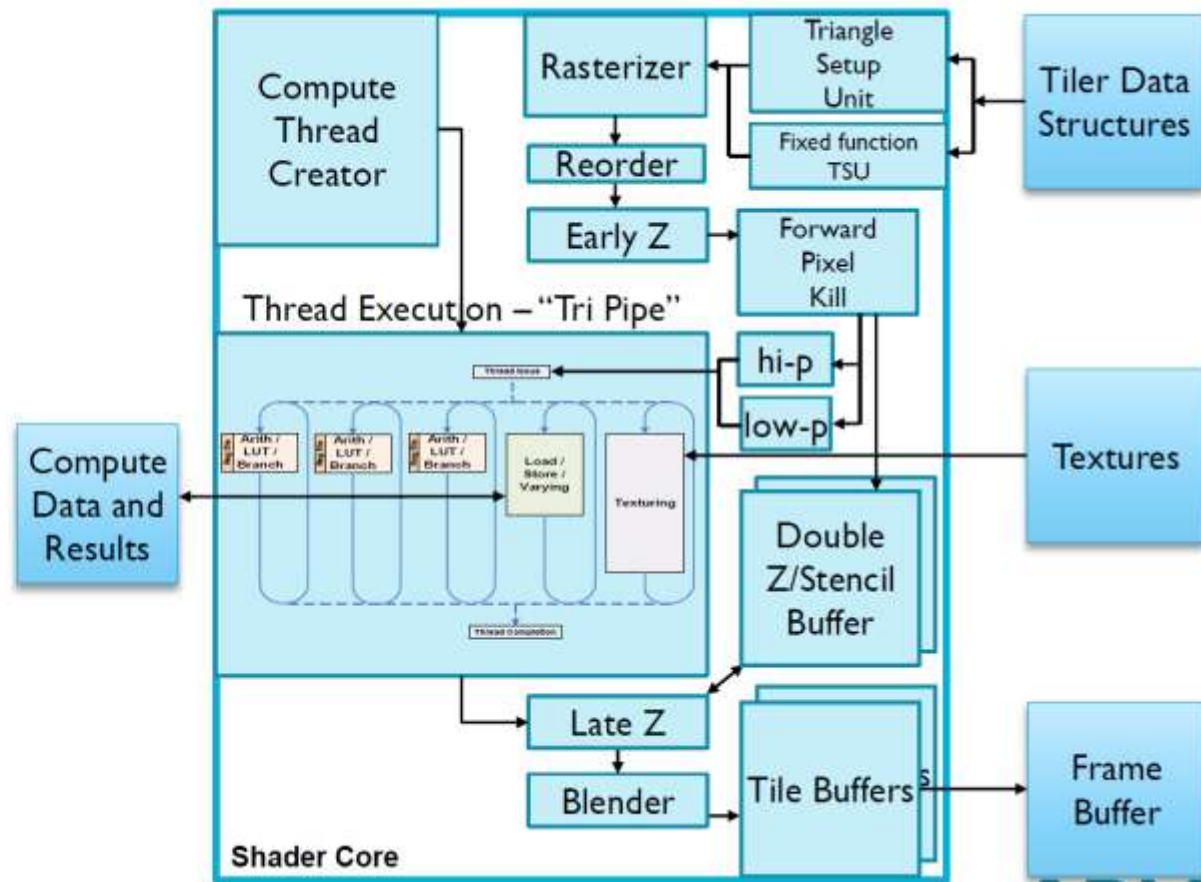
Example block diagram of a 2. gen. (Midgard) GPU (Mali-T880) [35]



5.6.2 Integrating ARM's next generation Mali-G71 GPU (11)

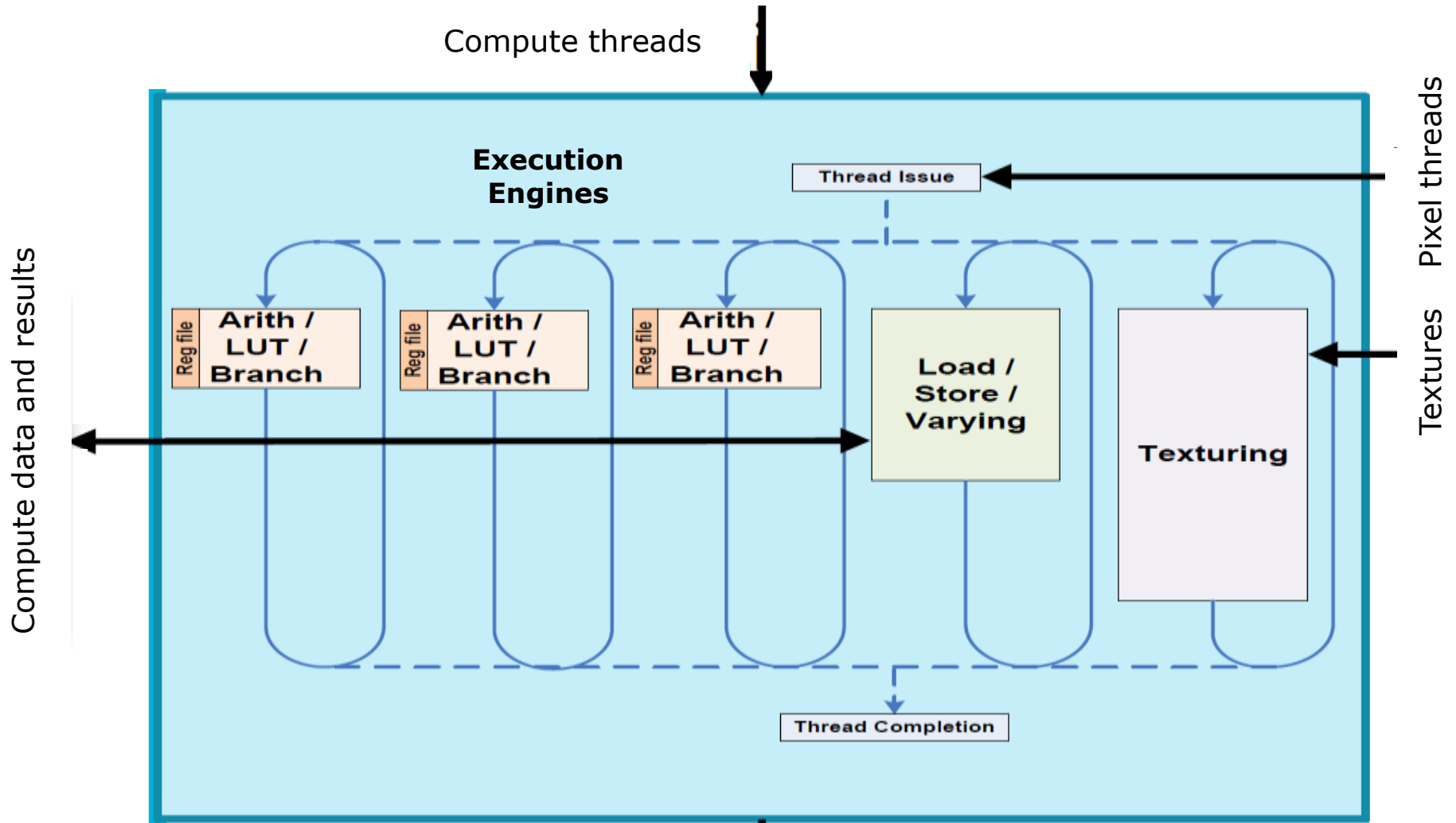
Example: Shader core a 2. gen. (Midgard) GPU (Mali-T880) [35]

- Collection of programmable and fixed function blocks
- Support for simultaneous execution of vertex and fragment jobs
- Programmable block is called the "Tri Pipe".
- All data (textures, render targets, descriptors, etc) is accessed via the cache subsystem.



5.6.2 Integrating ARM's next generation Mali-G71 GPU (12)

Thread execution in a shader core of the Mali-T880 [35]



5.6.2 Integrating ARM's next generation Mali-G71 GPU (13)

Number of Execution Engines in the ARM Mali 2. gen (Midgard) GPUs [36]

GPU model	No. of Execution Engines
T628	2
T678	4
T720	1
T760	2
T880	3

5.6.2 Integrating ARM's next generation Mali-G71 GPU (14)

Layout of an Execution Engine (called Arithmetic Pipe) in 2. gen. (Midgard) Mali GPUs [35], [36]

Each Execution Engine incorporates

- three vector units (VMUL, VADD, V_SPU), these are 4x FP32 SIMD units and
- two scalar units (SADD, SMUL), these are 1x FP32 wide,

as indicated below.

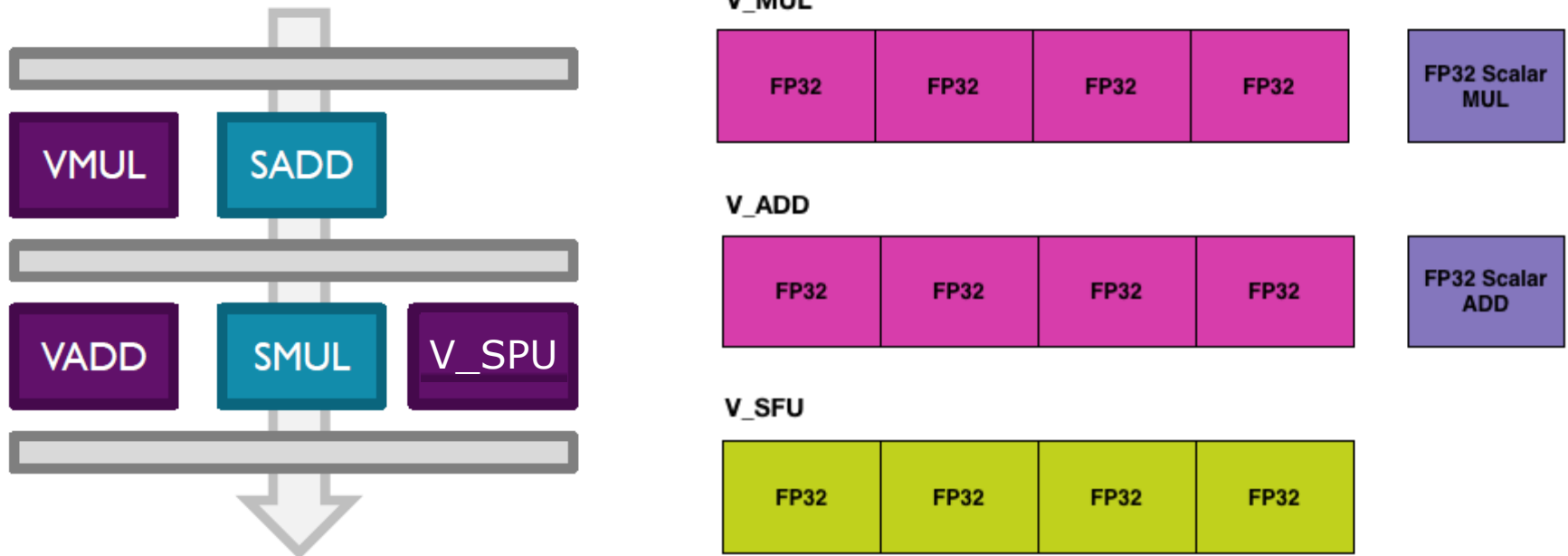
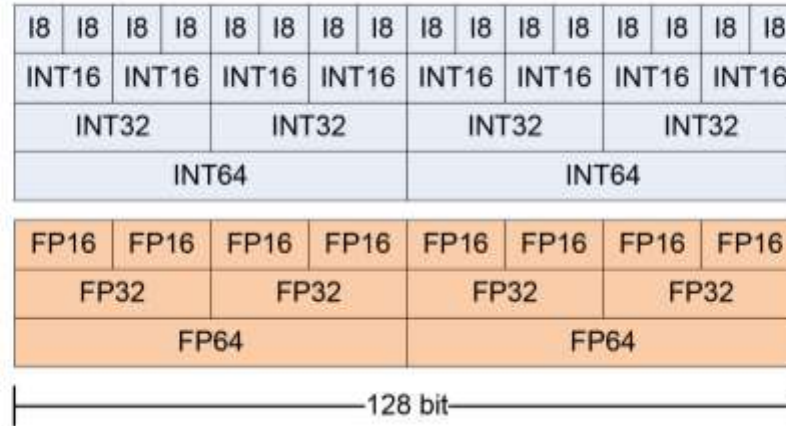


Figure: Layout of an Execution Engine of a 2. gen. (Midgard) Mali GPU [35], [36]

Note that the VMUL, VADD units perform in two cycles MADD operations.

Compute capabilities of the VMUL and VADD units [36]



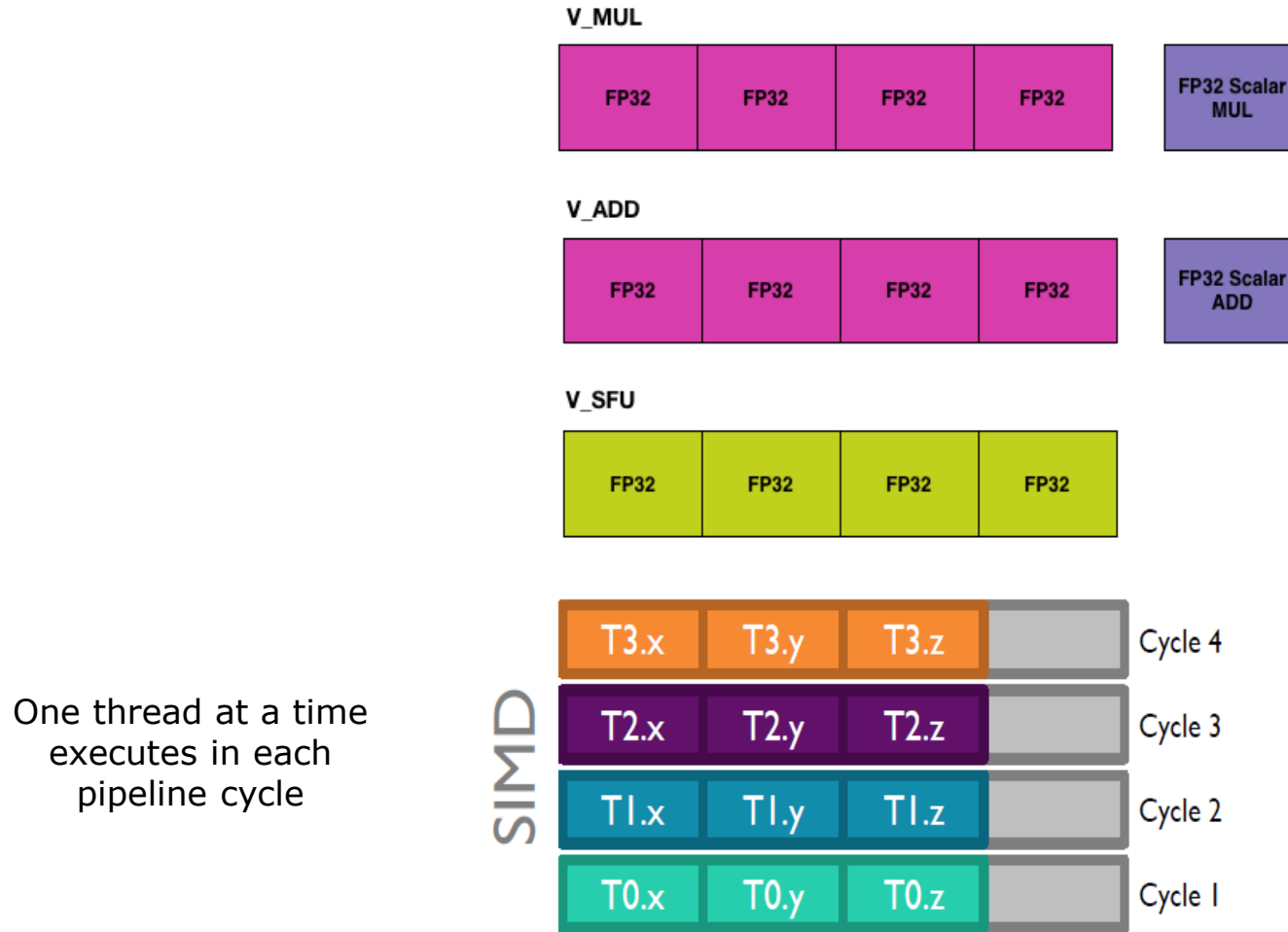
- Optimized for chain calculations
 - Vector/scalar units in parallel
 - Vector/scalar pairs in series

Peak FP32 rate per Execution Engine per cycle:

$$2 \times 4 \times \text{FP32} + 2 \times \text{FP32} + 7 \times \text{FP32 (VFSU)} = 17 \text{ FP32 / Execution Engine per cycle}$$

5.6.2 Integrating ARM's next generation Mali-G71 GPU (16)

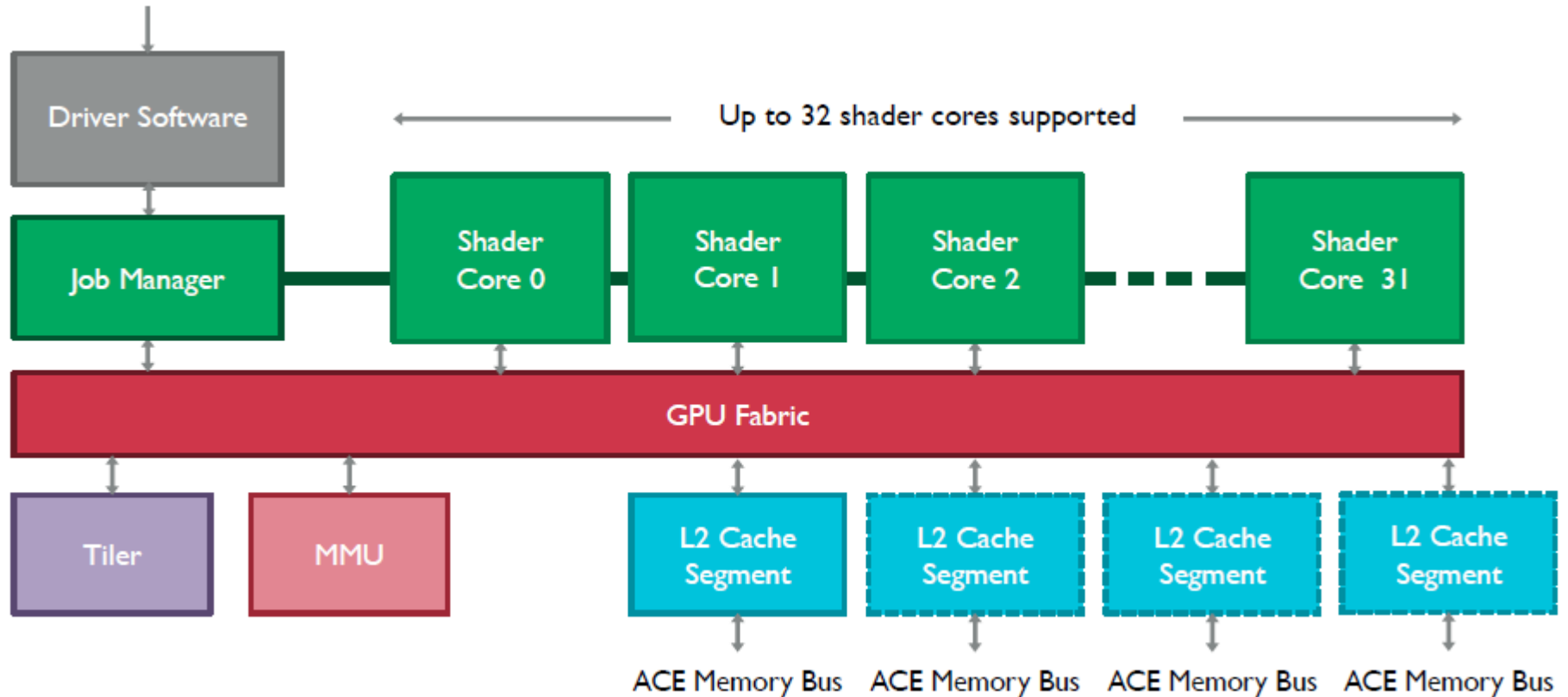
Thread execution model on an Execution Engine [35], [36]



5.6.2 Integrating ARM's next generation Mali-G71 GPU (17)

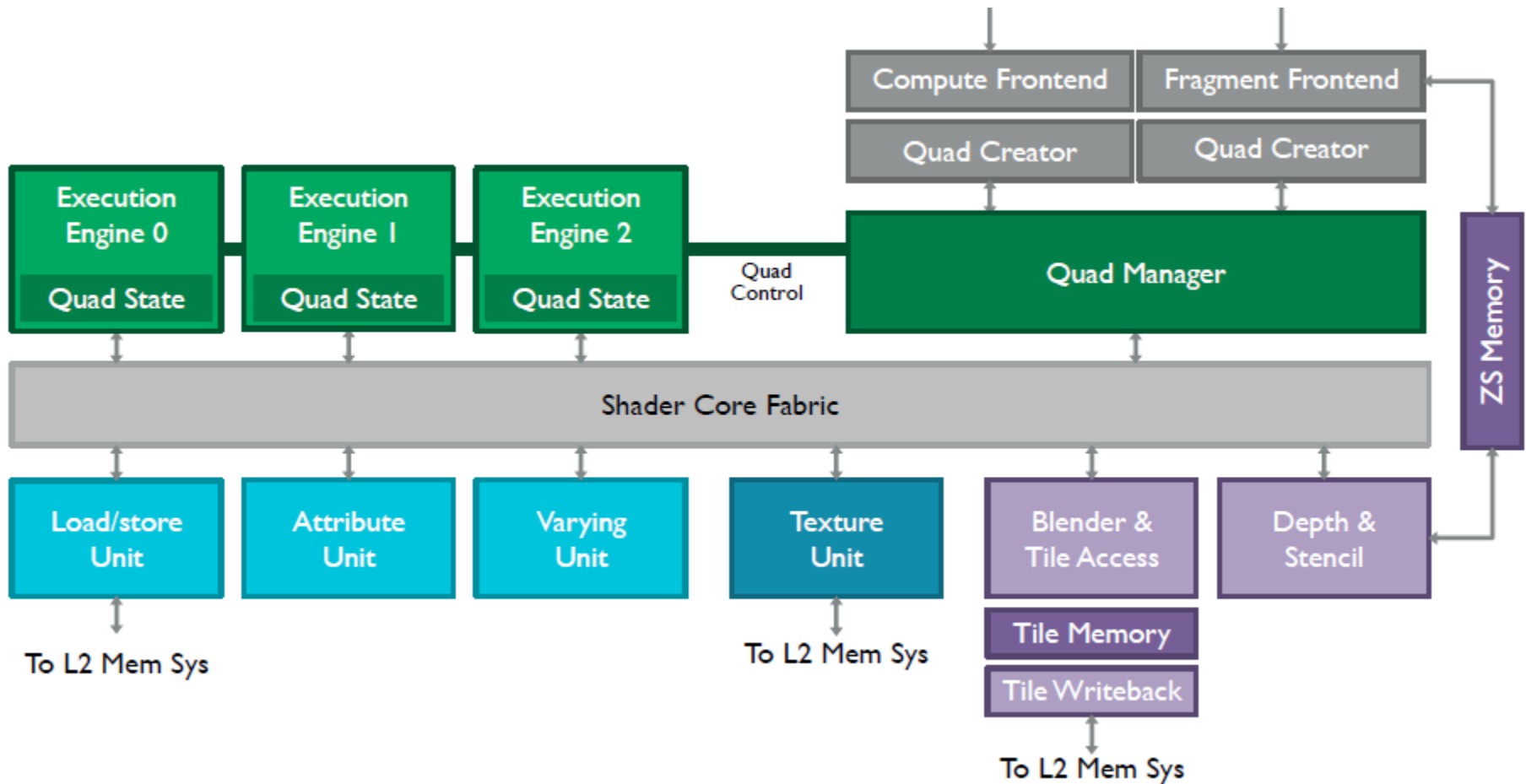
b) Arithmetic processing on 3. generation (Biforce) GPUs [33]

Example block diagram of a 3. gen. (Biforce) GPU [33]



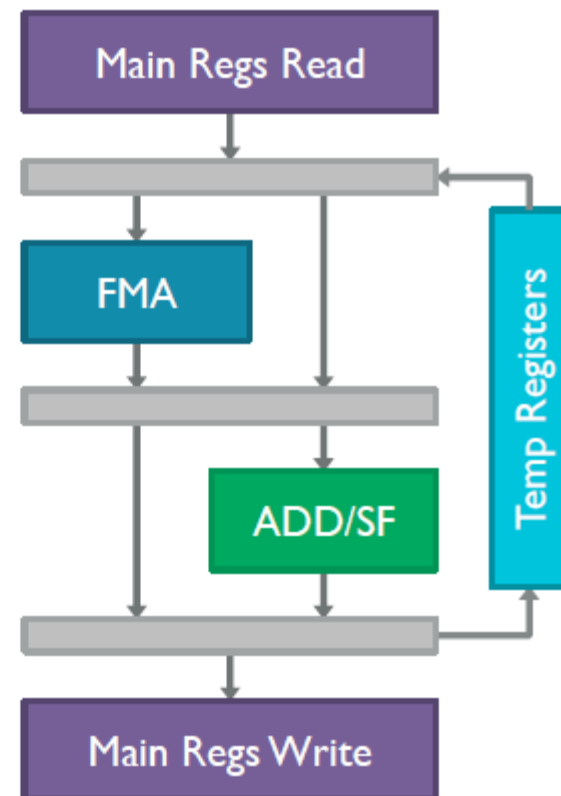
5.6.2 Integrating ARM's next generation Mali-G71 GPU (18)

Mali-G71 shader core design [33]



3. gen. (Bifrost) Execution Engine [33]

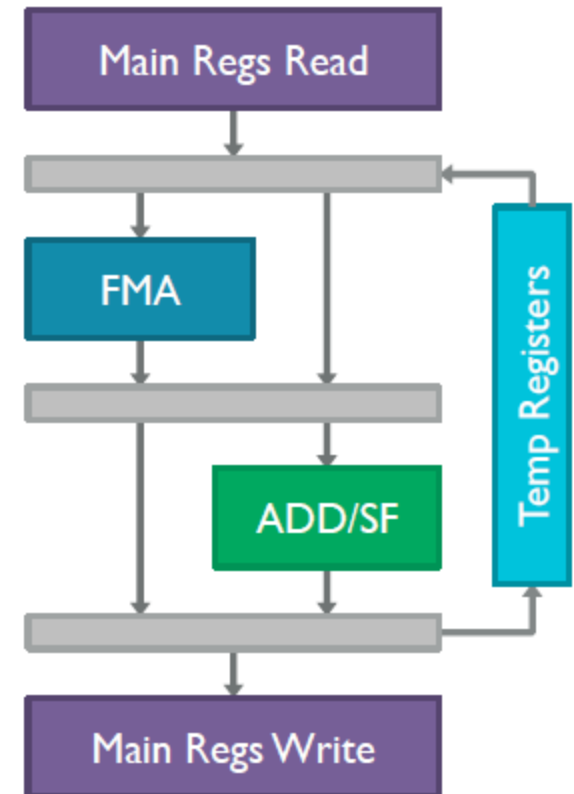
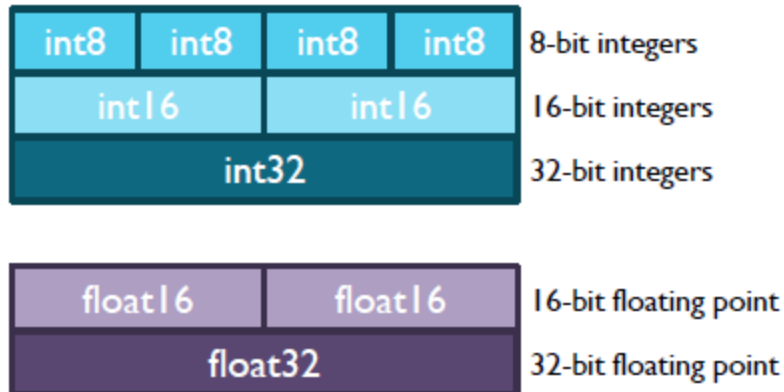
- Executes quad-parallel scalar operations
 - 4x32-bit multiplier FMA
 - 4x32-bit adder ADD
 - Adder includes special function unit
- Smaller and more area efficient
- Simplified layout eases compilation
 - Better scheduling in today's code
 - Better utilization
- One instruction word contains two instructions



5.6.2 Integrating ARM's next generation Mali-G71 GPU (20)

The FMA functional unit of the 3. gen. (Bifrost) Execution Engine [33]

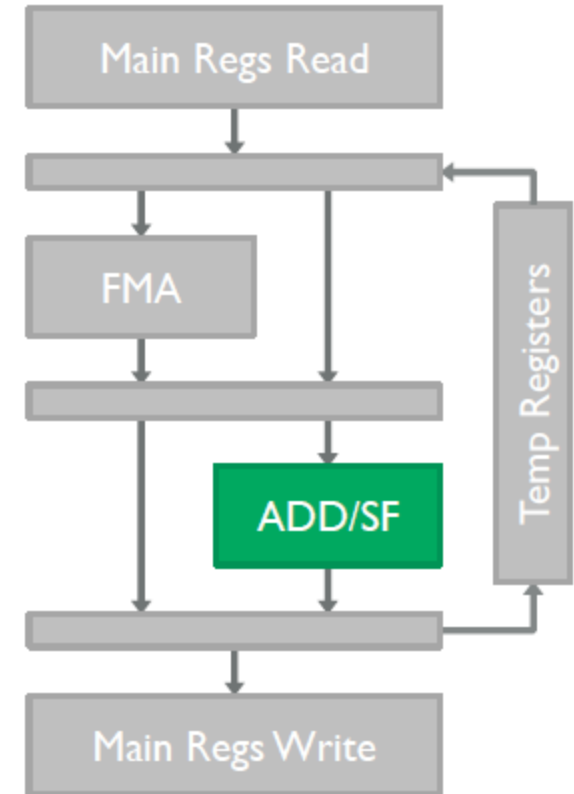
- Retains support for smaller width data types
 - 2x performance for FP16 useful for pixel shaders



5.6.2 Integrating ARM's next generation Mali-G71 GPU (21)

The ADD/SF functional unit of the 3. gen. (Bifrost) Execution Engine [33]

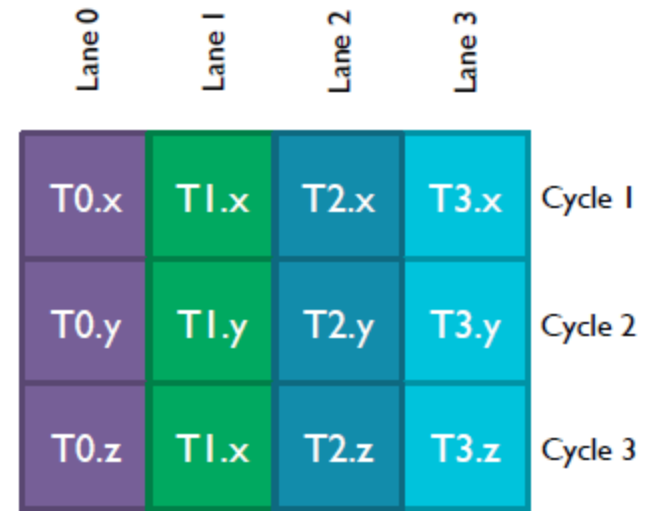
- Special function hardware is smaller than Midgard's equivalent
 - Many transcendental functions supported
 - Special functions provide building blocks for compiled shader code



Thread execution model on an Execution Engine [33]

Quad vectorization

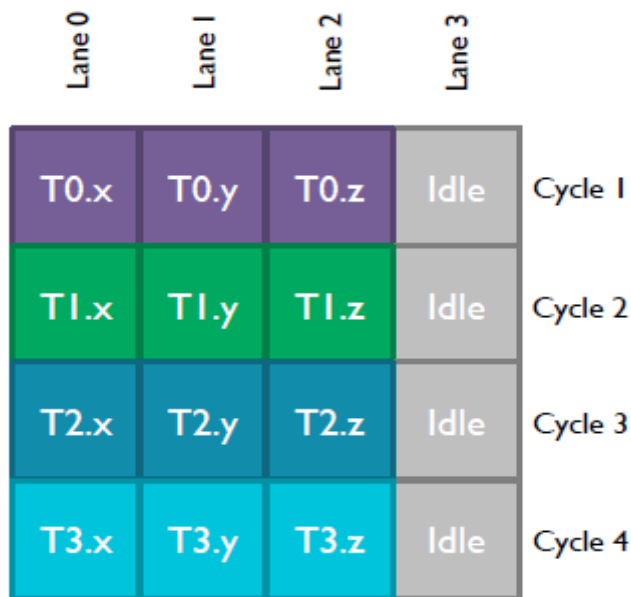
- Bifrost uses quad-parallel execution
 - Four scalar threads executed in lockstep in a “quad”
 - One quad at a time executes in each pipeline stage
 - Each thread fills one 32-bit lane of the hardware
 - 4 threads doing a vec3 FP32 add takes 3 cycles
 - Improves utilization
- Quad vectorization is compiler friendly
 - Each thread only sees a stream of scalar operations
 - Vector operations can *always* be split into scalars



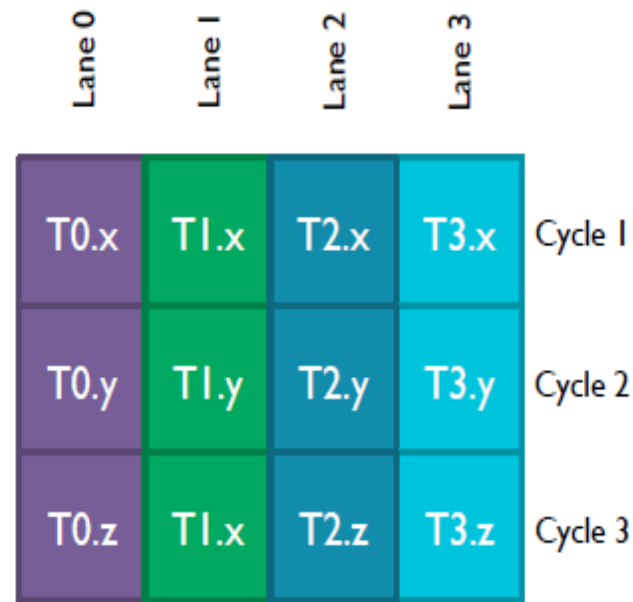
5.6.2 Integrating ARM's next generation Mali-G71 GPU (23)

Contrasting the thread execution models of 2. and 3. gen. Mali GPUs [33]

SIMD vectorization (Midgard)



Quad vectorization (Bifrost)



Midgard GPUs use SIMD vectorization

- One thread at a time executes in each pipeline stage
- Each thread must fill the width of the hardware

Sensitive to shader code

- Code always evolving
- Compiler vectorization is not perfect

Bifrost GPUs use quad-parallel execution

- Four scalar threads executed in lockstep in a “quad”
- One quad at a time executes in each pipeline stage
- Each thread fills one 32-bit lane of the hardware
- 4 threads doing a vec3 FP32 add takes 3 cycles
- Improves utilization

Quad vectorization is compiler friendly

- Each thread only sees a stream of scalar operations

Remark

Also AMD switched from VLIW4 SIMD vectorization to QUAD vectorization in their GCN (Graphics Core Next) graphics computing architecture in 2011 [36]

Vector Units



VLIW4 SIMD

- 64 Single Precision multiply-add
- 1 VLIW Instruction \times 4 ALU ops \rightarrow dependency limited
- Compiler manages register port conflicts
- Specialized, complex compiler scheduling
- Difficult assembly creation, analysis, and debug
- Suited for graphics, less flexible for compute
- Careful optimization required for peak performance

GCN Quad SIMD

- 64 Single Precision multiply-add
- 4 SIMDs \times 1 ALU op \rightarrow occupancy limited
- No register port conflicts
- Standardized compiler scheduling & optimizations
- Simplified assembly creation, analysis, and debug
- Simplified tool chain development and support
- Stable and predictable performance

c) Vulkan graphics on 2. (Midgard) and 3. (Biforce) generation GPUs [37]

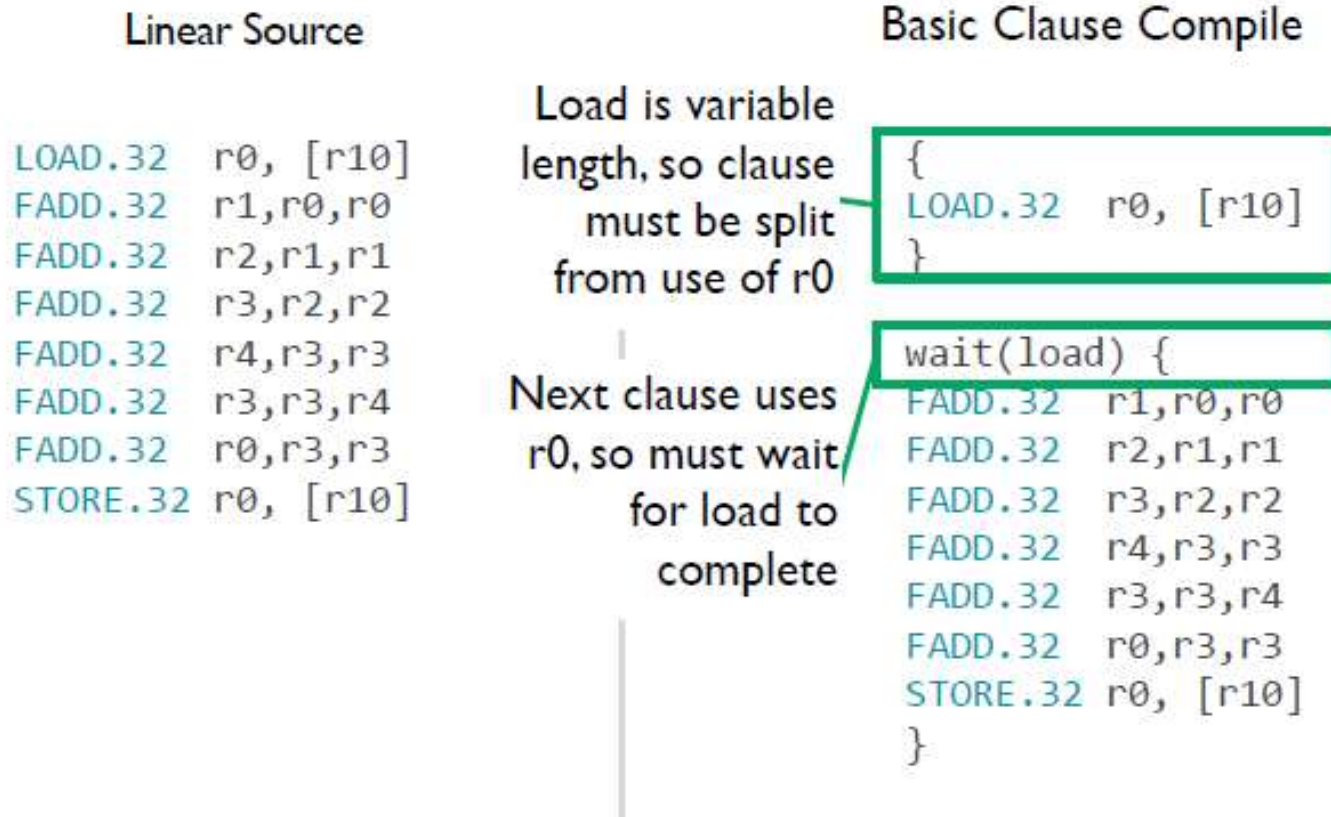
- **Vulkan** is a new generation graphics and compute API that provides high-efficiency, cross-platform access to up-to-date GPUs used in a wide variety of devices from PCs and consoles to mobile phones and embedded platforms.
- Khronos launched the Vulkan 1.0 specification in February 2016 and Khronos members, like ARM, Intel, NVIDIA, released Vulkan drivers and SDKs immediately.

d) Clause execution on 3. generation (Biforce) gPUs

- **Clause**: a group of instructions which **executes atomically**.
- **Architectural state visible after clause completion**.

5.6.2 Integrating ARM's next generation Mali-G71 GPU (27)

Example for clause execution in 3. gen. (Bifrost) GPUs [33]



5.6.2 Integrating ARM's next generation Mali-G71 GPU (28)

Benefits of 3. gen. (Bifrost) GPUs vs. 2. gen. (Midgard) GPUs [33]

- Leverages Mali's scalable architecture
 - Scalable to 32 shader cores
- Major shader core redesign
 - New scalar, clause-based ISA
 - New quad-based arithmetic units
- New geometry data flow
 - Reduces memory bandwidth and footprint
- Support for fine grain buffer sharing with the CPU



20%
Higher energy
efficiency*



Scalable to 32
Shader cores



20%
Bandwidth
Improvement*



40%
Better
performance
density*

*Compared to Mali-T880 on same process node under the same conditions. **ARM**

5.6.3 HSA (Heterogeneous System Architecture) compliance

5.6.3 HSA (Heterogeneous System Architecture) compliance

The road towards HSA (Heterogeneous System Architecture)

The vision of Si-level integration of CPU and GPU cores

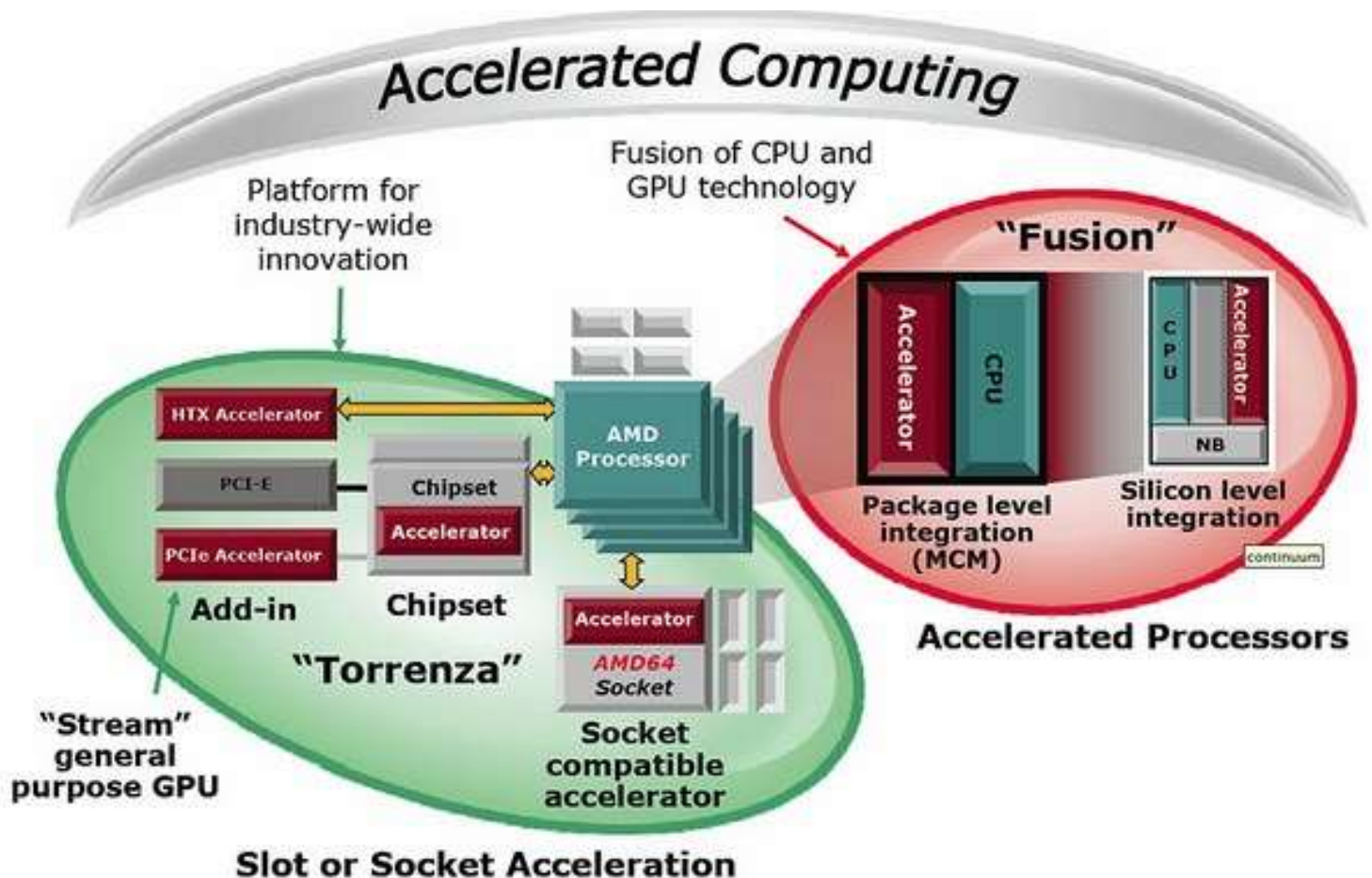
- The roots of HSA are going back to 2006 when AMD revealed their plan to integrate CPU cores and the GPU on the same silicon die, called the CPU/GPU Silicon Fusion [60].
- At that time AMD planned to introduce their Fusion processors in late 2008 or early 2009.
- For supporting their intention AMD acquired ATI, a successful graphics firm, in 10/2006.
- We note that ARM acquired a small, Norwegian graphics firm at the same time that became the core of ARM's graphics division and developed the Mali GPU line.

5.6.3 HSA (Heterogeneous System Architecture) compliance (2)

Enhancing AMD's Fusion concept to Accelerated Computing [38]

Accelerated computing widens the concept of Si-level integration to the integration of CPU cores and accelerators in 03/2007.

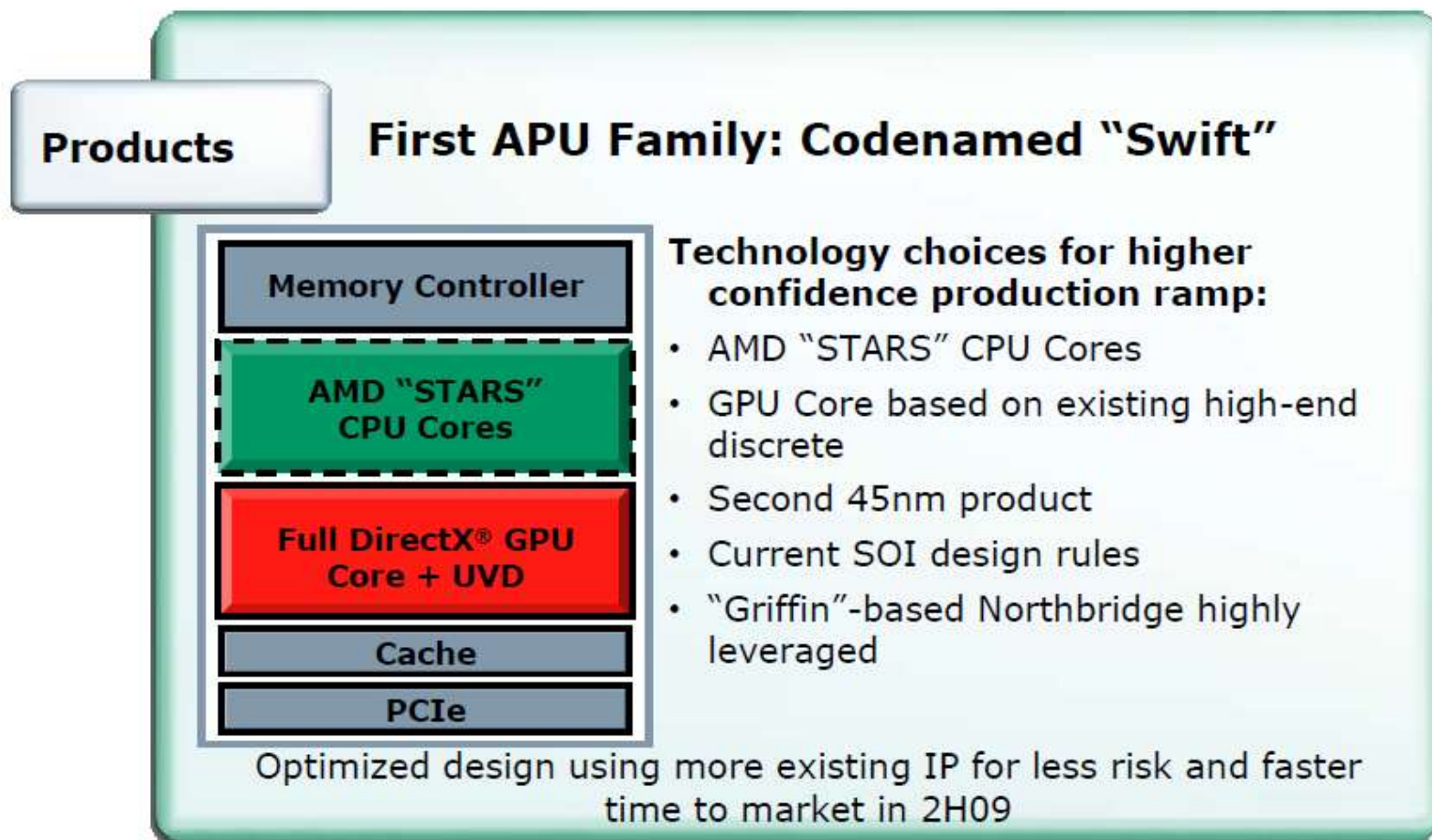
Here GPUs are considered as a specific type of accelerators (graphics accelerators).



5.6.3 HSA (Heterogeneous System Architecture) compliance (3)

AMD's aim to introduce the first APU family called Swift in 2H/2009 [39]

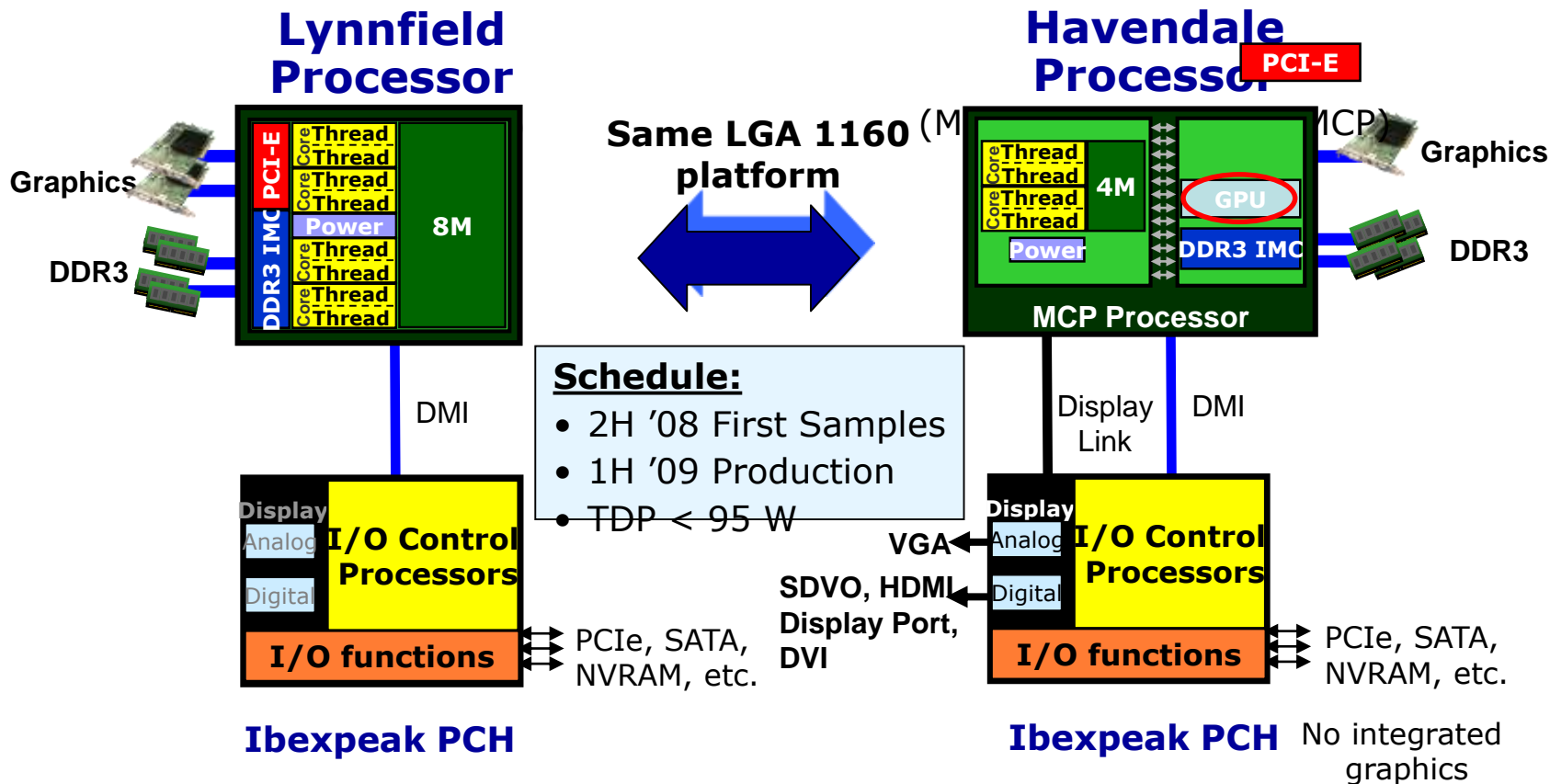
- In 12/2007 at their Financial Analyst Day AMD gave birth to a new term **APU (Accelerated Processing Unit)** designating their processors implementing the Fusion concept).
- At the same time AMD announced their first APU family, the **Swift family** [39] as well



5.6.3 HSA (Heterogeneous System Architecture) compliance (4)

Intel's aim to introduce in-package integrated graphics in 1H/2009

In 09/2007 Intel announced an **in-package integrated GPU** that is an alternative of the 2. gen. Nehalem (Lynnfield) processor, as indicated below.



Lynnfield & Havendale can be supported on one platform

5.6.3 HSA (Heterogeneous System Architecture) compliance (5)

Cancellation of both AMD's and Intel's GPU integration plans in 11/2008 and 01/2009 respectively.

Both firms postponed their plans to integrate GPUs in the 45 nm technology until the 32 nm technology with a higher transistor budget becomes available [40], [41].

5.6.3 HSA (Heterogeneous System Architecture) compliance (6)

Introducing in package and on-die integrated graphics by Intel and AMD

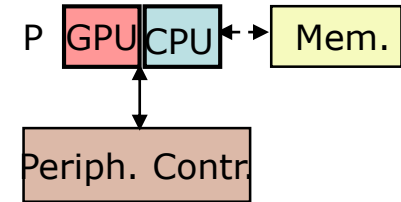
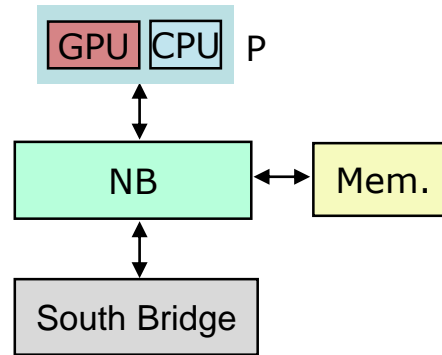
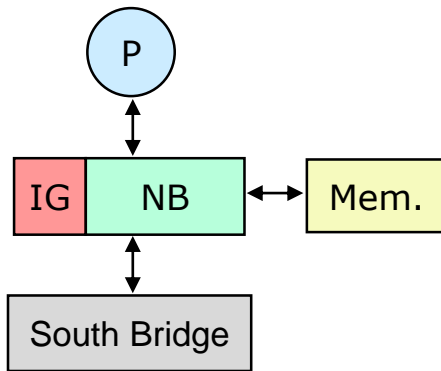
Implementing integrated graphics

In the north bridge

**In the processor package
(as an MCP on a separate die)**

On the processor die

Both the CPU and the GPU
are on separate dies
and are mounted into a single package



Implementations around
1999 - 2009

Intel's Havendale (DT) and
Auburndale (M)
(scheduled for 1H/2009
but cancelled in 01/2009)
Arrandale (DT, 1/2010) and
Clarkdale (M, 1/2010)

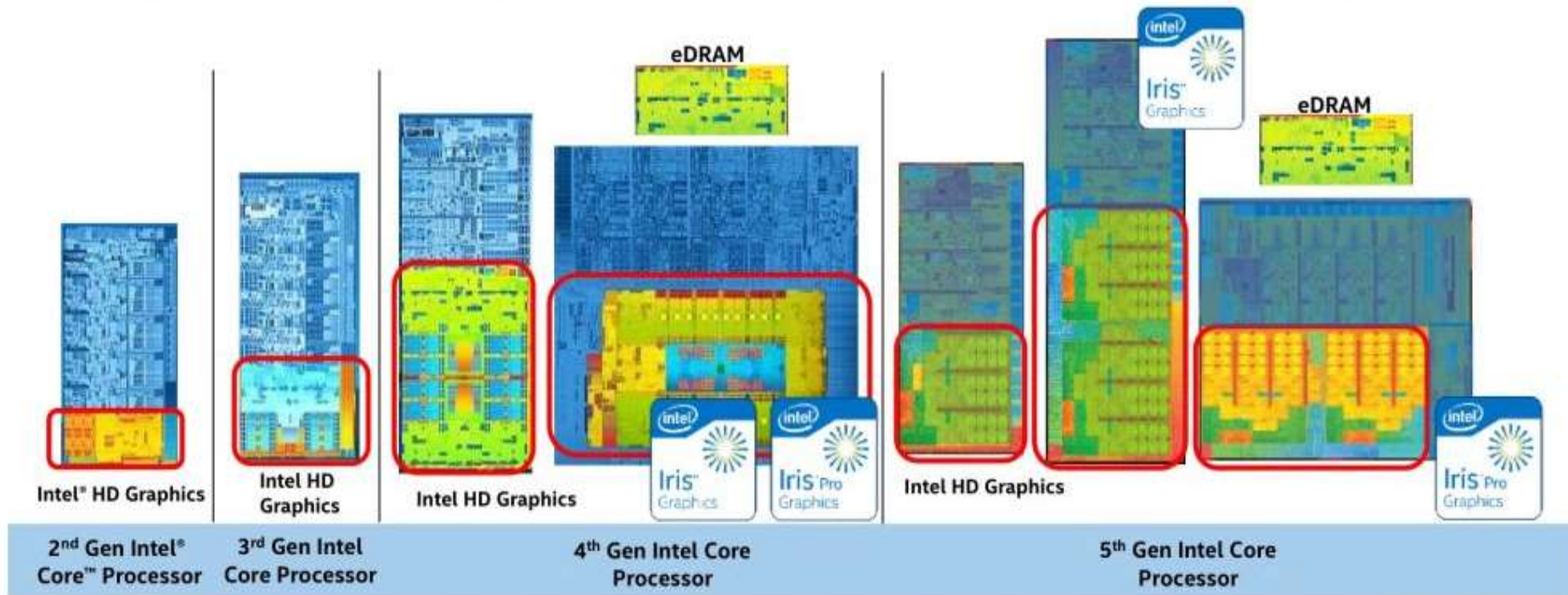
Intel's Sandy Bridge (1/2011)
AMD's Bobcat-based APUs (M, 1/2011)
and Llano APUs (DT, 6/2011)

MCP: Multi-Chip Package
M: Mobile **DT:** Desktop



5.6.3 HSA (Heterogeneous System Architecture) compliance (7)

Evolution of integrated graphics in Intel's processor families [42]



5.6.3 HSA (Heterogeneous System Architecture) compliance (8)

Introducing the concept of HSA "Heterogeneous Systems Architecture" by AMD

In 01/2012 AMD rebranded their FSA (Fusion System Architecture) term to **HSA (Heterogeneous System Architecture)** [43] and in 02/2012 introduced the HSA concept that designates an efficient open ecosystem for accelerated processors (called APUs by AMD), as indicated below [44].

HETEROGENEOUS SYSTEM ARCHITECTURE

A KEY ENABLER TO OUR APU VALUE PROPOSITION

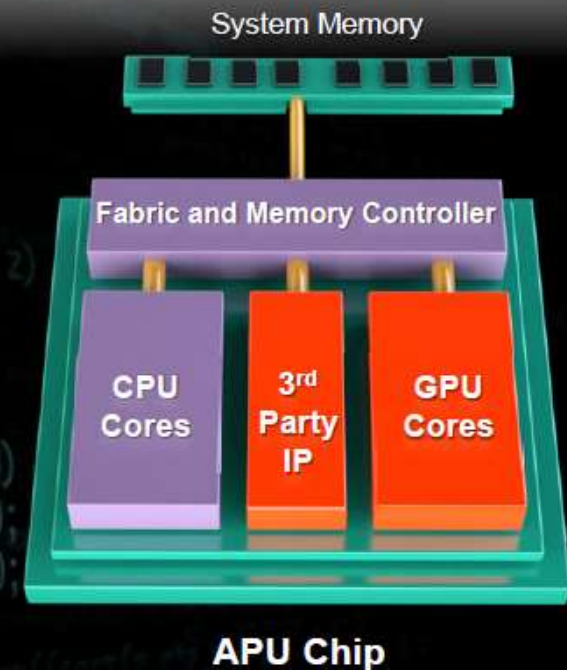


HSA is an enabler for APU efficiency and differentiation

- Unleash our industry leading GPU cores on a broad range of applications beyond graphics
- CPU and GPU work cooperatively together directly in system memory
- Makes programming the GPU as easy as C++
- Up-to 125%* OpenCL benchmark advantage vs. competition

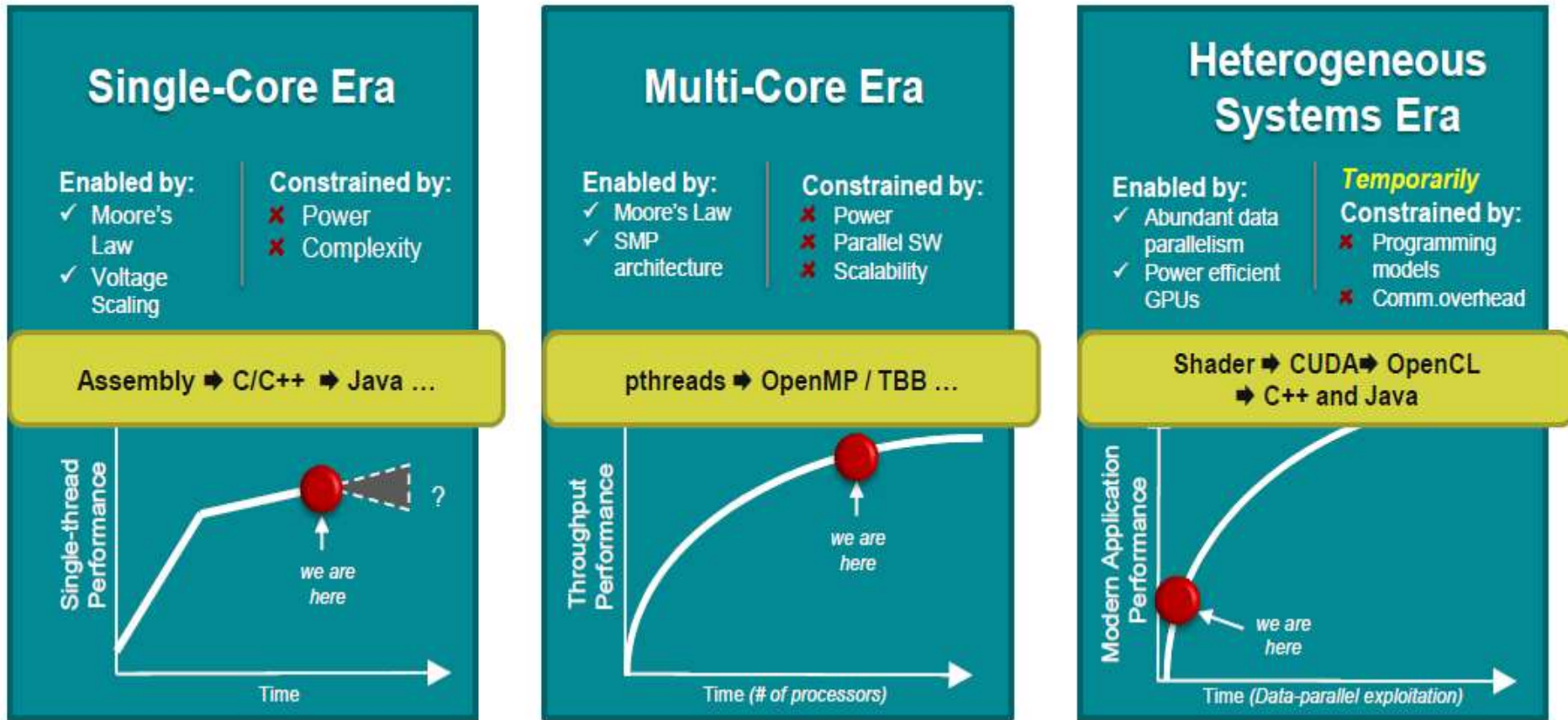
Key value propositions

- Lower power for modern applications!
- Easy for application developers to use
- Drives new class of applications
 - e.g., analytics, search, facial recognition



5.6.3 HSA (Heterogeneous System Architecture) compliance (9)

HSA's role in processor evolution [45]



TBB (Threading Building Blocks): is a C++ template library, developed by Intel for parallel programming on multi-core processors.

OpenMP (Open Multi-Processing) is an application programming interface (API) that supports shared memory multiprocessing programming in C, C++, and Fortran on most platforms, instruction set architectures and operating systems, including Solaris, AIX, HP-UX, Linux, and Windows (Wikipedia).

5.6.3 HSA (Heterogeneous System Architecture) compliance (10)

Announcing the HSA Foundation in 06/2012

At the initiative of AMD the **HSA Foundation** was established in 06/2012.

The founding members are: see below. [46]



Note that neither Intel nor NVIDIA are among the foundation members.

5.6.3 HSA (Heterogeneous System Architecture) compliance (11)

Evolution of the HSA standard [47]

- **HSA 1.0: 03/2015**

It incorporates **changes for efficient implementation of high-level languages**, such as C++, Java and Python on heterogeneous computing hardware and a more detailed documentation.

- **HSA 1.1: 05/2016**

- Backward compatible with HSA 1.0
- **Multi-vendor oriented**, vendor neutral device drivers
- **Wider range of devices** supported, including DSPS, ISPs (Image Processors)

5.6.3 HSA (Heterogeneous System Architecture) compliance (12)

Other approaches to support programming of heterogeneous platforms

- **HSA** is a multi-vendor, open **standard** concept to efficiently support heterogeneous platforms.
- By contrast, **both Intel and NVIDIA** choose a **proprietary solution** for this.
- **NVIDIA** developed their **CUDA language and the PTX virtual ISA** for heterogeneous systems, built up of either of CPU and GPU cards or SOCs with on-chip integrated CPU and GPU, like the TEGRA K1 (Q2/2014), TEGRA X1 (Q2/2015) SOCS.
- **Intel's solution** is to provide **unified virtual memory**, called **SVM** (shared Virtual Memory) and **falling back to OpenCL 2.0** in supporting HLL programming.

Specific features and requirements of HSA -1

- **HSA** relates to a **heterogeneous system architecture** consisting typically of CPU cores, a GPU and accelerators and it takes for granted that **each code segment runs on a unit that provides the fastest and most efficient execution**, so e.g. serial, branch intensive code will run on CPU cores whereas data parallel code on the GPU, as indicated below..

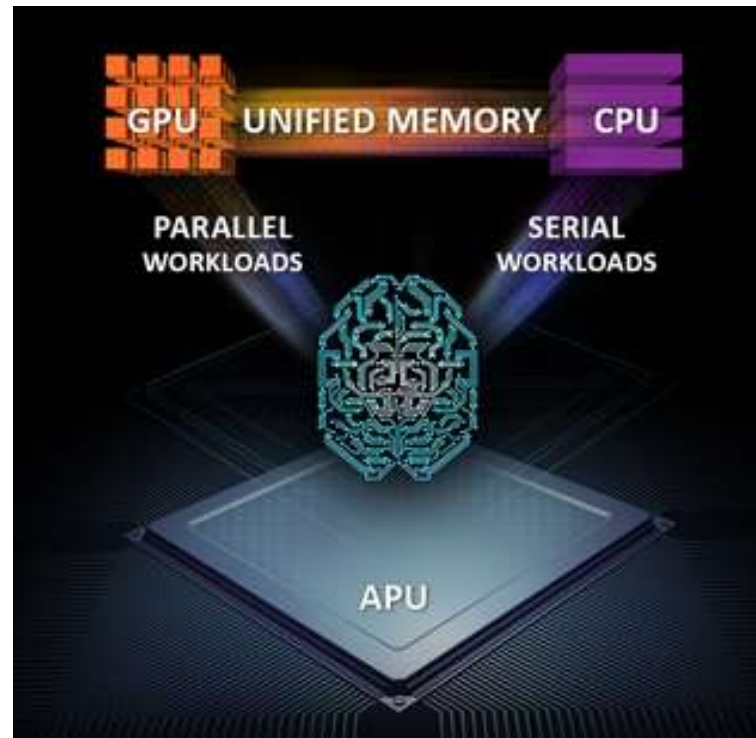


Figure: Running serial workloads on CPU cores and parallel workload on a GPU in HSA
[48]

Specific features and requirements of HSA -2

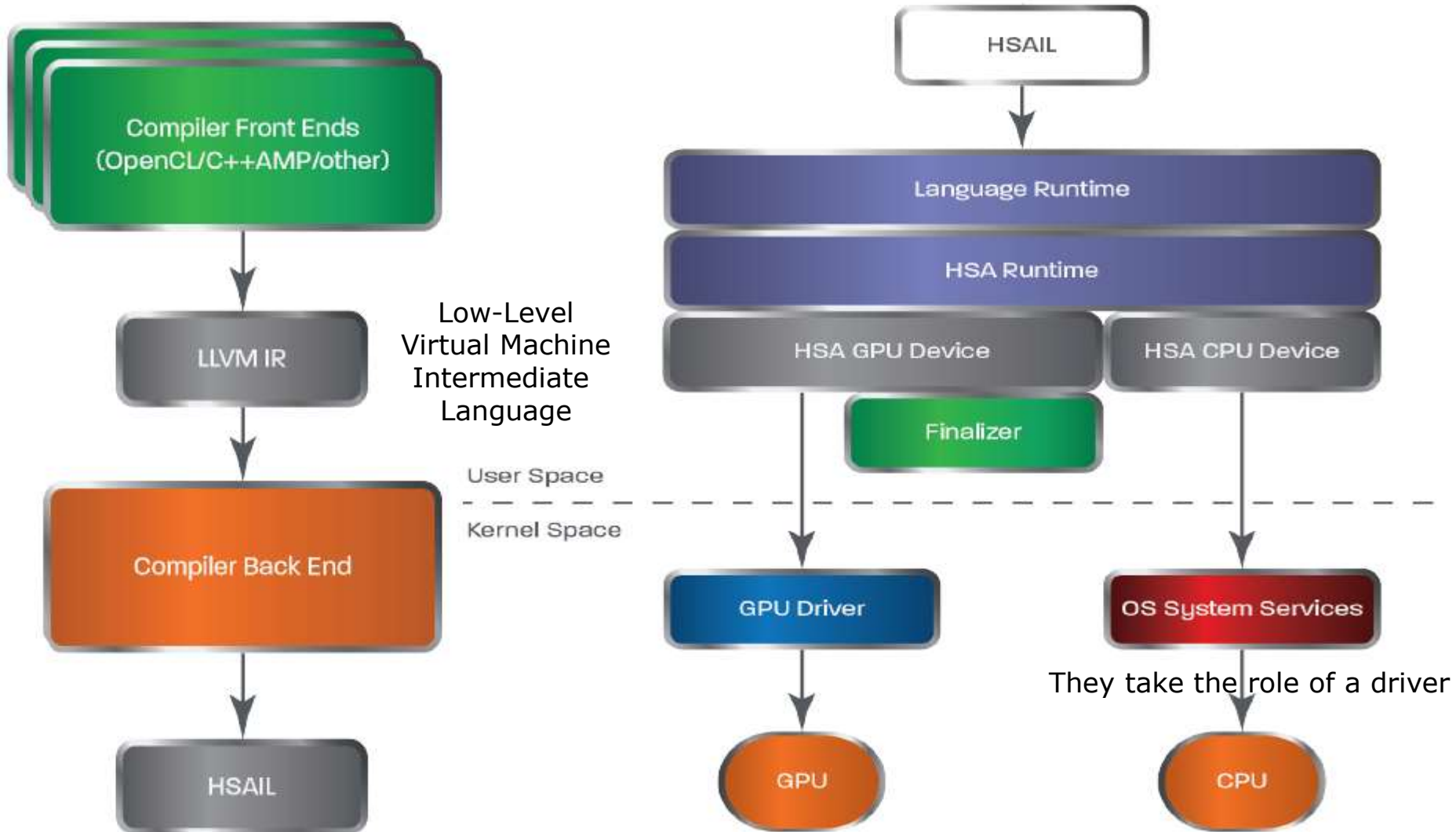
- An **efficient, vendor agnostic processing ecosystem** for heterogeneous platforms has a number of **requirements**, as presented e.g. in [49], [50].
- From these requirements subsequently, we will discuss the following **four major ones**:
 - b1) Using a vendor agnostic virtual ISA (called HSAIL) for implementing the targeted HSA processing ecosystem to foster industry acceptance,
 - b2) Providing HLL support for easy of coding of HSA platforms,
 - b3) Providing a cache coherent, uniform, shared virtual memory for all processing units to avoid inefficient data copying between different memory spaces and
 - b4) Providing an efficient mechanism for forwarding tasks between the CPU and the GPU to eliminate in-efficient OS interactions.

b1) Using a vendor agnostic virtual ISA (called HSAIL) for implementing the targeted HSA processing ecosystem to foster industry acceptance [45]

- **HSAIL** is a **virtual ISA** developed for data parallel programs.
HSAIL supports only compute workloads and does not support graphics-specific instructions.
- Source programs, e.g. in OpenCL 2.0 will be compiled to the HSAIL, and then interpreted (finalized) to the CPU or GPU ISA, as indicated below.
- HSAIL is based on a similar concept as e.g. Java bytecode.

5.6.3 HSA (Heterogeneous System Architecture) compliance (16)

Concept of the virtual ISA of HSA [49]



5.6.3 HSA (Heterogeneous System Architecture) compliance (17)

Remark 1

- NVIDIA has a similar virtual ISA for GPU computing, designated as **PTX** (Parallel Thread eXecution).
- NVIDIA supports programming heterogeneous (GPU) systems by the **CUDA HLL**.
- Then compiling of CUDA is a two stage process, as seen in the next Figure.

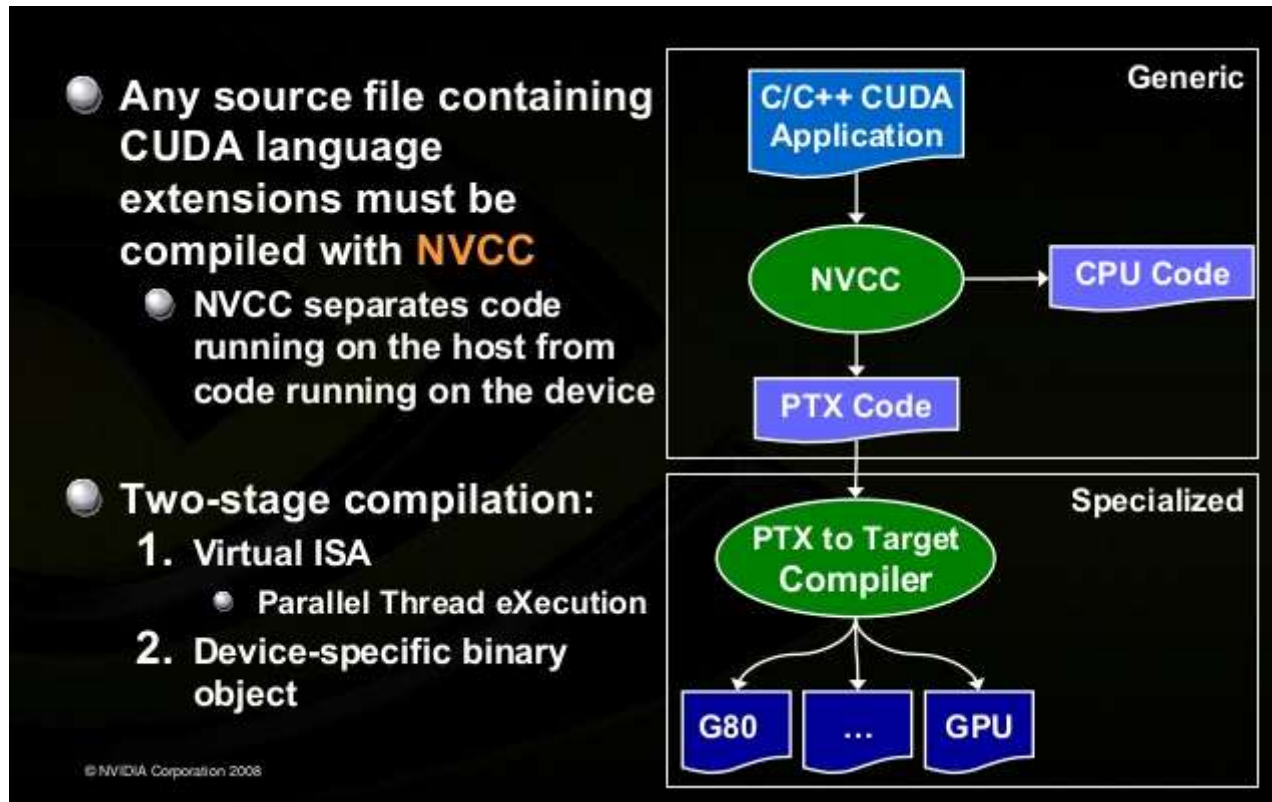
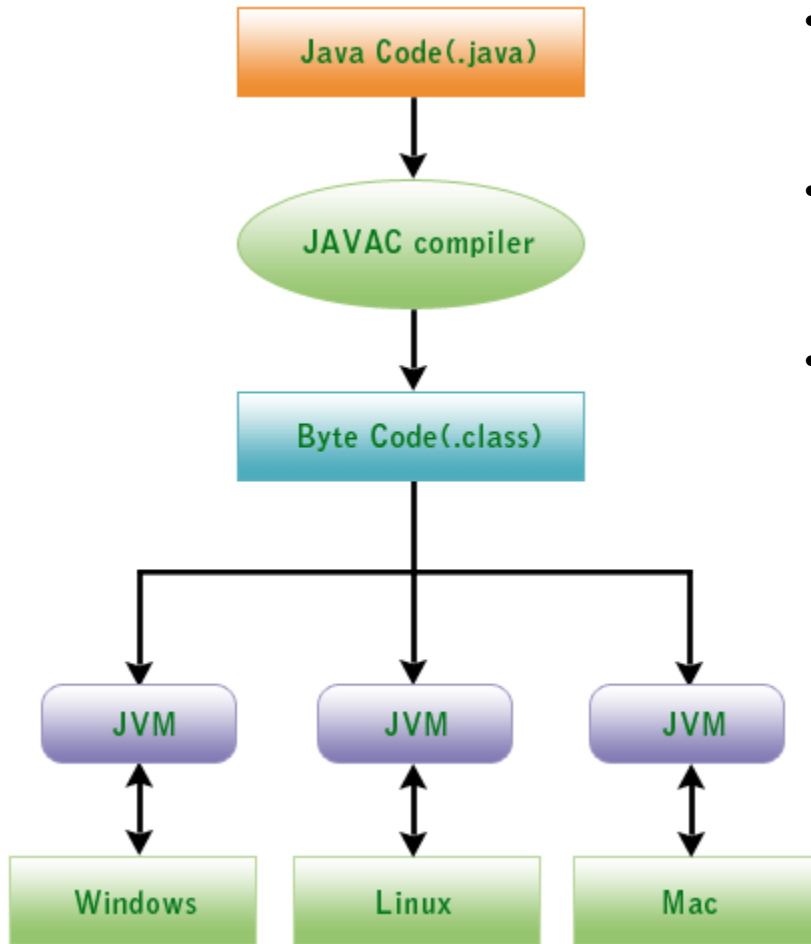


Figure: Compiling CUDA for NVIDIA GPUs [51]

5.6.3 HSA (Heterogeneous System Architecture) compliance (18)

Remark 2: Compiling and executing Java programs on different platforms



- The bytecode is processed by the **JVM** component of the **Java Runtime Environment (JRE)**.
- Each platform needs one or more JVMs that suits the OS and the target ISA, as indicated in the Figure.
- It is then the task of the JVM to execute the bytecode e.g. by interpreting each bytecode instructions.

JVM: Java Virtual Mashine

Figure: Different JVMs for different OSs and CPU ISAs [61]

5.6.3 HSA (Heterogeneous System Architecture) compliance (19)

b2) Providing HLL support for easy of coding of HSA platforms [50], [47]

- Obviously, developers for HSA are not expected to write code in HSAIL. Instead, HSA Foundation took care for the availability of HLLs for easy of programming.
- Major HLL languages available for HSA (04/2017):
 - OpenCL 2.0
 - C++ AMP 1.2 (C++ Accelerated Massive Parallelism) and
 - Python (Continuum Analytics' Numba Python compiler).
- It was planned also that Java 9 will support HSA by allowing to generate HSAIL directly from Java bytecode but to date it was not yet released.
- Here we note that rather than participating in the multi vendor HSA Consortium established for providing the needed programming ecosystem for heterogeneous systems NVIDIA followed an alternative, proprietary path with their CUDA language and PTX virtual ISA.

5.6.3 HSA (Heterogeneous System Architecture) compliance (20)

b3) Providing a cache coherent, uniform, shared virtual memory for all processing units to avoid inefficient data copying between different memory spaces

A memory with the said features is called **hUMA (HSA UMA or heterogeneous UMA)** by AMD.

A uniform memory provides the **same visibility** for all processing units (e.g. the CPU and the GPU) into the entire memory space (up to 32 GB), as indicated below.

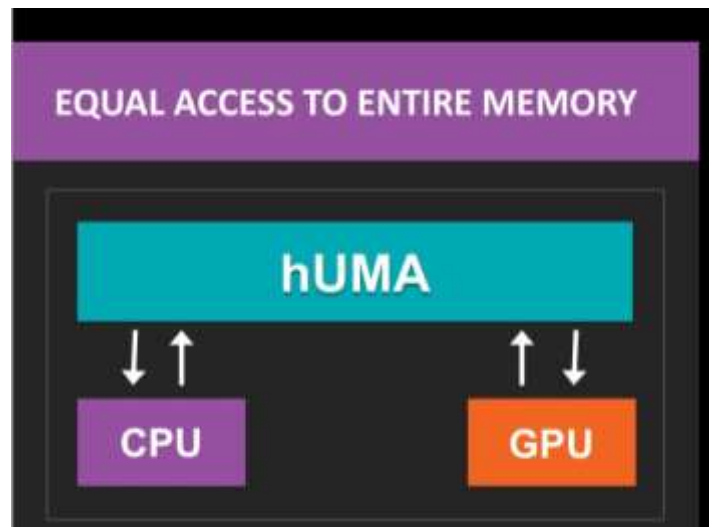


Figure: Principle of hUMA [62] providing the same visibility for both the CPU and the GPU into the entire memory space (up to 32 GB)

Uniform memory is implemented by **using the same mechanism** (e.g. page tables) to translate **virtual addresses to physical addresses** for both the CPU and the GPU.

5.6.3 HSA (Heterogeneous System Architecture) compliance (21)

Illustration of key features of hUMA [52]

Coherent memory

Ensures CPU and GPU caches both see an up-to-date view of data



Uniform memory

Provides the same visibility for both the CPU and GPU in the entire memory space

Physical Memory

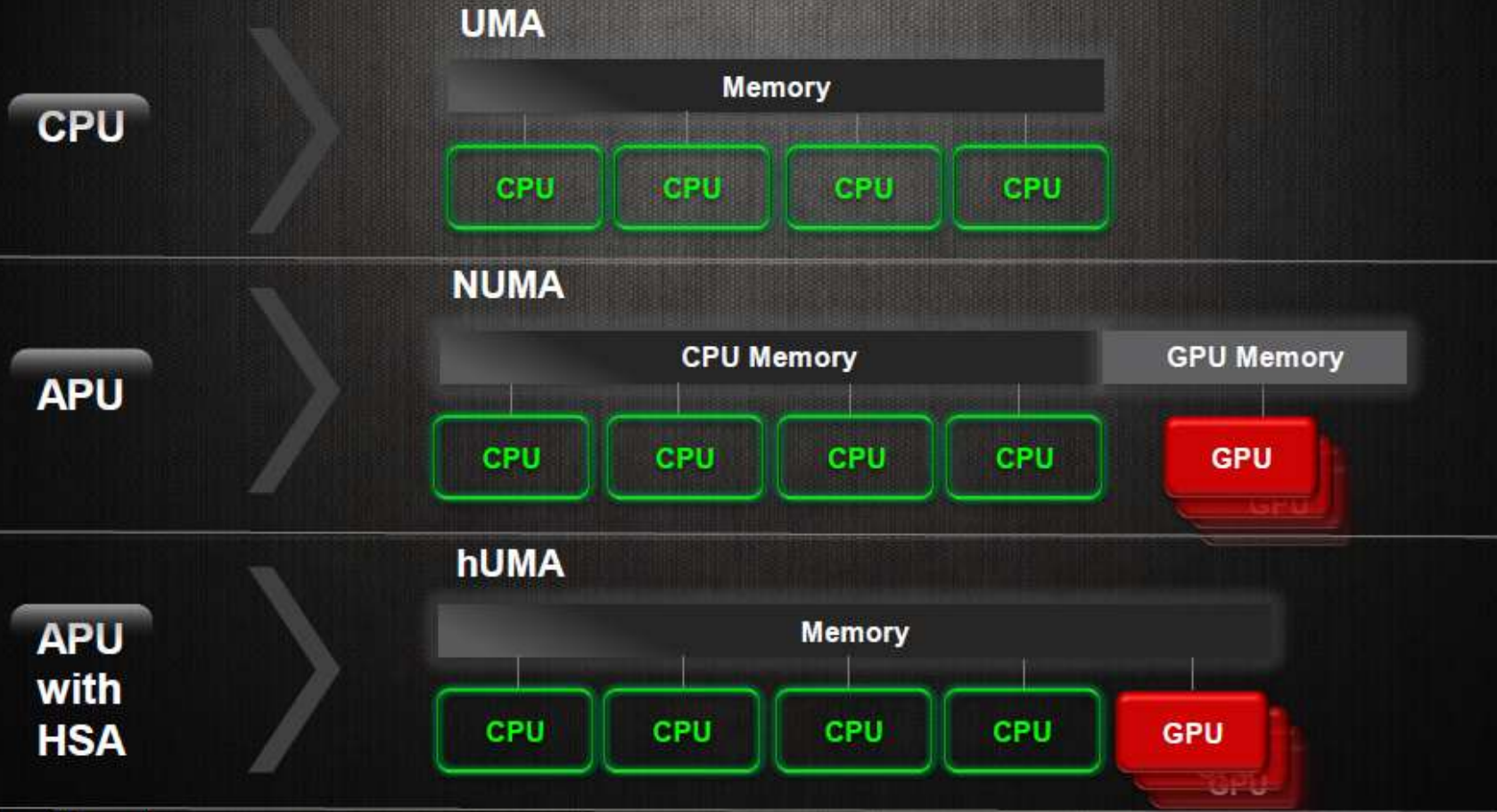
b) HSA (Heterogeneous System Architecture) compliance ()

Virtual Memory

Entire memory space:
Both CPU and GPU can access and allocate any location in the system's virtual memory space

Evolution of memory management while GPUs emerged [52]

INTRODUCING hUMA



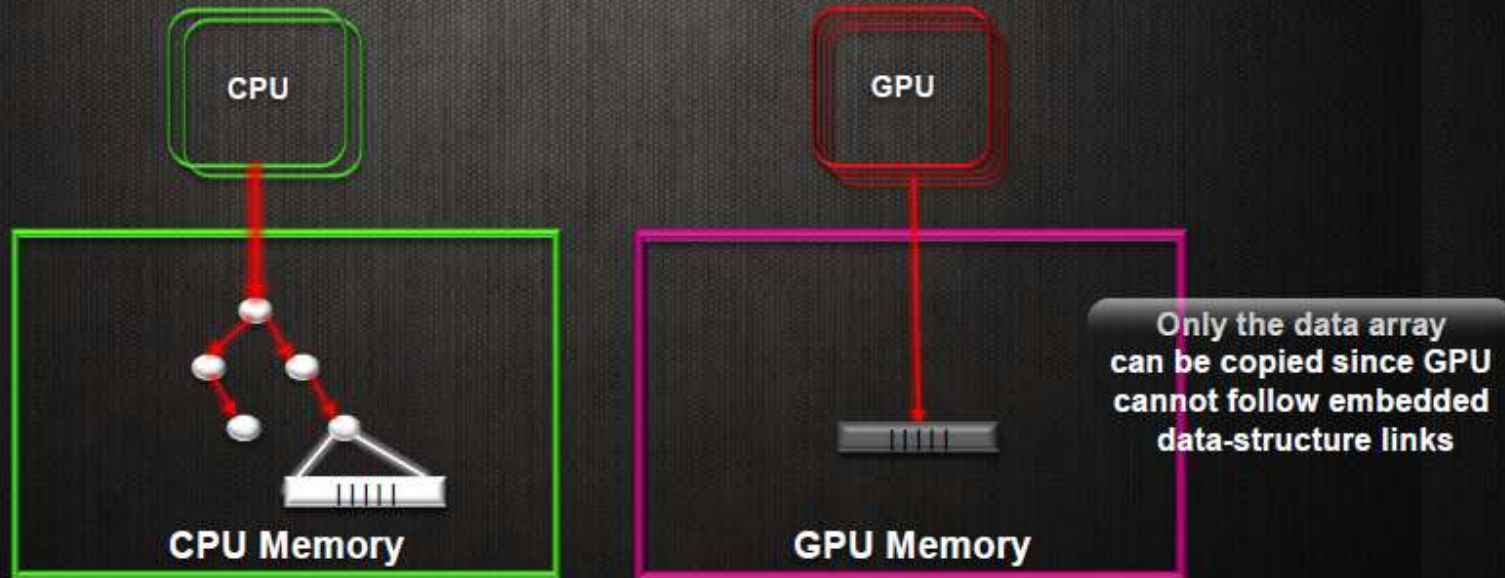
Data copying between the CPU and GPU memory without hUMA) [52]

WITHOUT POINTERS* AND DATA SHARING



Without hUMA:

- CPU explicitly copies data to GPU memory
- GPU completes computation
- CPU explicitly copies result back to CPU memory



5.6.3 HSA (Heterogeneous System Architecture) compliance (24)

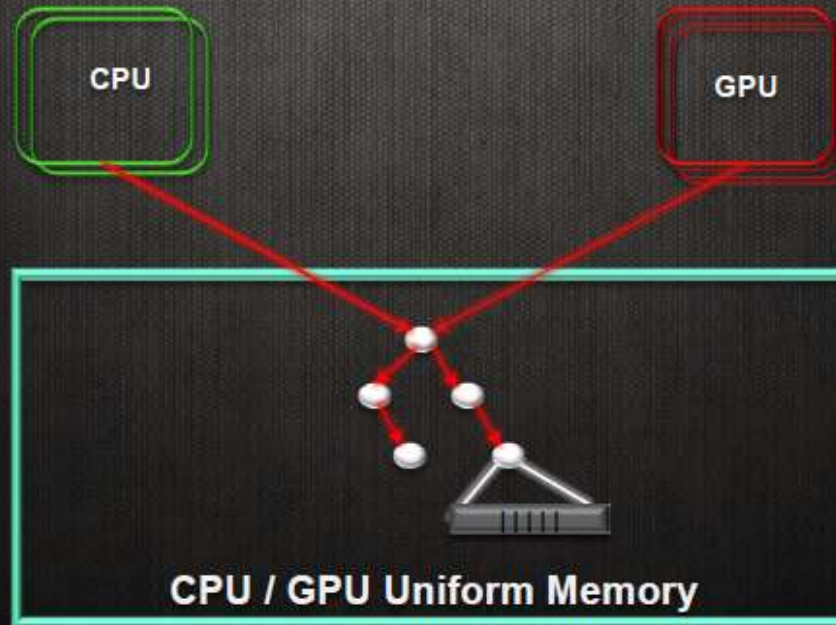
Data copying between the CPU and GPU memory with hUMA) [52]

WITH POINTERS* AND DATA SHARING



With hUMA:

- CPU simply passes a pointer to GPU
- GPU completes computation
- CPU can read the result directly – **no copying needed!**



CPU can pass a pointer to entire data structure since the GPU can now follow embedded links

Processors supporting HSA

At this time (05/2017) there are only a few processors that support HSA, including

- AMD Kaveri (2014), (first Steamroller-based APU), first proc. supporting HSA
- AMD's Carizzo (2015) (Excavator-based) HSA 1.0 support
- Samsung Exynos 8895 (2017)

Remarks -1

- Allegedly, AMD's Zen-based APUs will also support HSA [53].
- According to their roadmap for 2014/2015 NVIDIA planned to introduce unified virtual memory in their Maxwell GPU based Tegra X1 but changed this schedule and postponed the implementation of unified virtual memory until their Pascal GPU based Tegra Parker (P1) processor, announced in 08/2016.

This modification is the implication of NVIDIA's decision made in 2014/2015 to abandon the mobile market and developing processors for VR, AI and self-driving cars.

- As NVIDIA's next processor, the TegraX1 targeted presumable Nintendo's Switch game console there was no need to support data parallel computations like for scientific/engineering applications on the GPU in an efficient way.
- By contrast, NVIDIA's subsequent Tegra Parker processor targeted self-driving cars, this however implies a lot of AI tasks, so supporting the efficient execution of computing intensive tasks on a GPU became a priority that called for the implementation of unified virtual memory.

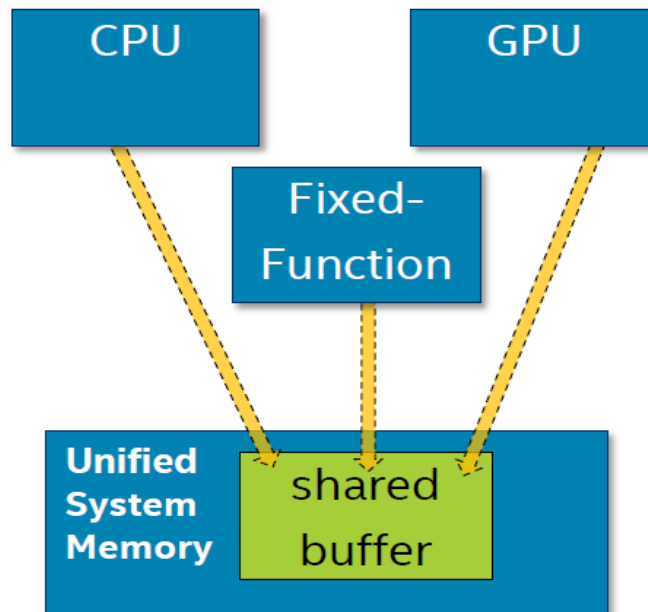
Remarks -2

- Also Intel began supporting the efficient execution of data parallel workloads on a GPU by implementing unified virtual memory, called SVM (Shared Virtual Memory) first in their Broadwell family in 2014, followed by the Skylake line (2015) as briefly shown subsequently.
- Nevertheless, Intel, like NVIDIA, don't take part in the HSA Consortium but follows an individual approach for supporting efficient computing on GPUs. Intel's solution is to provide unified virtual memory and falling back to OpenCL 2.0 in supporting HLL programming.

Shared virtual memory in the Broadwell line (2014) [54]

Shared Physical Memory (aka Unified Memory Architecture)

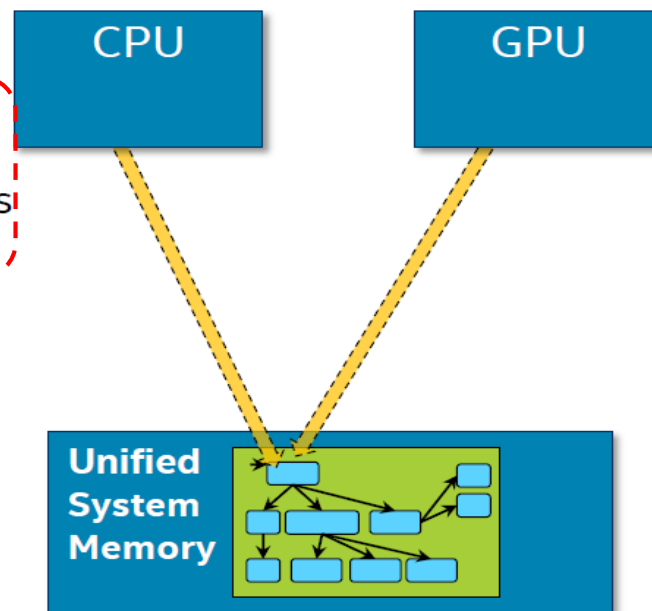
- No separate GDDR memory package or controller
- Processor Graphics has full performance access to system memory
- “Zero Copy” CPU & Graphics data sharing
- Enabled by buffer allocation flags in OpenCL*, DirectX*, etc.



Shared Physical Memory means “Zero Copy” Sharing

Shared Virtual memory (SVM) [54]

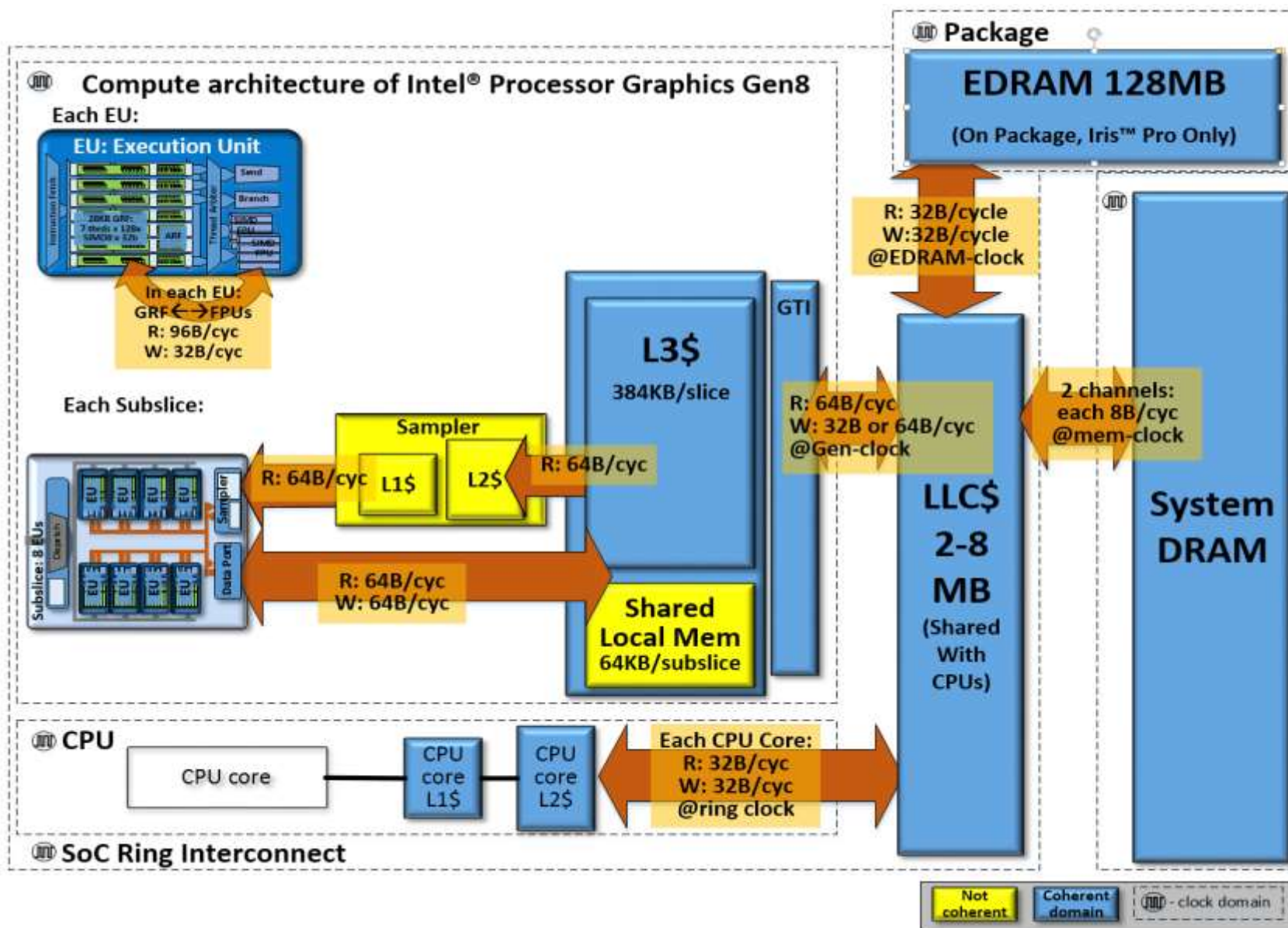
- Significant feature, new in Gen8
- Seamless sharing of pointer rich data-structures in a shared virtual address space
- Hardware-supported byte-level CPU & GPU coherency
- OpenCL* 2.0 Shared Virtual Memory:
 - Coarse & fine grained SVM
 - CPU & GPU atomics as synchronization primitives
 - System SVM as soon as OSVs are ready



Shared Virtual Memory enables seamless pointer sharing

5.6.3 HSA (Heterogeneous System Architecture) compliance (29)

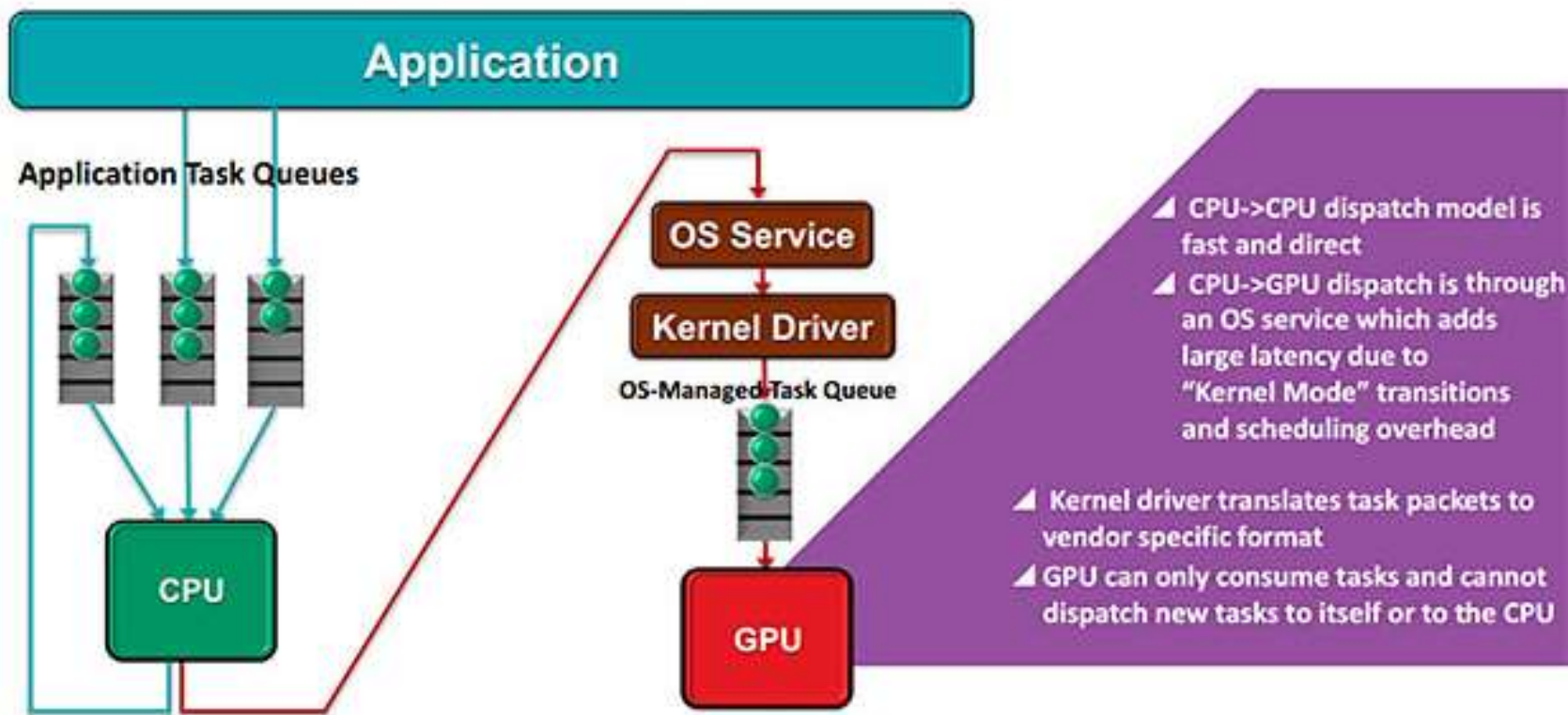
Coherent memory spaces in Broadwell's SVM (with Gen8 Graphics) [55]



5.6.3 HSA (Heterogeneous System Architecture) compliance (30)

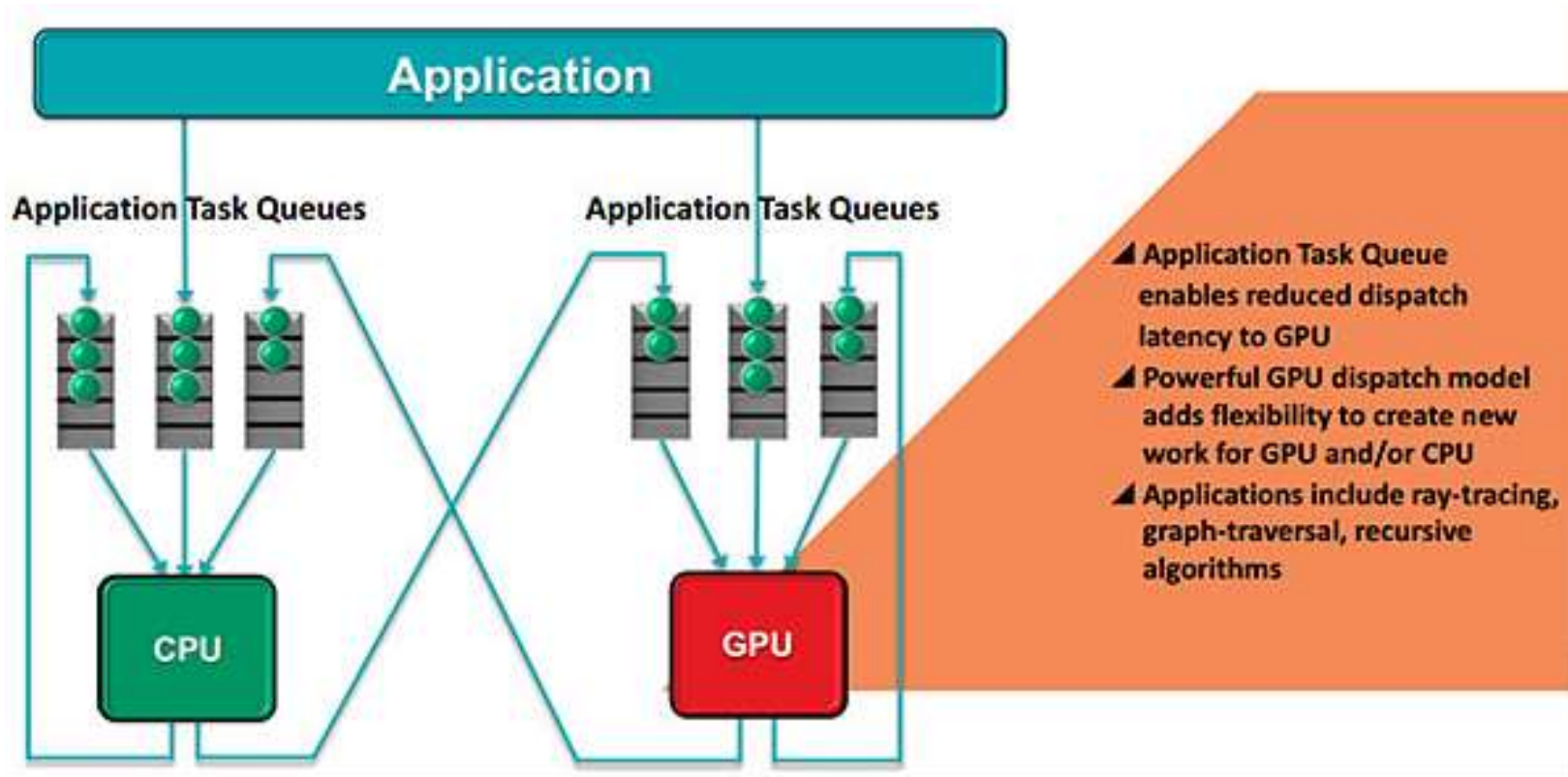
b4) Providing an efficient mechanism for forwarding tasks between the CPU and the GPU to eliminate in-efficient OS interactions

Traditional management of application task queues (based on [56])



5.6.3 HSA (Heterogeneous System Architecture) compliance (31)

Application task management with heterogeneous queuing (hQ) [56]



Main features of hQ [56]

- **Heterogeneous queuing (hQ)** is **symmetrical**.
It allows both the CPU and the GPU to generate tasks for themselves and for each other.
- Work is specified in a **standard packet format** that will be **supported by all ISA-compatible hardware**, so there's no need for the software to use vendor-specific code.
- **Applications can put packets directly into the task queues** that will be accessed **by the hardware**.
- **Each application can have multiple task queues**, and a virtualization layer allows HSA hardware to see all the queues.

5.6.4 Support for LPDDR4x memory

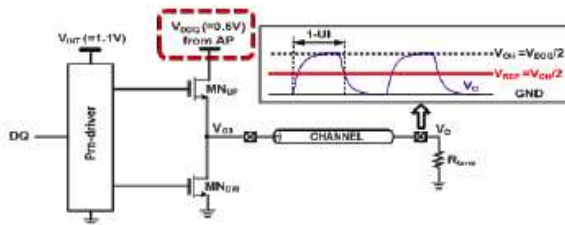
5.6.4 Support for LPDDR4x memory [36]

- **LPDDR4x** is an enhancement of the LPDDR4 memory technology.
- It lowers the output driver voltage (VDDQ) from 1.1 V to 0.6 V, this results in a 20 % reduction of DRAM power consumption, as indicated in the next Figure.
- Vendors began shipping LPDDR4x memory in the beginning of 2017.
- To date only Samsungs Exynos 8895 and Qualcomm's Snapdragon 835 make use of the LPDDR4x memory.

5.6.4 Support for LPDDR4x memory (2)

Contrasting LPDDR4 (LP4) and LPDDR4x (LP4x) [36]

LVSTL, VDDQ=0.6v (Tx only)



LP4/LP4x Spec Comparison

Spec	LPDDR4	LPDDR4X
Function	Same	
Speed*	3200/3733/4266	3200/3733/4266
AC Timing	Same	
VDD1/VDD2	Same(VDD1=1.8V/VDD2=1.1V)	
VDDQ	1.1V	0.6V
Power Eff. (mW/GB/s)	1X	0.8X

LP4x = Lower Power



5.6.5 Separate security processing unit

5.6.5 Separate security processing unit

- It is used for user authentication, mobile payments etc.
- It represents an enhanced security subsystem like Apple's Secure Enclave.

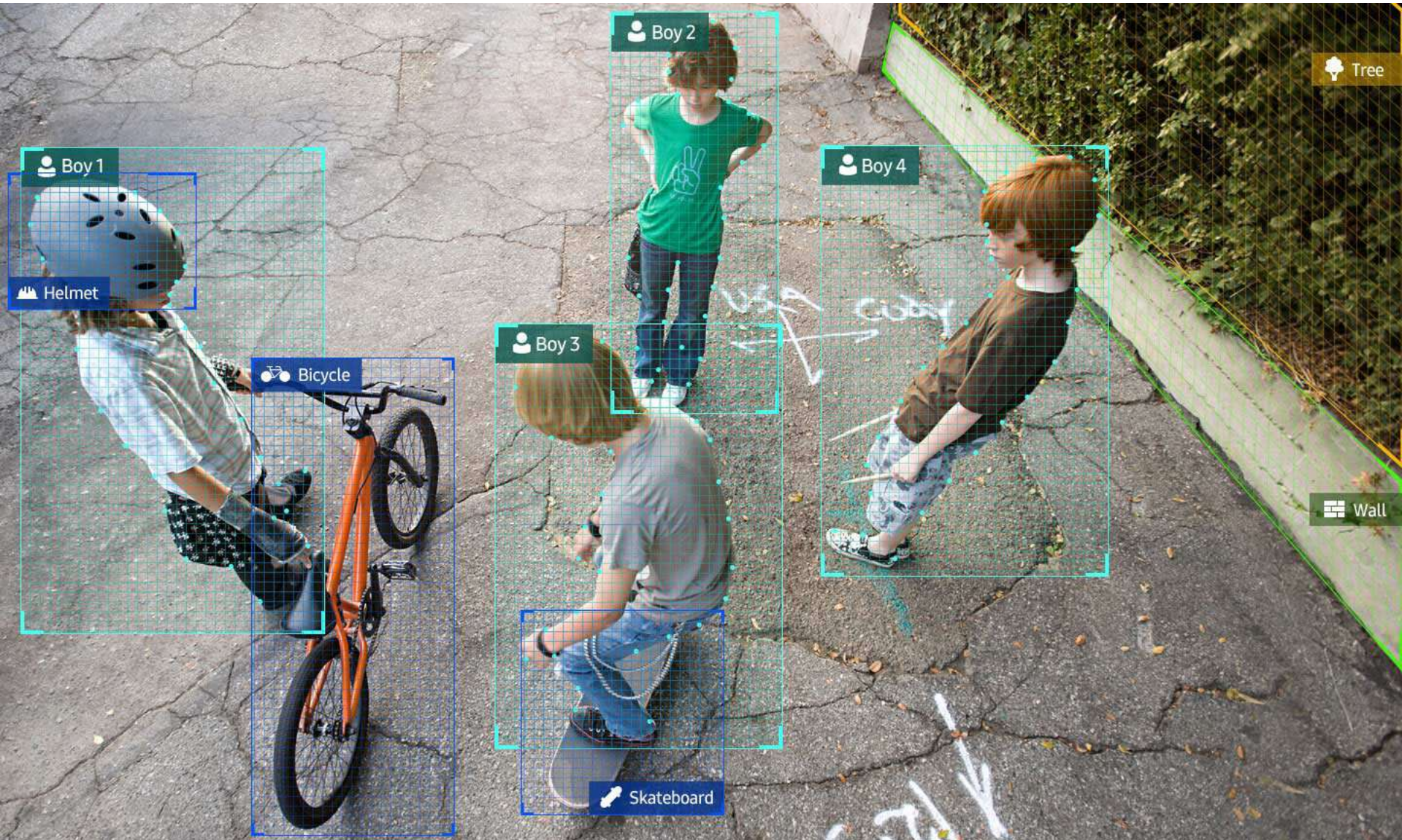
5.6.6 Vision processing Unit (VPU)

5.6.6 Vision Processing Unit (VPU) -1 [73]

The **Vision Processing Unit (VPU)** of the Exynos 8895 is designed for **enhancing machine vision technology**, including corner detection, recognition of objects, analyzing visual information coming from a camera, VR etc., as indicated in the next Figure.

5.6.6 Vision processing Unit (VPU) (2)

Object recognition by Samsung's VPU implemented in the Exynos 8895 [73]



Vision Processing Unit (VPU) -2 []

- **VPUs differ from GPUs** as they are **designed from the ground up to efficiently process computer vision algorithms** whereas GPUs target processing graphics in general [74].
- Presently, there is no available information about the architecture of Samsung's VPU implemented in the Exynos 8895.
- However, in order to give a glimpse about the built up of a GPU subsequently we briefly present the block diagram of the **Mirriad 2 VPU from Movidius**, now part of Intel (since 09/2016).

Main features of the Miriad 2 VPU (2015) [75]

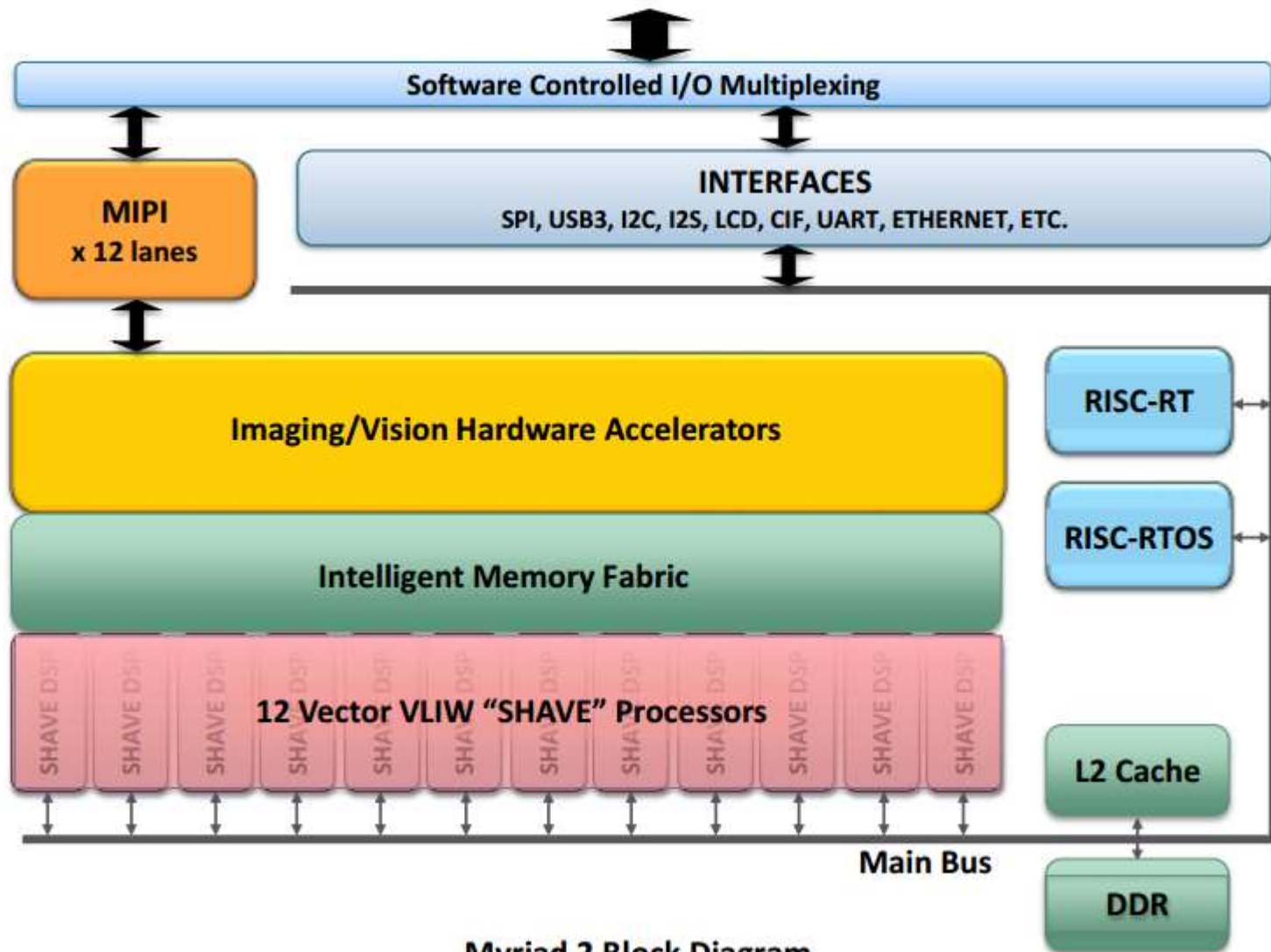
It is based

- on **two 32-bit RISC processors**, one is dedicated to scheduling within the SoC, and the other to running user code within a real-time OS (RTOS)
- about **20 hardware accelerators** for imaging/vision,
- **12 VLIW cores** (termed as SHAVE processors) and
- **2 MB of on-chip memory** that is shared between the CPUs, SHAVE processors and fixed-function accelerators and

as shown in the next two Figures.

5.6.6 Vision processing Unit (VPU) (5)

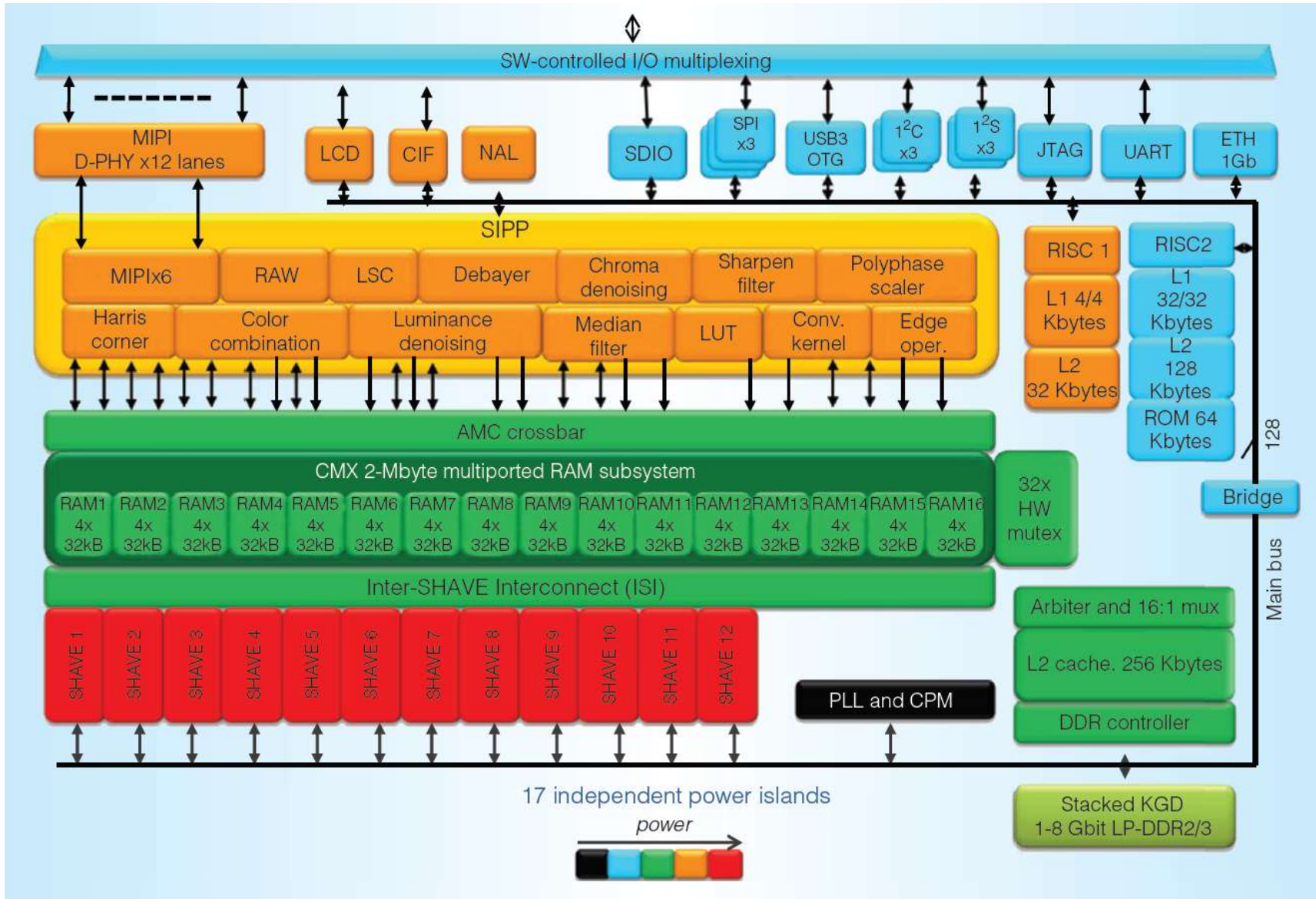
Block diagram of the Miriad 2 VPU from Movidius, now part of Intel [75]



Myriad 2 Block Diagram

5.6.6 Vision processing Unit (VPU) (6)

A more detailed block diagram of the Miriad 2 VPU [76]

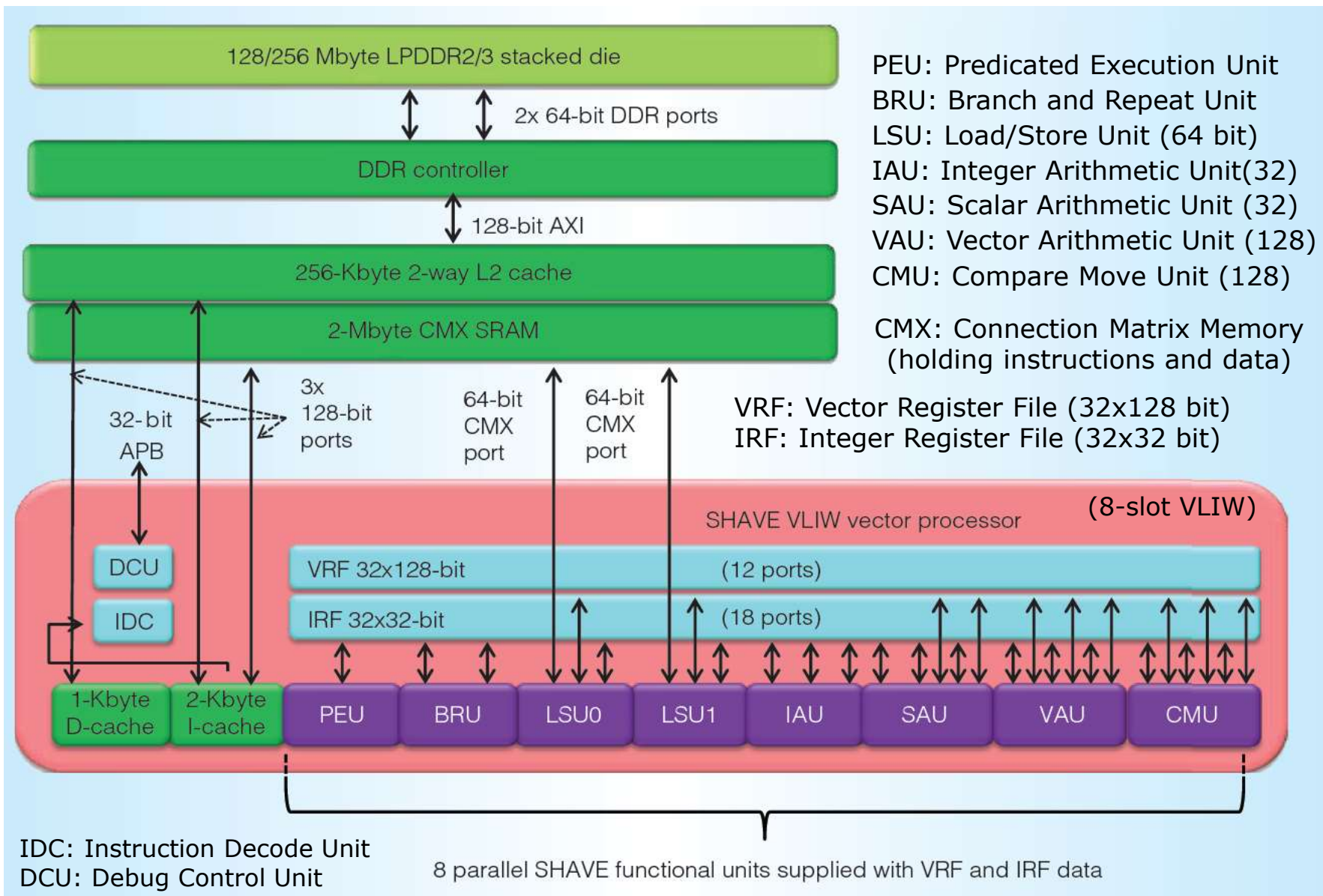


The SHAVE processor [76]

- The 12 SHAVE processors are the kernel piece of the VPU.
- Each SHAVE processor is an 8-way variable length VLIW unit.
- Separate fields of the VLIW instructions control the functional units.
- Individual fields are enabled separately by a header in the variable length VLIW instruction.
- Variable length VLIW instructions are fetched in 128-bit chunks, and the average instruction width is around 80 bits.
- The functional units access their operands from a 128 bits x 32-entry vector register file with 12 ports and a 32-bit x 32-entry integer register file with 18 ports, as indicated in the next Figure.

5.6.6 Vision processing Unit (VPU) (8)

Block diagram of the SHAVE VLIW core (from the Miriad I) [76]



5.7 Samsung's first SOC supporting the DynamIQ cluster technology: the Exynos 9 Series 9810 (2018)

- 5.7.1 The Exynos 9 Series 9810 Overview
- 5.7.2 Microarchitecture of the M3 core
- 5.7.3 The DynamIQ technology as an evolution of the big.LITTLE technology

5.7.1 The Exynos 9 9810 - Overview

5.7.1 The Exynos 9 Series 9810 - Overview

- It is fabricated based on Samsung's **10 nm FinFET LPP process**.
- The Exynos 9 9810 is based on the **M3 core** and provides **about 100 % single-core and 40 % multi-core performance increase** over the 8895 that arises
 - partly from the **6-wide microarchitecture** of the M3 processor used and
 - partly from using the **DynamIQ core technology** instead of the big.LITTLE technology.
- The Exynos 9810 is the kernel piece of one alternative of Samsung's **Galaxy S9, S9+**.

The other alternative is using Qualcomm's Snapdragon 845 for these mobiles (sold in the US).

- It was announced in 01/2018 and first shipped in **02/2018**.

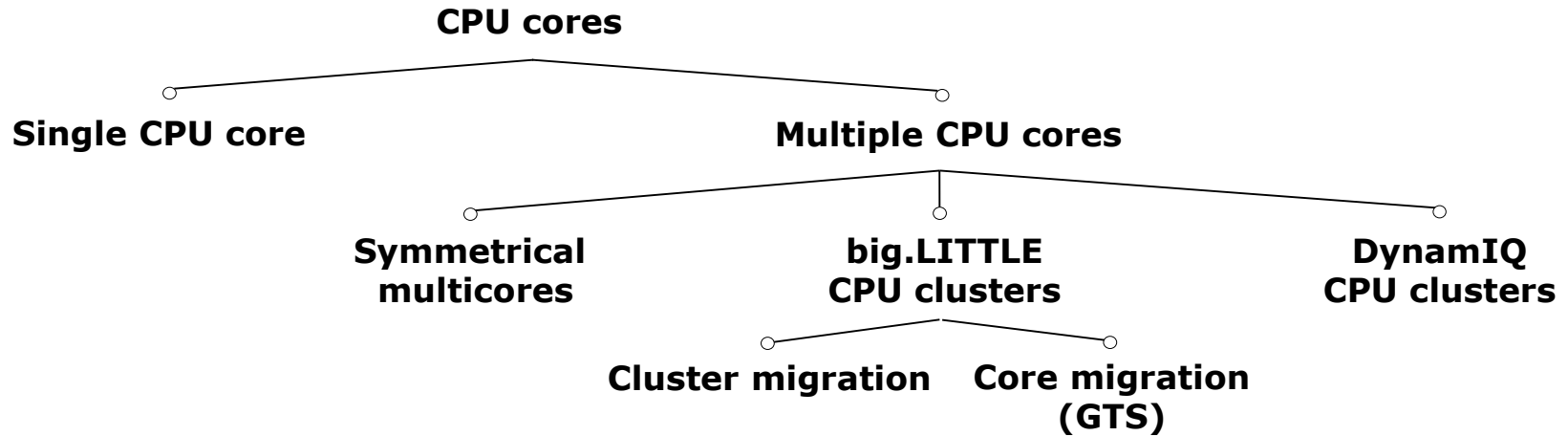
5.7.1 The Exynos 9 Series 9810 - Overview (2)

Main features of Samsung's Exynos 9 Series 9810 (2018)

SoC		CPU				GPU	Memory technology	Availability	Utilizing devices (examples)
Model number	fab	Instr. set	Cores	No of cores	fc (GHz)				
Exynos 5 Octa (Exynos 5420)	28 nm HKMG	ARM v7	Cortex-A15+ Cortex-A7	4+4	1.8-1.9 1.2-1.3	ARM Mali-T628 MP6 @ 533 MHz; 109 GFLOPS	32-bit DCh LPDDR3e-1866 (14.9 GB/sec)	Q3 2013	Samsung Chromebook 2 11.6", Samsung Galaxy Note 3/Note 10.1/Note Pro 12.2, Samsung Galaxy Tab Pro/Tab S
Exynos 5 Octa (Exynos 5422)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.9-2.1 1.3-1.5	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Galaxy S5 (SM-G900H)
Exynos 5 Octa (Exynos 5800)	28 nm HKMG		Cortex-A15+ Cortex-A7	4+4	2.1 1.3	ARM Mali-T628 MP6 @ 533 MHz 109 GFLOPS	32-bit DCh LPDDR3/DDR3-1866 (14.9 GB/sec)	Q2 2014	Samsung Chromebook 2 13,3"
Exynos 5 Octa (Exynos 5430)	20 nm HKMG		Cortex-A15+ Cortex-A7	4+4	1.8-2.0 1.3-1.5	ARM Mali-T628 MP6 @ 600 MHz; 122 GFLOPS	32-bit DCh LPDDR3e/DDR3-2132 (17.0 GB/sec)	Q3 2014	Samsung Galaxy Alpha (SM-G850F)
Exynos 7 Octa (Exynos 5433)	20 nm HKMG	ARM v8-A	Cortex-A57+ Cortex-A53	4+4	1.9 1.3	Mali-T760 MP6 @ 700 MHz; 206 GFLOPS (FP16)	32-bits DCh LPDDR3-1650 (13.2 GB/s)	Q3/Q4 2014	Samsung Galaxy Note 4 (SM-N910C)
Exynos 7 Octa (Exynos 7420)	14 nm FinFET		Cortex-A57+ Cortex-A53	4+4	2.1 1.5	Mali-T760 MP8 @ 772 MHz; 227 GFLOPS (FP16)	32-bits DCh LPDDR4-3104 (24.9 GB/s)	Q2 2015	Samsung Galaxy S6 S6 Edge
Exynos 7 Octa (Exynos 7885)	14 nm HKMG		Cortex-A73+ Cortex-A53	4+4	2.2 1.6	Mali-G71 MP2	32-bits DCh LPDDR4x	Q1 2016	Samsung Galaxy A8
Exynos 8 Octa (Exynos 8890)	14 nm FinFET		Samsung M1+ Cortex-A53	4+4	2.6-2.3 1.6	Mali-T880 MP12 @ 650 MHz; 265.2 GFLOPS (FP16)	32-bits DCh LPDDR4-3588 (28.7 GB/s)	Q1 2016	Samsung Galaxy S7 Samsung Galaxy S7 Edge
Exynos 9 Series (Exynos 8895)	10 nm FinFET		Samsung M2+ Cortex-A53	4+4	2.5 1.7	Mali-G71 MP20	32-bits DCh? LPDDR4x	Q2 2017	Samsung Galaxy S8 Samsung Galaxy S8 Plus
Exynos 9 Series (Exynos 9810)	10 nm FinFET	Samsung M3+ Cortex-A55	4+4	2.9 1.9	Mali-G72 MP18	32-bits DCh? LPDDR4x	Q1 2018	Samsung Galaxy S9 Samsung Galaxy S9 Plus	

5.7.1 The Exynos 9 Series 9810 – Overview (3)

Evolution of the width of mobile cores



Apple	ARM1176 (2007) until A4 (2010)	→	A5 (2011) (2C)	→	A10 (2016) (2+2)	→	A11 (2017) (2+4)		
Samsung Exynos	3110 (2010)	→	3250 2C (2011) 4412 4C (2012)	→	5410 (2013) (4+4)	→	5420 (2013) (4+4)	→	9810 (2018) (4+4)
Qualcomm Snapdragon	MSM 7225 (2010)	→	8260 2C (2013) 400 4C (2013)	→		→	808 (2+4) (2014) 810 (4+4) (2015)	→	845 (2018) (4+4)
Huawei Kirin	(K3V1) (2009)	→	(K3V2 (2012))	→		→	920 (4+4) (2014)	→	980 (2018) (4+4)
MediaTek	MT6218B (2003)	→	MT6582 4C (2013) MT6592 8C (2013)	→		→	MT6595 8C (2014)	→	MT6795 10C (2019)

5.7.1 The Exynos 9 Series 9810 – Overview (4)

Main features of the Exynos 9810 vs. the Exynos 8995 [68]

Samsung Exynos SoCs Specifications		
SoC	Exynos 9810	Exynos 8895
CPU	4x Exynos M3 @ 2.9 GHz 4x 512KB L2 ?? 4x Cortex A55 @ 1.9 GHz 4x 128KB L2 4096KB L3 DSU ??	4x Exynos M2 @ 2.314 GHz 2048KB L2 4x Cortex A53 @ 1.690GHz 512KB L2
GPU	Mali G72MP18	Mali G71MP20 @ 546MHz
Memory Controller	4x 16-bit CH LPDDR4x @ 1794MHz	4x 16-bit CH LPDDR4x @ 1794MHz
Media	10bit 4K120 encode & decode H.265/HEVC, H.264, VP9	28.7GB/s B/W 4K120 encode & decode H.265/HEVC, H.264, VP9
Modem	Shannon Integrated LTE (Category 18/13) DL = 1200 Mbps 6x20MHz CA, 256-QAM UL = 200 Mbps 2x20MHz CA, 256-QAM	Shannon 355 Integrated LTE (Category 16/13) DL = 1050 Mbps 5x20MHz CA, 256-QAM UL = 150 Mbps 2x20MHz CA, 64-QAM
ISP	Rear: 24MP Front: 24MP Dual: 16MP+16MP	Rear: 28MP Front: 28MP
Mfc. Process	Samsung 10nm LPP	Samsung 10nm LPE

5.7.1 The Exynos 9 Series 9810 – Overview (5)

The Samsung Exynos 9810 vs. the Qualcomm Snapdragon 845 [69]

	Snapdragon 845	Exynos 9810
CPU	4 x Kryo 385 @2.8GHz + 4 x Kryo 385 @1.8 GHz	4 x M3 (Cortex-A75 based) @2.9GHz + 4 x Cortex-A55 @1.9GHz
GPU	Adreno 630: Open GL ES 3.2, Open CL 2.0, Vulkan, DirectX 12	Mali-G72: 18-cores @700MHz
RAM	LPDDR4x	LPDDR4x
AI co-processor	Hexagon 685 DSP	3x VPU
Modem	Qualcomm Snapdragon X20 LTE modem: •Download speed: 1.2Gbps •Upload speed: 150Mbps	Custom Cat.18 LTE modem: •Download speed: 1.2Gbps •Upload speed: 200Mbps
Battery Charging	Quick Charge 4+ (USB PD Compatible)	Samsung Adaptive Fast Charge, Fast Wireless Charging (Qi & PMA)
Wi-Fi	Multi-gigabit Wi-Fi ac/b/g/n with MU-MIMO	Dual-Band Wi-Fi ac/b/g/n with MU-MIMO
Bluetooth	5.0	5.0
Camera	32MP Single, 16MP Dual	24MP Single, 16MP+16MP Dual
Video Recording and Encoding	4K (3840×2160) @60fps, 10bit HDR, Rec 2020 color gamut, H.264 (AVC), H.265 (HEVC), VP9	MFC, 4K (3840×2160) @120fps, 10-bit HEVC (H.265), H.264, VP9
Manufacturing	2nd-gen 10nm LPP FinFET	2nd-gen 10nm LPP FinFET

5.7.1 The Exynos 9 Series 9810 – Overview (6)

Exynos 9810's enhanced modem speeds vs. the Exynos 8895 [73], [77]

Exynos 9810



Exynos 8895



Main innovations of the Exynos 9 Series 9810

- It is built on the **M3 core** (called **Meerkat**)
- **It includes an L3 cache**, shared by 4 cores (exclusive to the private L2 caches)
- It is based on the **DynamIQ** core technology

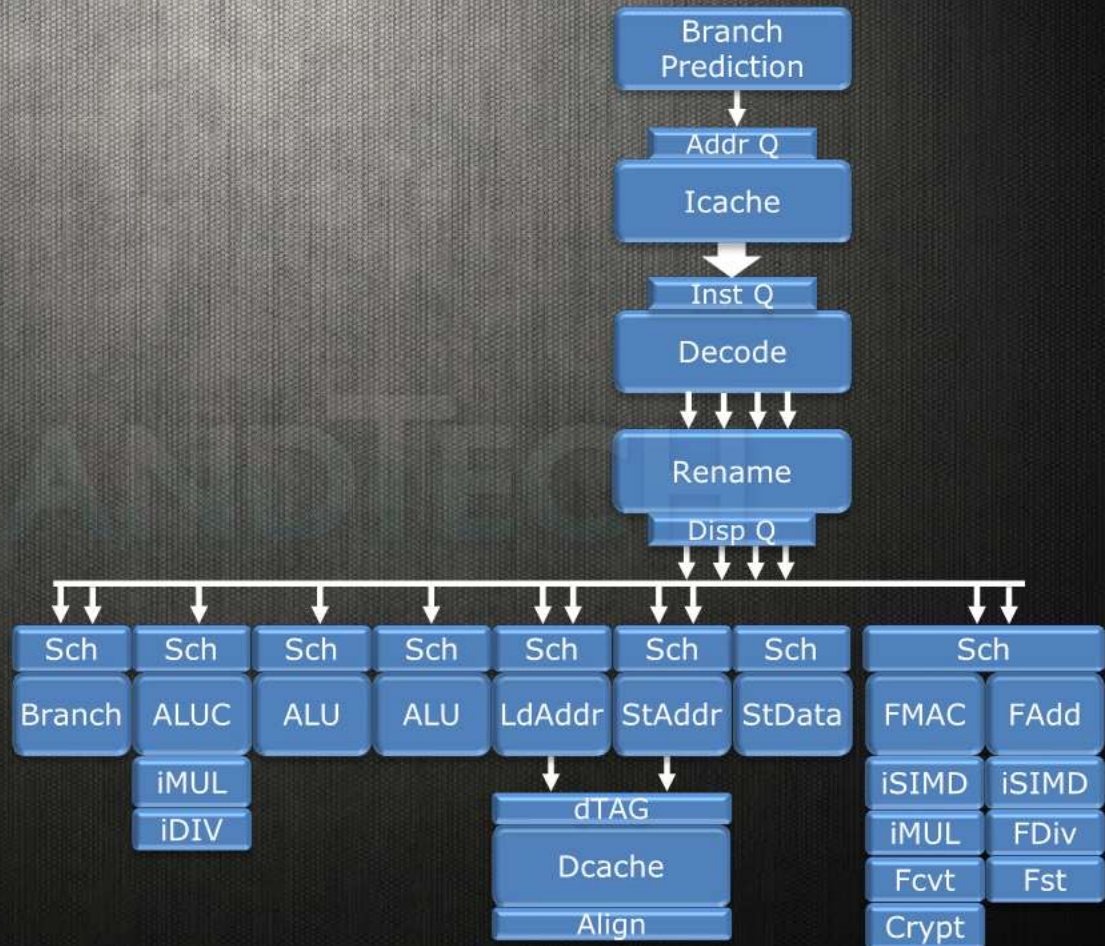
These innovations will be discussed next.

5.7.2 Microarchitecture of the M3 core

5.7.2 Microarchitecture of the M3 core (1)

For comparison: Microarchitecture of the M1/M2 cores [79]

Samsung M1 Micro-Architecture

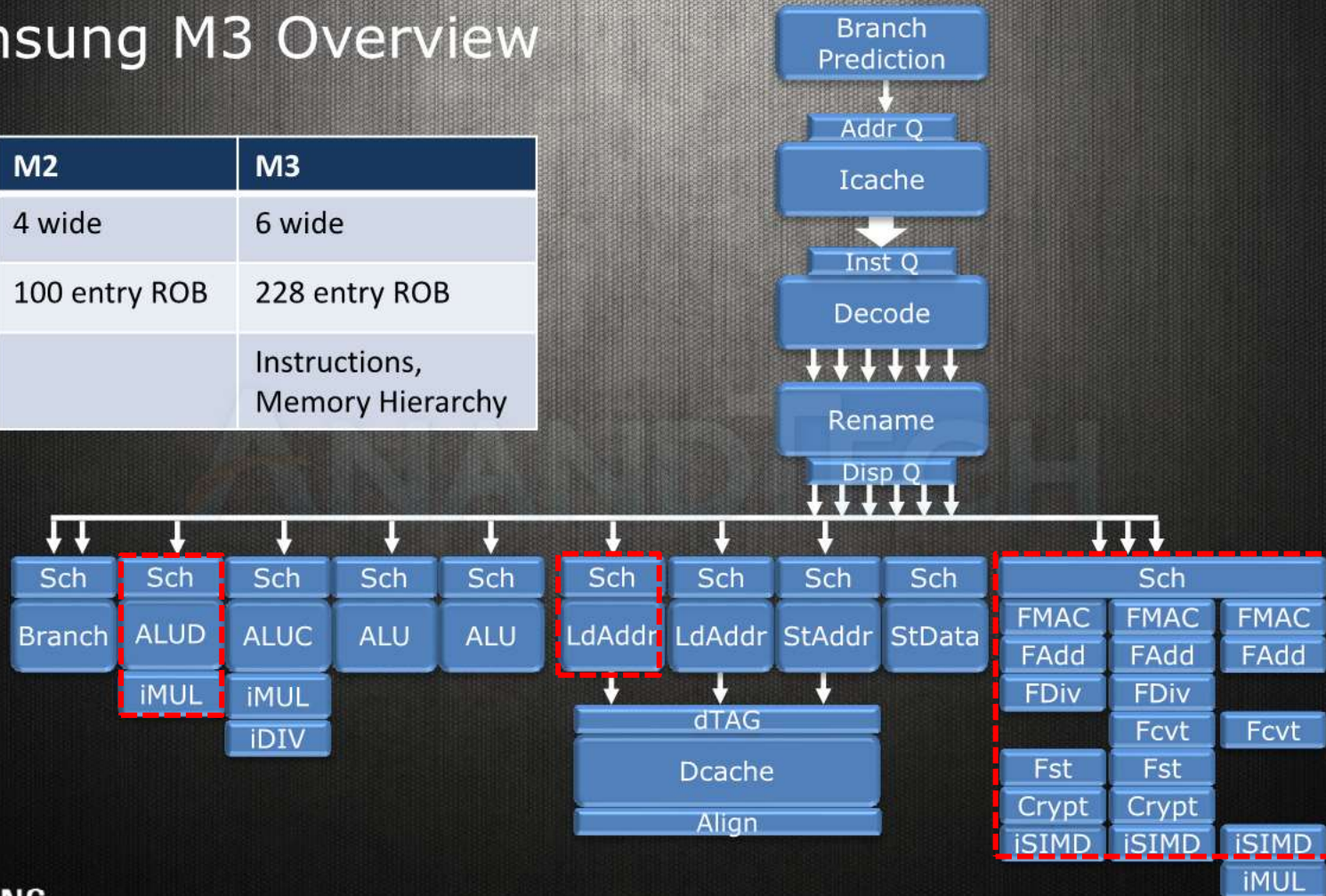


5.7.2 Microarchitecture of the M3 core (2)

Enhancements of the microarchitecture of the M3 core [79]

Samsung M3 Overview

	M2	M3
Wider	4 wide	6 wide
Deeper	100 entry ROB	228 entry ROB
Faster		Instructions, Memory Hierarchy



Enhancements of the M3 front-end [79]

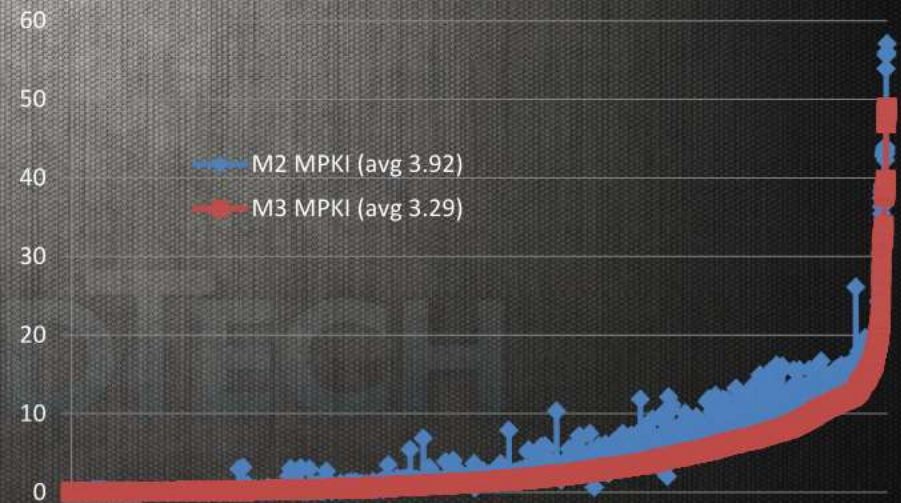
Samsung M3 Front End

M3 Branch Prediction:

- 128-entry microBTB (2x)
 - 4K-entry mainBTB: improved branch-taken latency
 - 16K capacity L2 BTB (2x capacity, 2x bandwidth)
 - Conditional predictor improvements including more weights for Neural Net
 - Improvements to the Indirect Predictor
- => Net of above led to average MPKI reduction ~15%

M3 Instruction Fetch:

- 64KB/4-way
- Read up to 48-Bytes / cycle (2x fetch width)
- Decoupling Instruction Queue (nearly 2x deeper)
- 512 entry ITLB (2x)



MPKI comparison across ~4800 traces sorted by M3

Enhancements of the M3 “middle machine” [79]

Samsung M3 Middle Machine

- Wider:
 - Decode up to 6 inst/cycle (1.5x wider than prior)
 - Several fusion idioms supported
 - Rename, Dispatch, Retire: up to 6 uops/cycle (1.5x wider than prior)
 - Up to 9 integer ops issued/cycle (versus 7 in prior)
 - 4th ALU including a 2nd integer multiplier
 - 2nd Load AGU – part of doubling load bandwidth
- Deeper:
 - 228-entry ROB (>2x deeper program window than prior)
 - 126-entry distributed scheduler (>2x deeper than M1)
- Faster Instructions:
 - Additional 1-cycle latency ops
 - Some ops optimized to 0-cycle latency
 - Integer Divider now radix 16 (4 bits/cycle) versus prior radix 4 (2 bits/cycle)

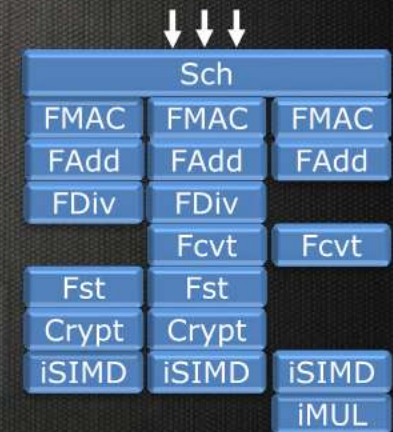
Enhancements of the M3 FPU [79]

Samsung M3 FPU

- Wider:
 - 3rd dispatch and issue ports (1.5x)
 - 3 128b FMAC/FADD (versus M1 1 128b FMAC + 1 128b FADD) => 2x maximum FLOPS
 - 2nd 128b Load port => critical to feed the FP “beast”
- Deeper Out-of-Order
 - 62-entry scheduler (nearly 2x versus prior)
 - 192-entry FP PRF (2x versus prior)
- Faster Instructions:
 - FMAC : 4-cycle MAC (was 5)
3-cycle Mul (was 4)
 - FADD: 2-cycle (was 3)
 - FDIV: radix64 (was radix4)
=> 6 bits/cycle versus 2



M1 FPU



M3 FPU

Enhancements of the M3 Load/Store Unit [79]

Samsung M3 Load/Store Unit

- Bandwidths:
 - 2-Load/cycle (2x read bandwidth vs. prior)
 - 1-Store/cycle
 - Additional Stream/Copy Optimizations
- Depth:
 - Larger schedulers
 - Doubled Store Buffer
 - 12 outstanding misses (8 prior)
- Latencies:
 - 64KB/8-way D\$ for 4 cycle (integer)
 - twice the former capacity @ same latency
 - Enhanced and Hybridized Prefetcher
 - TLBs
 - New mid-level 512-entry DTLB
 - Enhanced unified L2TLB – 4K entry (vs 1K)



Longer pipeline of the M3 core [79] -1

Samsung M3 Core Pipeline



Deeper and Wider were not free. Versus M1:
 1: A second stage of dispatch was added
 2: A second stage for PRF read was added

M3 cache hierarchy [79]

Samsung M3 Cache Hierarchy

M1/M2: 16B/cycle/CPU
shared L2 (inclusive of D\$)

- 2MB, 16-way, 22c

L2/BIU: 28 outstanding transactions

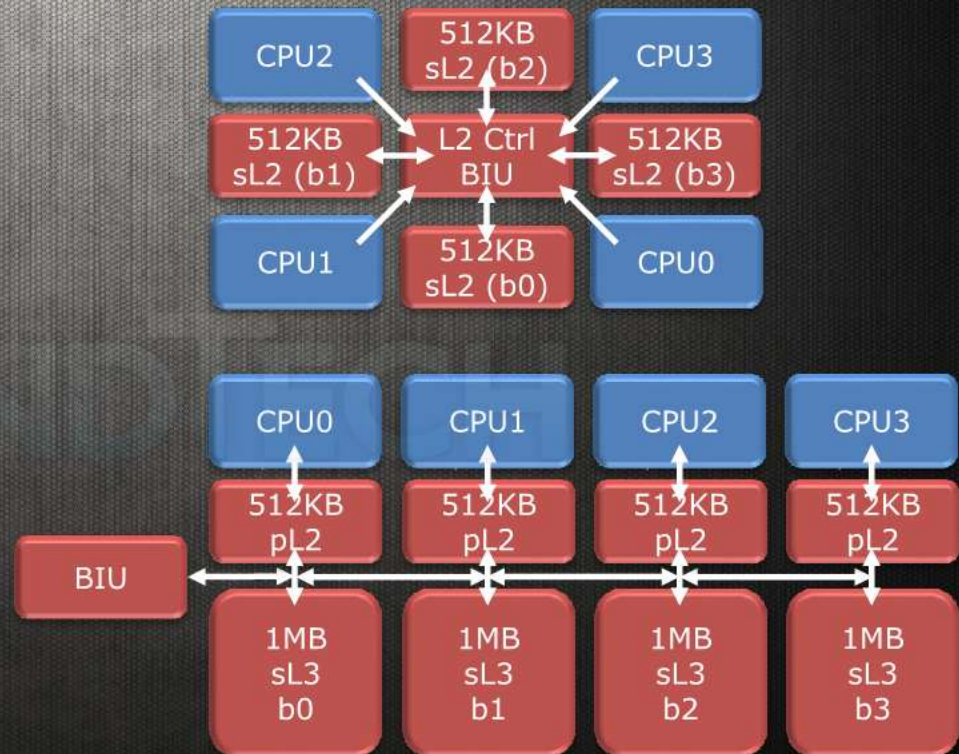
M3: 32B/cycle/CPU (2x bandwidth)
Private L2 (inclusive of D\$)

- 512KB, 8w, 12c

SL3 (exclusive of L2\$)

- 4MB, 16/way, ~37c typical (NUCA)
- Slice design - 1MB per slice
=> Goal: configurability

BIU – 80 outstanding transactions



M3 cache hierarchy [79] -2

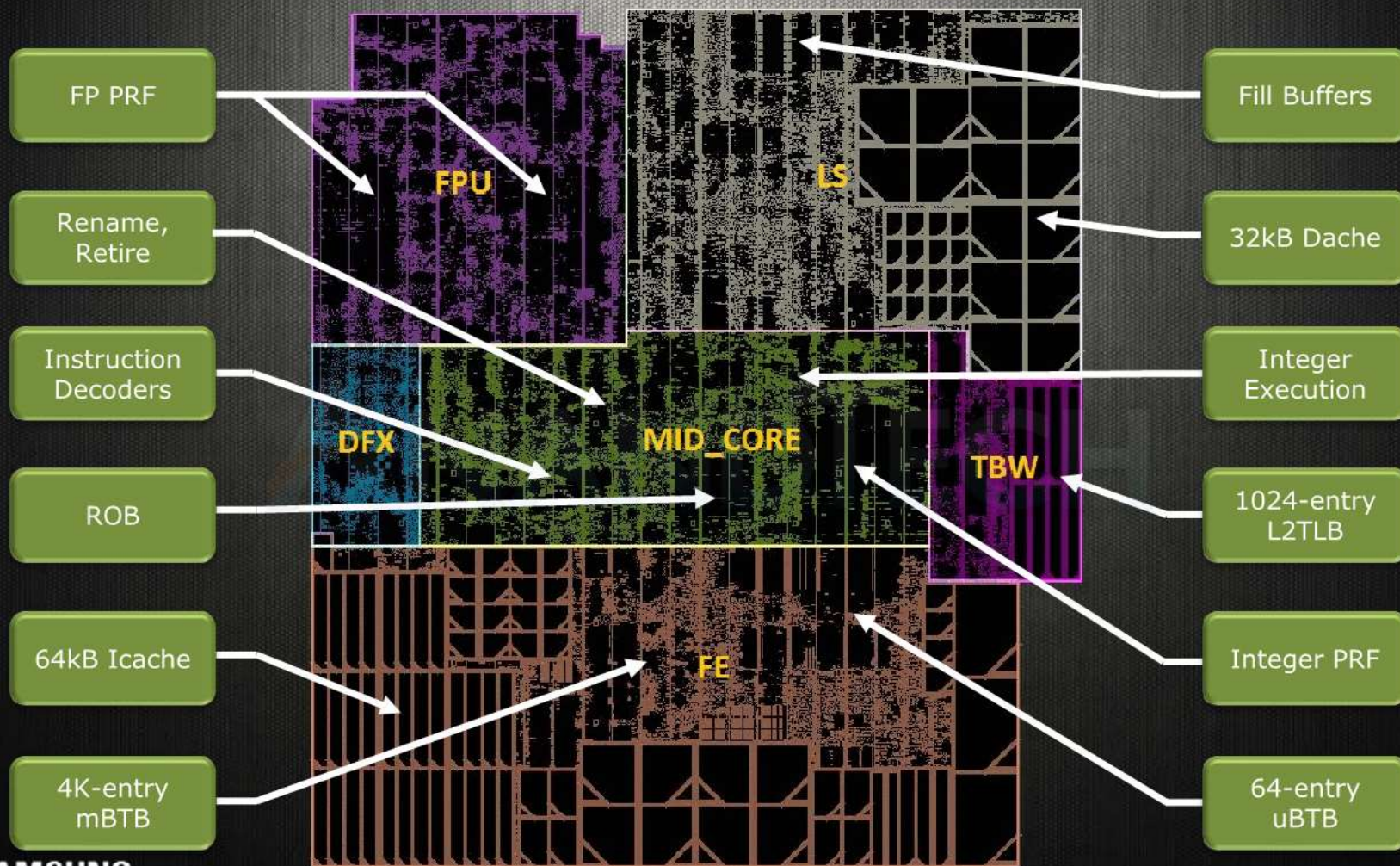
- As seen in the previous Figure M3 has an **L3 cache** included.
- The L3 cache is implemented **sliced**, in **NUCA style (Non-Uniform Cache Architecture)** rather than in **UCA (Uniform cache Architecture)** as in AMD's CCX module.

A core can access an **adjacent slice** in **32 cycles**, and the **furthest slice** in **44 cycles**.

- The L3 cache is **exclusive** to the related private L2 cache.
- Further on, an L3 slice is **on the same clock plane as the related CPU core**.

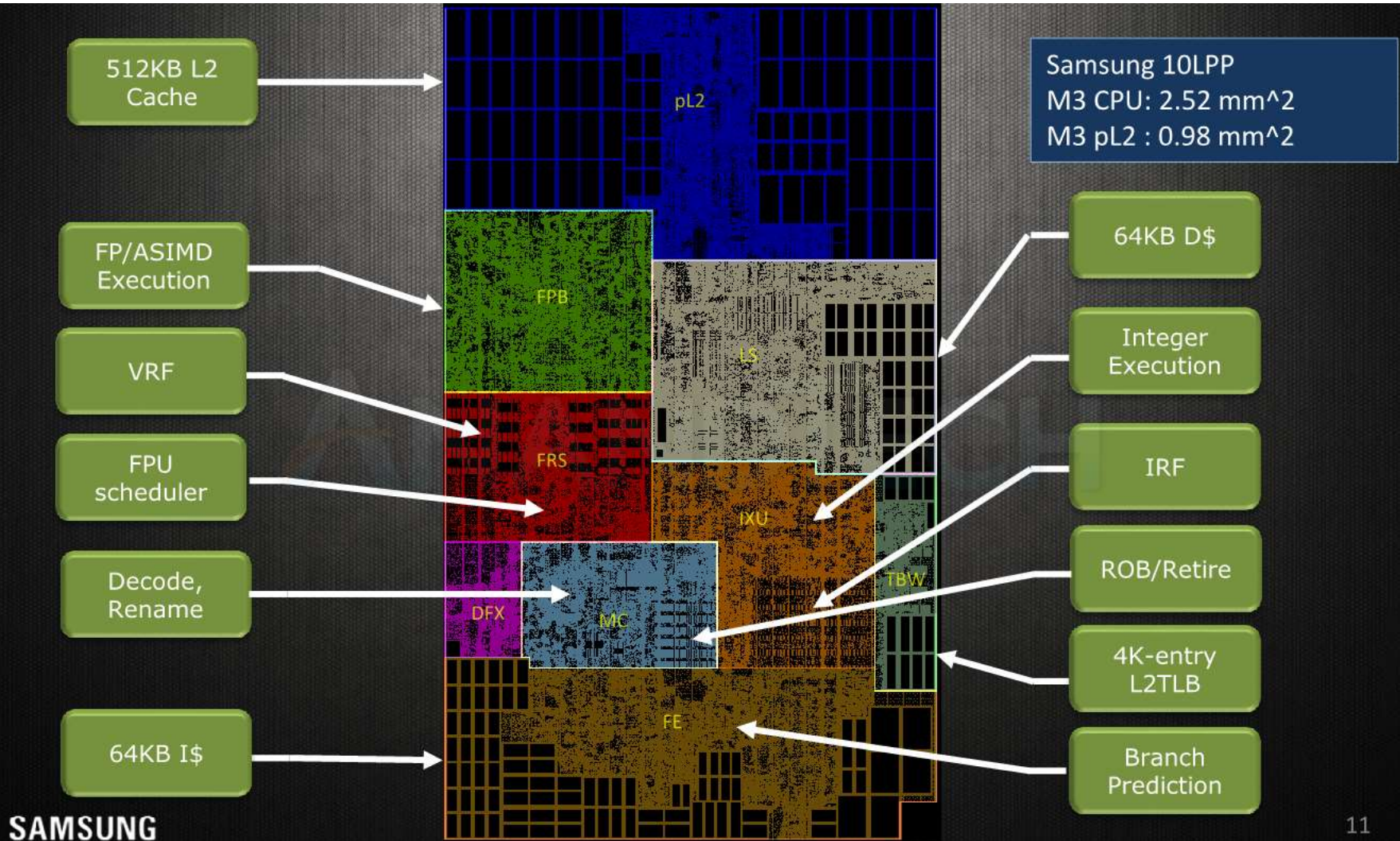
5.7.2 Microarchitecture of the M3 core (10)

M1 Core Layout [97]



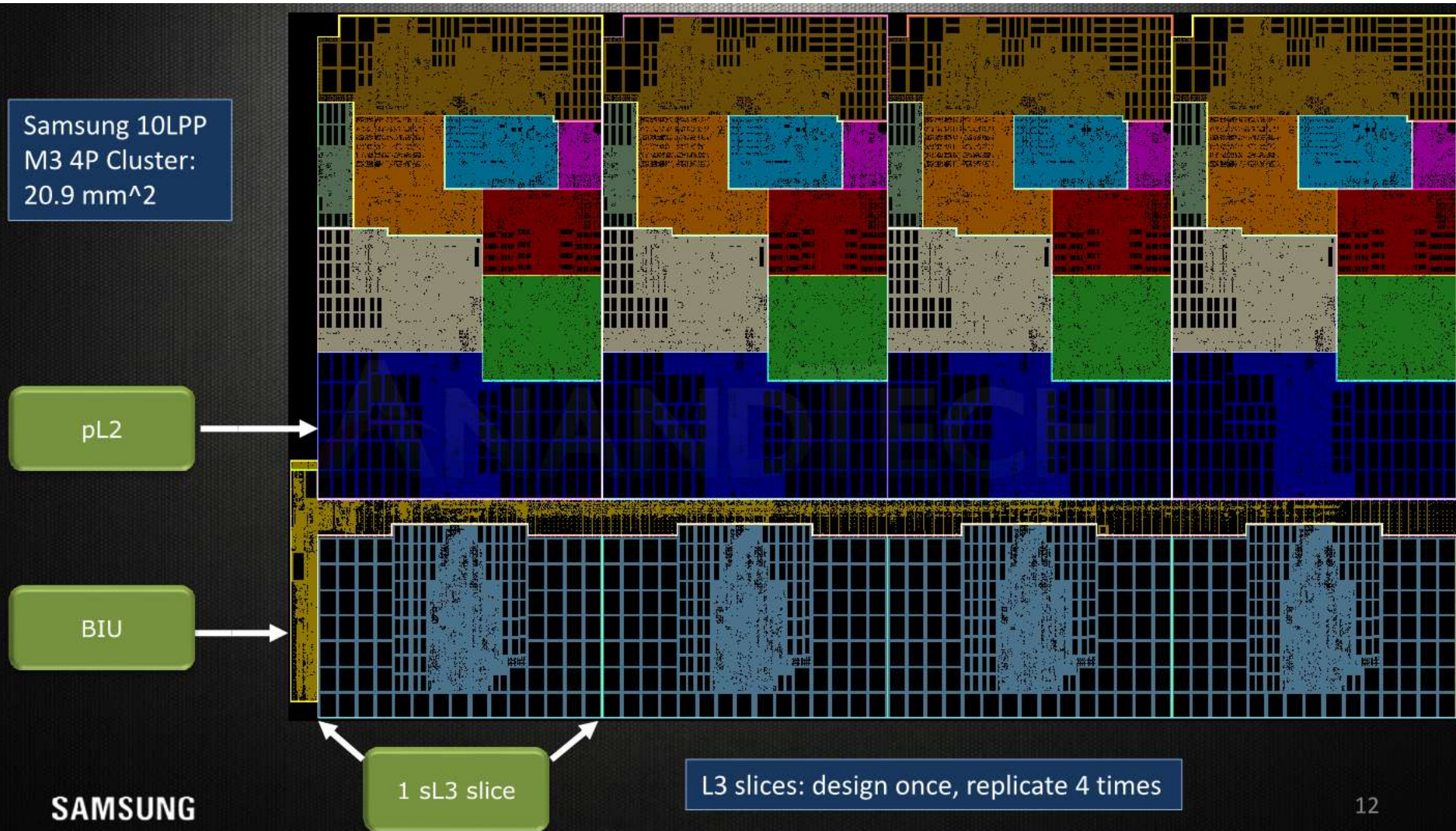
5.7.2 Microarchitecture of the M3 core (11)

M3 Core Layout [97]



5.7.2 Microarchitecture of the M3 core (12)

Exynos 9810 Floor Plan [79]



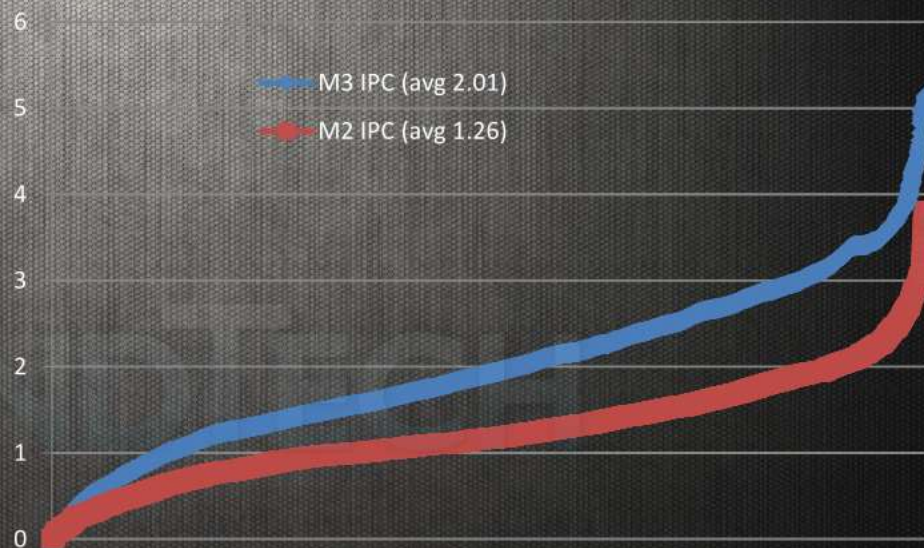
M2/M3 IPC values across ~4800 traces []

Performance Infrastructure

Dedicated performance team ran comprehensive simulations to guide tradeoffs across the design.

~4800 traces including:
Spec (2K/2K6, Int/FP), GBv4, Antutu, Octane, Sunspider, Bbench, browsermark, and more.

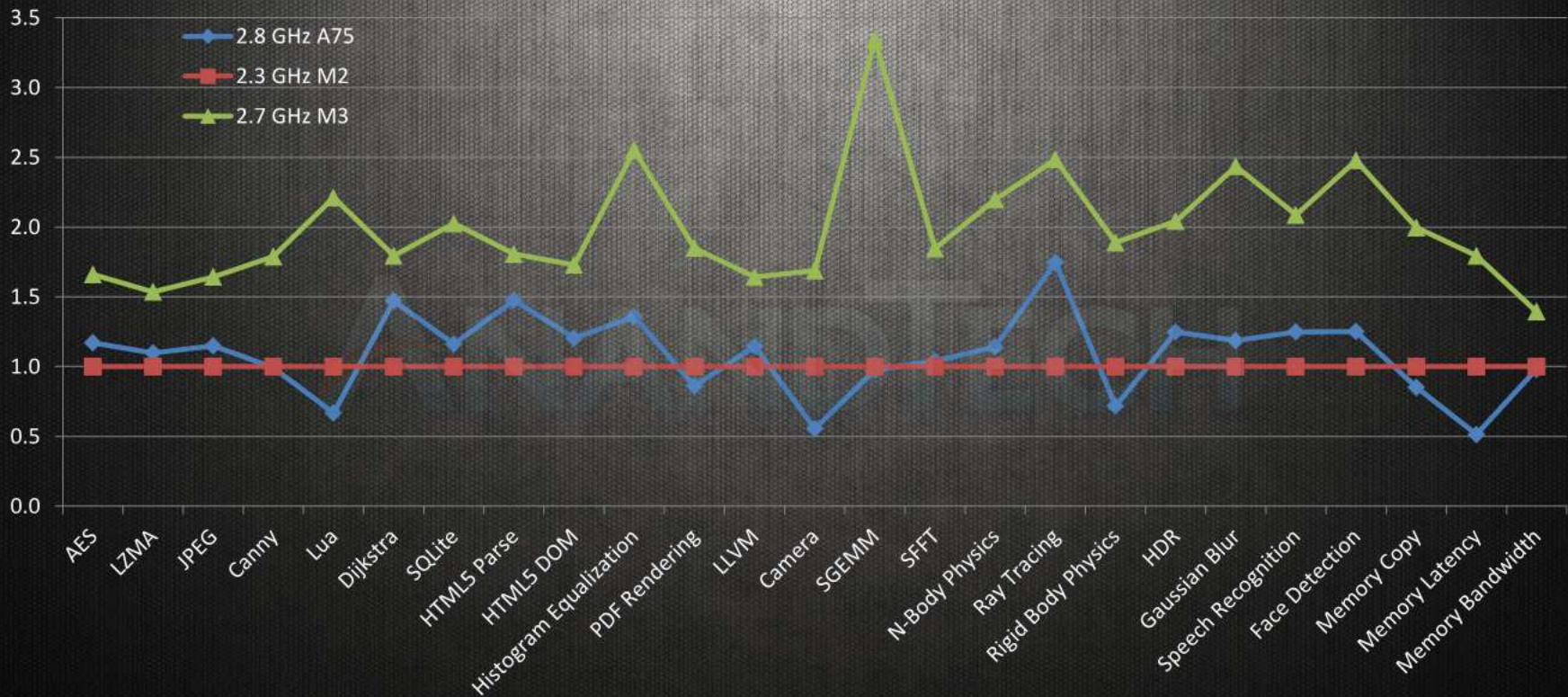
Correlation team ran hundreds of execution snippets across RTL and model to find design mistakes and improve prediction accuracy. Emulator team provided additional support by running long term simulations – found branch predictor “leak” this way.



Simulated IPC comparison across ~4800 unweighted traces; Both M2 and M3 sorted here; IPC will vary across applications/benchmarks

5.7.2 Microarchitecture of the M3 core (14)

Single thread performance of A75 and M3 relative to M2 while running GeekBench 4 [79]



New level of performance established for the Android eco-system

SAMSUNG

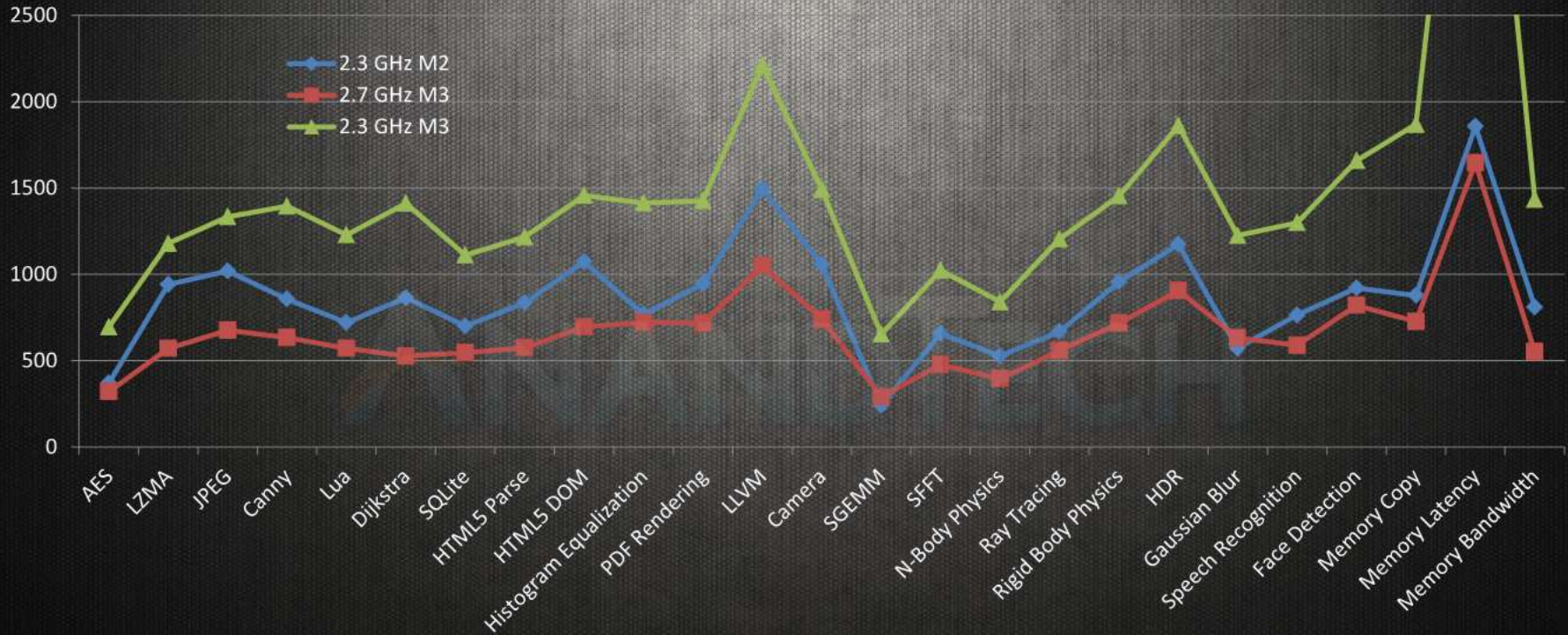
13

It represents commercial performance in the mobiles:
Exynos 8895 (M2), Exynos 9810 (M3) and the Snapdragon 845 (A75).

5.7.2 Microarchitecture of the M3 core (15)

Single thread performance/W of M2 and M3 while running GeekBench 4 [79]

Silicon Perf/W Comparison – GB4 single



Perf/Power Efficiency superior for M3 at iso-frequency with M2;
Efficiency in 1P frequency boost mode in-range

Evolution of M3 [79]

New CPU every year

M3 – 2018

M2 – 2017

M1 – 2016

Samsung M3

- Next Gen Planning Started - Q2 2014
- RTL Start - Q1 2015
- Fork features for incremental M2 - Q3 2015
- Replan for a bigger M3 push - Q1 2016
- Tapeout EVT0 - Q1 2017
- Product Launch: Q1 2018

Team now on strong annual cadence: expect more improvements every year

5.7.2 Microarchitecture of the M3 core (17)

Remarks to the developments of M1-M3 (taken from [79])

“Samsung’s CPU IP is developed in Austin, Texas, at “Samsung’s Austin R&D Center”, or SARC. The centre was founded in 2010 with the goal of establishing in-house IP for Samsung’s S.LSI division and Exynos chipsets. Staffed with ex-AMD, ex-Intel and various other talented industry veterans, what we saw come out - alongside memory controllers and custom interconnects - was also the of course more visible IPs: Samsung’s first custom CPUs.

The Exynos M1 is said to have started its design cycle sometime in 2012 and saw a quite short 3 year development phase, starting from scratch to first tape-out. It made its first appearance in the Exynos 8890 in the 2016 Galaxy S7. Over the years SARC has been expanding, and in 2017 the Advanced Computing Lab (ACL) in San Jose was opened and added to the SARC’s joint charter – adding custom GPU IP to its design portfolio that we hope to see productised in a couple of years.

The Exynos M1 being designed from scratch, it’s natural to expect that follow-up generations would be using it as the starting point for further development. Following the tape-out of the M1, the SARC team started off the M3 design with the existing M1 RTL back in Q1 of 2015. At first, this was meant to be an incremental development. However, there was a larger change of plans later on in Q1 2016, as goals were set higher for a much larger performance push.

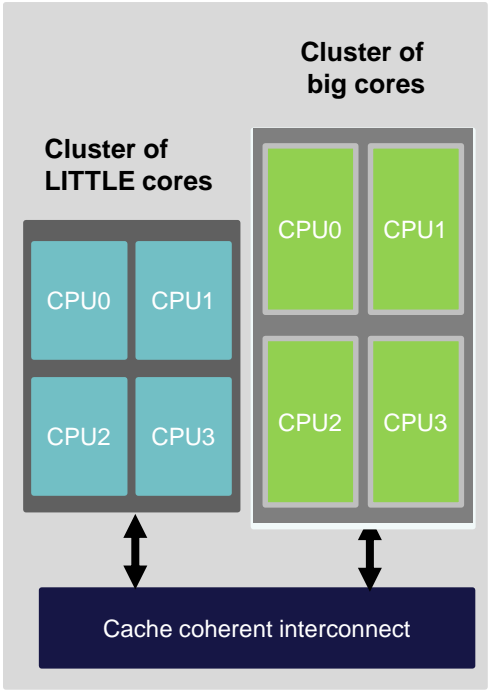
The existing improvements were forked in Q3 2015 into what became the M2 – which was initially meant to only be a 10LPE port of the M1 (Which was 14LPP). As a reminder, the M2 had a robust ~20% IPC improvement across workloads, which allowed it to outperform the M1 even though it was clocked 12% slower in production silicon. Samsung had achieved this by implementing some of the originally planned M3 features into the M2, while the new M3 design became more aggressive.”

5.7.3 The DynamIQ technology as an evolution of the big.LITTLE technology

5.7.3 The DynamIQ technology an evolution of the big.LITTLE technology (1)

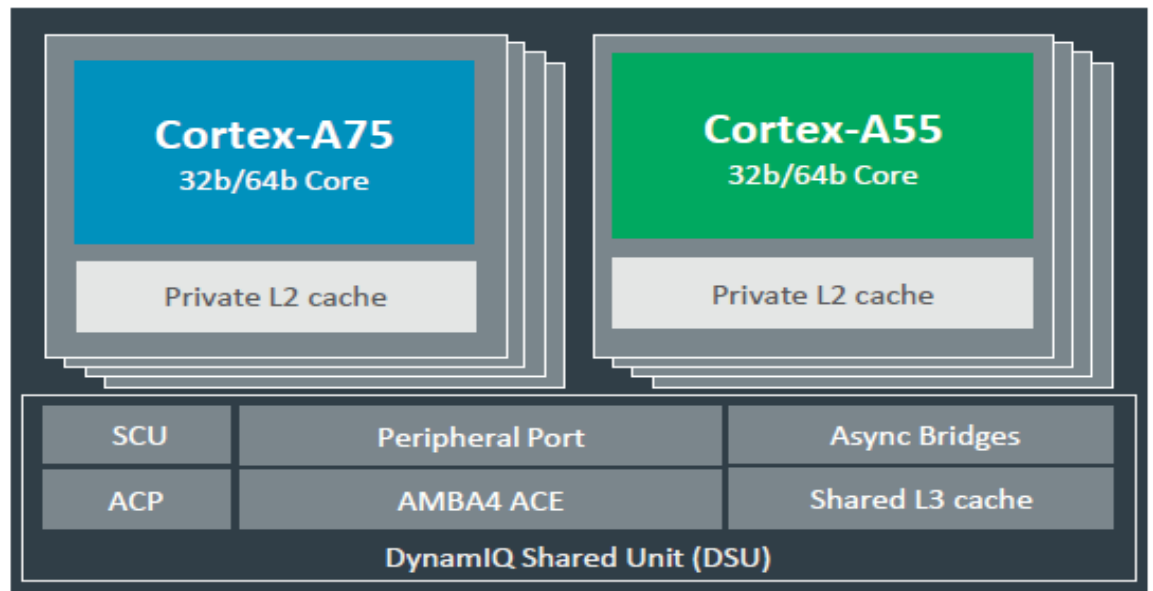
5.7.3 The DynamIQ technology as an evolution of the big.LITTLE technology

Two big.LITTLE core clusters



Two stand alone clusters with up to 4 cores (2011)

A single DynamIQ core cluster (65)



To cache coherent interconnect through the AMBA4 ACE bus



1b+7L



2b+6L



4b+4L



1b+2L



1b+3L



1b+4L

A single cluster of up to 8 cores of up to two core types (but up to 4 big cores) (2017)

5.7.3 The DynamIQ technology an evolution of the big.LITTLE technology (2)

Benefits of the DynamIQ cluster technology:

- support of the v8.2 ISA
- greater flexibility
- redesigned memory subsystem with higher bandwidth and lower access time
- improved power efficiency through intelligent power management, called **EAS (Energy Aware Scheduling)** (not discussed).

5.7.3 The DynamIQ technology an evolution of the big.LITTLE technology (3)

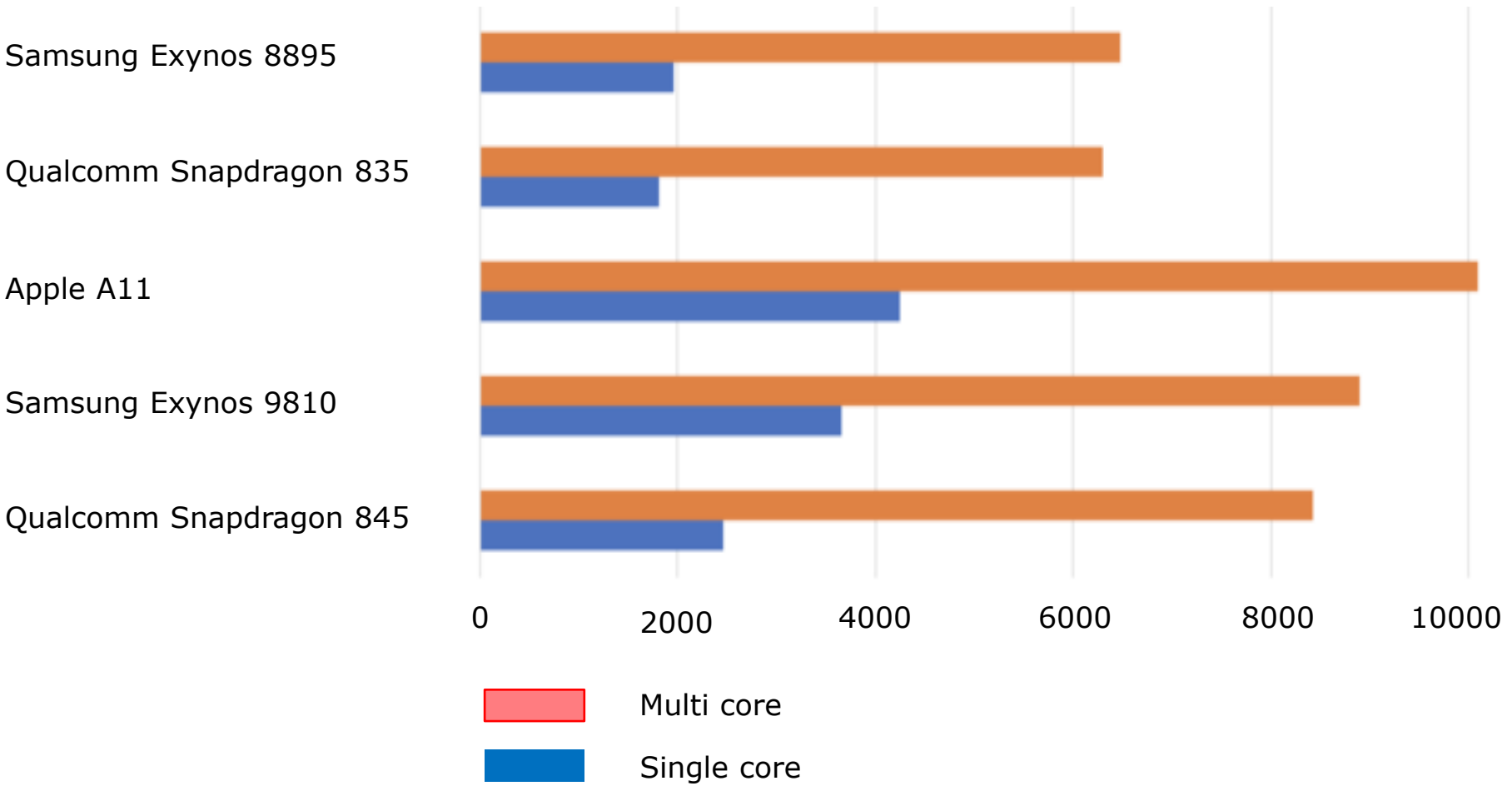
Main enhancements of DynamIQ core clusters

- a) Up to 8 CPU cores of up to 2 ARMv8.2 ISA based core types
- b) Private (per-core) L2 caches in the CPU cores
- c) DynamIQ Shared Unit (DSU) with a shared L3 cache and snoop filter
- d) Capability for partitioning the cores and the L3 cache
- e) Finer-grain frequency and voltage control
- f) Mesh interconnect (CMN-600) for server systems
- g) Use of a scratch pad system cache to increase throughput
- h) Cache stashing

(Above features are discussed in the Chapter: ARM processor lines).

5.7.3 The DynamIQ technology an evolution of the big.LITTLE technology (4)

Geekbench 4 scores [70]



Geekbench is a cross-platform benchmark that simulates real-world scenarios.
Geekbench 4 scores against a baseline of 4000 provided by Intel's Core i7-6600U @ 2.60 GHz.

6. References

6. References (1)

- [1]: Se-Hyung Y. et al., A 32nm high-k metal gate application processor with GHz multi-core CPU, ISSCC, 2012
- [2]: Kim M., Kim H., Chung H., Lim K., Samsung Exynos 5410 Processor – Experience the Ultimate Performance and Versatility, White Paper, 2013
- [3]: Shin Y., Shin K., Kenkare P., Kashyap R., 28nm high- metal-gate heterogeneous quad-core CPUs for high-performance and energy-efficient mobile application processor, IEEE, 2013
- [4]: Greenhaigh P., Big.LITTLE processing with ARM Cortex-A15 & Cortex-A7, EE Times, Oct. 24 2011, http://www.eetimes.com/document.asp?doc_id=1279167&page_number=2
- [5]: Jeff B., big.LITTLE Technology Moves Towards Fully Heterogeneous Global Task Scheduling, ARM TechCon, Nov. 2013, http://community.arm.com/servlet/JiveServlet/previewBody/7763-102-1-12076/big.LITTLE%20technology%20moves%20towards%20fully%20heterogeneous%20Global%20Task%20Scheduling_final%20%28pdf%29.pdf
- [6]: Wathan G., ARM big.LITTLE Technology Unleashed, ARM, https://www.arm.com/files/event/A3_big.LITTLE_Technology_Unleashed.pdf
- [7]: Мобильные процессоры. Компания Samsung, 4PDA, Jan. 21 2014, <http://4pda.ru/2014/01/21/136624/>
- [8]: Triggs R., A closer look at the Galaxy S6's Exynos 7420 SoC, Android Authority, <http://www.androidauthority.com/samsung-exynos-7420-closer-look-592117/>
- [9]: Smith C., These are the Galaxy S5's next-gen processors, BGR, Febr. 26 2014, <http://bgr.com/2014/02/26/galaxy-s5-processor-snapdragon-801-exynos-5422/>

6. References (2)

- [10]: Exynos 5 Hexa, Samsung, <http://www.samsung.com/global/business/semiconductor/product/application/detail?productId=7979&iaId=2341>
- [11]: Exynos 5 Octa, Samsung, http://www.samsung.com/global/business/semiconductor/minisite/Exynos/w/solution.html#?v=octa_5430
- [12]: Jonnalagadda H., Samsung announces 64-bit Exynos 7 Octa with significant performance improvements, Android Central, Oct. 16 2014, <http://www.androidcentral.com/samsung-officially-announces-64-bit-exynos-7-octa-significant-performance-improvements>
- [13]: Frumusanu A., Samsung's Exynos 5433 is an A57/A53 ARM SoC, AnandTech, Sept. 16 2014, <http://www.anandtech.com/show/8537/samsungs-exynos-5433-is-an-a57a53-arm-soc>
- [14]: Daniel P., Note 4 with octa-core Exynos 5433 crosses the 40 000 mark on AnTuTu, beats Snapdragon 805, Phone Arena, June 23 2014, http://www.phonearena.com/news/Note-4-with-octa-core-Exynos-5433-crosses-the-40-000-mark-on-AnTuTu-beats-Snapdragon-805_id57393
- [15]: Lee H-J., Shin Y., Bae S., Kim M., Kim K., 20nm High-K Metal Gate Heterogeneous 64-bit Quad-core CPUs and Hexa-core GPU for High performance and Energy-efficient Mobile Application Processor, ISOCC, 2015
- [16]: Frumusanu A., The Samsung Exynos 7420 Deep Dive - Inside A Modern 14nm SoC, AnandTech, June 29 2015, <http://www.anandtech.com/show/9330/exynos-7420-deep-dive>
- [17]: National Semiconductor PowerWise Adaptive Voltage Scaling Technology, 2012, <http://www.ti.com/lit/wp/snvy007/snvy007.pdf>

6. References (3)

- [18]: Advanced Power Controller – APC1, Rev. r0p0, IP Product Description
- [19]: Riemenschneider F., Samsungs Exynos 7420 unter der Lupe, CRN, Aug. 20 2015, <http://www.crn.de/telekommunikation/artikel-107635-3.html>
- [20]: Frumusanu A., Samsung Announces Exynos 8890 with Cat.12/13 Modem and Custom CPU, AnandTech, Nov. 12 2015, <http://www.anandtech.com/show/9781/samsung-announces-exynos-8890-with-cat1213-modem-and-custom-cpu>
- [21]: Frumusanu A., Early Exynos 8890 Impressions And Full Specifications, AnandTech, Febr. 21 2016, <http://www.anandtech.com/show/10075/early-exynos-8890-impressions>
- [22]: Burgess B., Samsung Exynos M1 Processor, Hot Chips 2016, https://www.hotchips.org/wp-content/uploads/hc_archives/hc28/HC28.22-Monday-Epub/HC28.22.20-Moble-Epub/HC28.22.220-ExynosM1-BradBurgess-SAMSUNG-FINAL.pdf
- [23]: Vintan L.N., Towards a High Performance Neural Branch Predictor, Proceedings of The International Joint Conference on Neural Networks - IJCNN '99, USA, July 10-16 1999, <https://pdfs.semanticscholar.org/7d78/eb5cc9b9e136f914f1c276d52176b6e83bc4.pdf>
- [24]: Jiménez D.A., Lin C., Dynamic Branch Prediction with Perceptrons, HPCA 2001, <https://www.cs.utexas.edu/~lin/papers/hpca01.pdf>
- [25]: The Journal of Instruction-Level Parallelism, 5th JILP Workshop on Computer Architecture Competitions (JWAC-5): Championship Branch Prediction (CBP-5)
- [26]: Jiménez D.A., Multiperspective Perceptron Predictor with TAGE, <https://www.jilp.org/cbp2016/paper/DanielJimenez2.pdf>

6. References (4)

- [27]: Dundas J., <https://www.linkedin.com/in/james-dundas-a6998a/>
- [28]: Burgess B., Chief architect of Bobcat, Chief CPU Architect, Samsung Austin R&D Center, Aug. 2011 – Pr, <https://www.linkedin.com/in/brad-burgess-093aa926>
- [29]: Goto H., ARM Cortex – A Family Architecture, 2010, <http://pc.watch.impress.co.jp/video/pcw/docs/423/409/p1.pdf>
- [30]: Exynos 9 Series (8895), A Mobile processor that goes beyond mobile innovation, http://www.samsung.com/semiconductor/minisite/Exynos/w/solution/mod_ap/8895/
- [31]: Smith R., Samsung Announces Exynos 8895 SoC: 10nm, Mali G71MP20, & LPDDR4x, AnandTech, Febr. 23 2017, <http://www.anandtech.com/print/11149/samsung-announces-exynos-8895-soc-10nm>
- [32]: Banerjee P., Exynos 8895 vs Snapdragon 835: What to expect from Samsung and Qualcomm this year, Digit, Febr. 24 2017, <http://www.digit.in/mobile-phones/exynos-8895-vs-snapdragon-835-what-to-expect-from-samsung-and-qualcomm-this-year-33885.html>
- [33]: Davies J., The Bifrost GPU architecture and the ARM Mali-G71 GPU, Hot Chips 28, Aug. 2016, https://www.hotchips.org/wp-content/uploads/hc_archives/hc28/HC28.22-Monday-Epub/HC28.22.10-GPU-HPC-Epub/HC28.22.110-Bifrost-JemDavies-ARM-v04-9.pdf
- [34]: Bifrost, Norse Mythology for Smart People, <http://norse-mythology.org/cosmology/bifrost/>
- [35]: Bratt I., The ARM Mali-T880 Mobile GPU, Hot Chips 27, 2015, https://www.hotchips.org/wp-content/uploads/hc_archives/hc27/HC27.25-Tuesday-Epub/HC27.25.50-GPU-Epub/HC27.25.531-Mali-T880-Bratt-ARM-2015_08_23.pdf

6. References (5)

- [36]: Smith R., ARM's Mali Midgard Architecture Explored, AnandTech, July 3 2014,
<http://www.anandtech.com/print/8234/arms-mali-midgard-architecture-explored>
- [37]: Vulkan, Khronos Group, <https://www.khronos.org/vulkan/>
- [38]: Nguyen T., AMD: Bringing "Torrenza" and "Fusion" Together, Daily Tech, March 17 2007,
<http://www.dailytech.com/article.aspx?newsid=6512>
- [39]: Rivas M., AMD Financial Analyst Day 2007, Dec. 14 2007
- [40]: Hruska J., AMD Fusion now pushed back to 2011, Ars Technica, Nov. 14 2008,
<http://arstechnica.com/uncategorized/2008/11/amd-fusion-now-pushed-back-to-2011/>
- [41]: Intel cans 45nm "Auburndale" and "Havendale" Fusion CPUs!, Jan. 31 2009,
<http://theovalich.wordpress.com/2009/01/31/exclusive-intels-cans-45nm-auburndale-and-havendale-fusion-cpus/>
- [42]: Blythe D., Technology Insight: Next Generation Intel Processor Graphics Architecture Code Name Skylake, IDF15
- [43]: Halfacree G., AMD ditches Fusion branding, Bit-Tech, Jan. 19 2012,
<http://www.bit-tech.net/news/hardware/2012/01/19/amd-ditches-fusion-branding/1>
- [44]: Papermaster M., Consumerization, Cloud, Convergence, AMD 2012 Financial Analyst Day, Febr. 2 2012
- [45]: Rogers P., Heterogeneous System Architecture Overview, Hot Chips Tutorial, Aug. 2013,
https://www.hotchips.org/wp-content/uploads/hc_archives/hc25/HC25.0T1-Hetero-epub/HC25.25.100-Intro-Rogers-HSA%20Intro%20HotChips2013_Final.pdf

6. References (6)

- [46]: HSA Foundation, <http://www.hsafoundation.com/>
- [47]: Smith R., HSA 1.1 Specification Launched: Extending HSA to More Vendors & Processor Types, AnandTech, May 31 2016, <http://www.anandtech.com/show/10387/hsa-11-specification-launched-multi-vendor-support>
- [48]: AMD and HSA, <http://www.amd.com/en-us/innovations/software-technologies/hsa>
- [49]: Kyriazis G., Heterogeneous System Architecture: A Technical Review, Rev. 1.0, Aug. 30 2012, <http://amd-dev.wpengine.netdna-cdn.com/wordpress/media/2012/10/hsa10.pdf>
- [50]: Moammer K., The One Vision That Intel, AMD And Nvidia Are All Chasing – Why Heterogeneous Computing Is The Future, WCCF Tech, Oct. 13 2015, <http://wccftech.com/intel-amd-nvidia-future-industry-hsa/2/>
- [51]: Zeller C., NVIDIA Tutorial CUDA 2008, <https://www.slideshare.net/angelamm2012/nvidia-cuda-tutorialnondaapr08>
- [52]: Chu H., AMD heterogeneous Uniform Memory Access, June 2013, http://events.csdn.net/AMD/GPUSat%20-%20hUMA_june-public.pdf
- [53]: Abazovic F., AMD's Naples Zen has 32 cores, Fudzilla, June 13 2016, <http://www.fudzilla.com/news/processors/40888-amd-naples-zen-has-32-cores>
- [54]: Hux A., 5th Generation Intel Core Processor Graphics: Gen8 Compute Architecture, IDF15, GVCS001

6. References (7)

- [55]: Junkins S., The Compute Architecture of Intel Processor Graphics Gen8, IDF 2014, <https://software.intel.com/sites/default/files/managed/71/a2/Compute%20Architecture%20of%20Intel%20Processor%20Graphics%20Gen8.pdf>
- [56]: Gasior G., AMD's heterogeneous queuing aims to make CPU, GPU more equal partners, Tech Report, October 22 2013, <http://techreport.com/news/25545/amd-heterogeneous-queuing-aims-to-make-cpu-gpu-more-equal-partners>
- [57]: Frumusanu A., Smith R., ARM A53/A57/T760 investigated - Samsung Galaxy Note 4 Exynos Review, AnandTech, Febr. 10 2015, <http://www.anandtech.com/show/8718/the-samsung-galaxy-note-4-exynos-review>
- [58]: Cutress I., AMD Gives More Zen Details: Ryzen, 3.4 GHz+, NVMe, Neural Net Prediction, & 25 MHz Boost Steps, AnandTech, Dec. 13 2016, <http://www.anandtech.com/show/10907/amd-gives-more-zen-details-ryzen-34-ghz-nvme-neural-net-prediction-25-mhz-boost-steps>
- [59]: Kirsch N., ARM Announces Mali-T760 GPU and Mali-T720 GPU, Legit Reviews, Oct. 29 2013, http://www.legitreviews.com/arm-announces-mali-t760-gpu-mali-t720-gpu_127366
- [60]: AMD Completes ATI Acquisition and Creates Processing Powerhouse, AMD, Oct. 25 2006
- [61]: Java Virtual Machine, Free Download Java Virtual Machine, How to Download JVM, Dev Manuals, Sept. 28 2010, <http://www.devmanuals.com/tutorials/java/corejava/javavirtualmachine.html>
- [62]: Tolman H., AMD Kaveri APU Architecture Overview, Jan. 14 2014, <http://benchmarkreviews.com/11622/amd-kaveri-apu-architecture-overview/>

6. References (8)

- [63]: Samsung's New Quad-core Application Processor Drives Advanced Feature Sets in Smartphones and Tablets (Designed on 32nm HKMG process), Samsung, April 26, 2012, http://www.samsung.com/semiconductor/minisite/Exynos/w/newsroom/press_release/Samsungs_New_Quad_core_Application_Processor_Drives_Advanced_Feature_Sets_in_Smartphones_and_Tablets/
- [64]: Merritt R., ARM stretches out with A5 core, graphics, FPGAs, Oct. 21 2009, <http://www.embedded.com/print/4085371>
- [65]: Triggs R., Everything you need to know about ARM's DynamIQ, Android Authority, May 29, 2017, <https://www.androidauthority.com/arm-dynamiq-need-to-know-770349/>
- [66]: Cho H.-D. et al., Benefits of the big.LITTLE architecture, Samsung, Febr. 2012
- [67]: Frumusanu A., The Samsung Exynos M3 - 6-wide Decode With 50%+ IPC Increase, AnandTech, Jan. 23 2018, <https://www.anandtech.com/show/12361/samsung-exynos-m3-architecture>
- [68]: Howse B., Frumusanu A., Samsung Announces New 9810 SoC: DynamIQ & 3rd Gen CPU, AnandTech, Jan. 3 2018, <https://www.anandtech.com/show/12212/samsung-announces-new-exynos-9810-soc>
- [69]: Do T., Samsung Exynos 9810 vs Qualcomm Snapdragon 845 – Which is the Better Processor for Samsung Galaxy S9?, Techwalls, Jan. 27 2018, <https://www.techwalls.com/samsung-exynos-9810-vs-qualcomm-snapdragon-845/>
- [70]: Kostadinov P., Samsung Galaxy S9 with Exynos 9810 on deck pops up in benchmark, humiliates Snapdragon 845, Febr. 13 2018, https://www.phonearena.com/news/Samsung-Galaxy-S9-S9-Plus-Exynos-9810-benchmarks-destroy-Snapdragon-845_id102444

6. References (9)

- [71]: List of Samsung System on Chips,
https://en.wikipedia.org/wiki/List_of_Samsung_System_on_Chips
- [72]: Chung H., Kang M., Cho H.-D., Heterogeneous Multi-Processing Solution of Exynos 5 Octa with ARM big.LITTLE Technology, Samsung, 2012,
https://s3.ap-northeast-2.amazonaws.com/global.semi.static/Heterogeneous_Multi-Processing_Solution_of_Exynos_5_Octa_with_ARM_bigLITTLE_Technology.pdf
- [73]: Exynos 9 Series (8895), Samsung,
<http://www.samsung.com/semiconductor/minisite/exynos/products/mobileprocessor/exynos-9-series-8895/>
- [74]: Versace M., Does Artificial Intelligence Require Specialized Processors?, TheNewsStack, 20 Oct. 2017,
<https://thenewstack.io/ai-hardware-software-dilemma>
- [75]: Hruska J., New Movidius Myriad X VPU Packs a Custom Neural Compute Engine, ExtremeTech, August 30, 2017,
<https://www.extremetech.com/computing/254772-new-movidius-myriad-x-vpu-packs-custom-neural-compute-engine>
- [76]: Brendan Barry B. et al., Always-on Vision Processing Unit for Mobile Applications, IEEE MICRO, March/April 2015, pp. 56-66
- [77]: Exynos 9 Series (9810), Samsung,
<http://www.samsung.com/semiconductor/minisite/exynos/products/mobileprocessor/exynos-9-series-9810/>
- [78]: GPU GFLOPS, <https://gflops.surge.sh/>

6. References (10)

- [79]: Frumusanu A., Hot Chips 2018: Samsung's Exynos-M3 CPU Architecture Deep Dive, AnandTech, Aug. 20, 2018, <https://www.anandtech.com/show/13199/hot-chips-2018-samsungs-exynosm3-cpu-architecture-deep-dive>