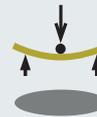
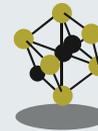
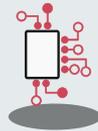


Chapter 1

Diversity of Processor Architectures



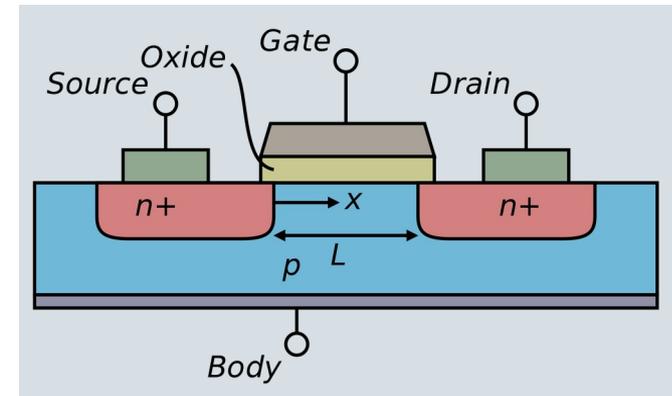
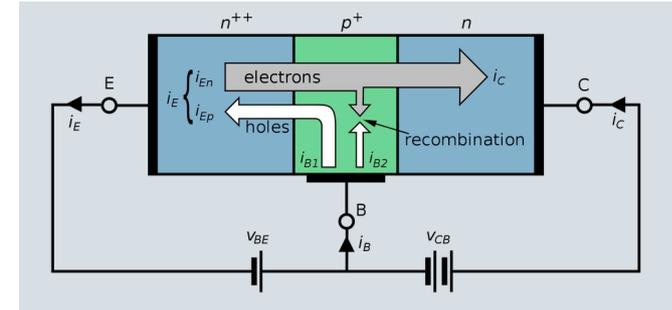
2021-2022

Quick reminder

1947: Invention of the **Bipolar Junction Transistor** →
by Bardeen, Schokley and Brattain (Bell labs), Nobel Prize winners

1958/1959: Creation of **Integrated Circuits**
by Texas Instruments (hybrid IC), then Fairchild (true monolithic IC)

1960: Invention of the **MOS Field-Effect Transistor** →
by Mohammed Atalla and Dawon Kahng

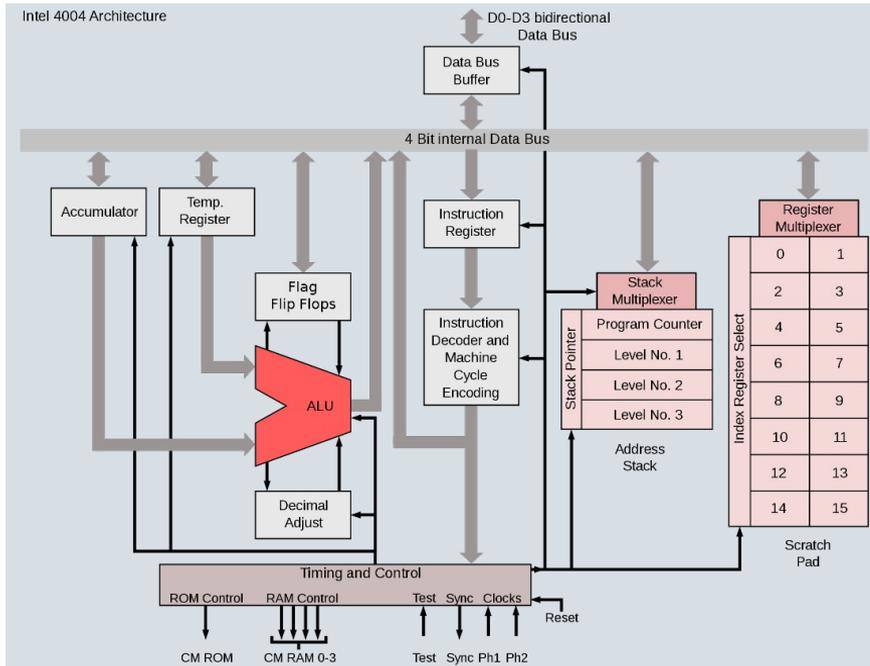


ON THE DIVERSITY OF PROCESSOR ARCHITECTURES

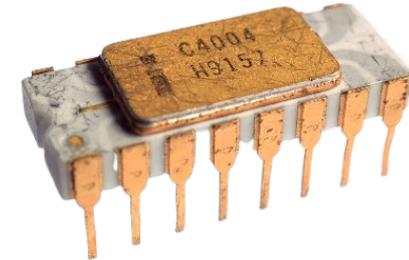
First processor

The first ever commercialised processor is the Intel 4004 in 1971.

It has 2,300 transistors with a 10 μm etching process (4-bit processor, 16 pins, 740 kHz, 90 kIPS or kilo-Instructions Per Second).



Intel 4004
integrated
circuit



Intel 4004 die



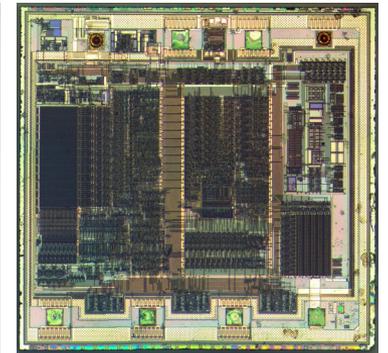
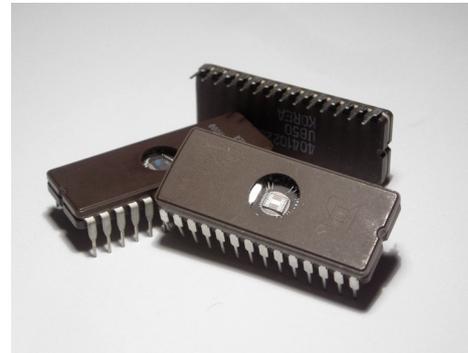
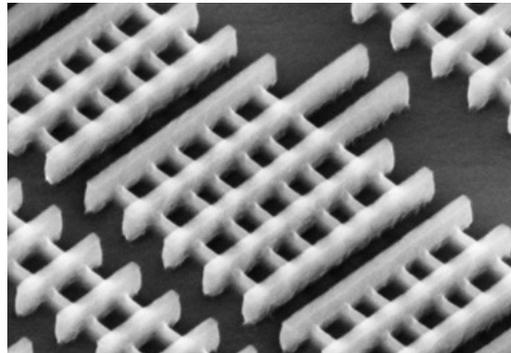
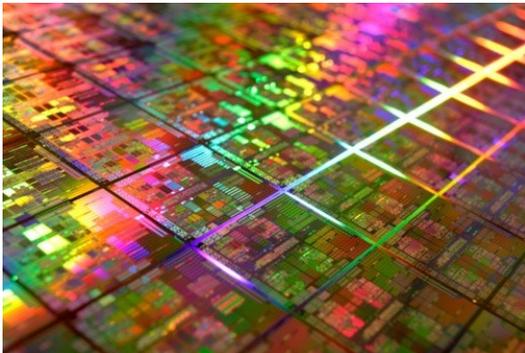
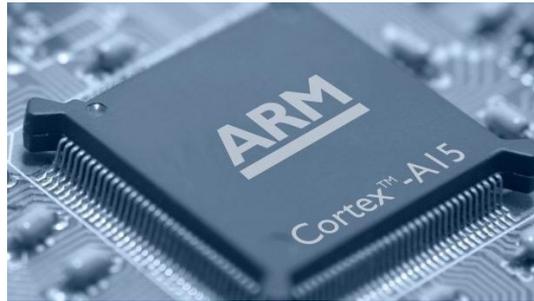
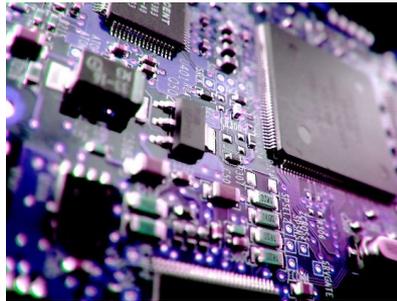
Intel 4004
architecture

ON THE DIVERSITY OF PROCESSOR ARCHITECTURES

Processors evolution

Ever since, processors have evolved following natural selection.

Those that matched specific needs improved while others disappeared from markets and research labs.



ON THE DIVERSITY OF PROCESSOR ARCHITECTURES

Processors evolution

As for animals and plants, the evolution process of processors is never-ending. New processor architectures are likely to be born in the next few years!



Let's take a look at the current processor architectures.

ON THE DIVERSITY OF PROCESSOR ARCHITECTURES

Common processor architectures

MCU

AP

GPP

SoC / SoB

FPGA

DSP

(GP) GPU

General architecture

Control processors

Hybrid architectures

Coprocessors or Calculus processors

MCU

Micro
Controller
Unit

AP

Application
Processor

GPP

General
Purpose
Processor

SoC / SoB

System
on
Chip / Board

- FPGA-AP
- FPGA-MCU
- GPP-GPU
- AP
- MCU-analog

FPGA

Field
Programmable
Gate Array

DSP

Digital
Signal
Processor

(GP) GPU

Graphics
Processing
Unit

General
Purpose
GPU

Computer →

CONTROL

CALCULUS

General architecture

Control processors

Hybrid architectures

Coprocessors or Calculus processors

MCU

Micro
Controller
Unit

AP

Application
Processor

CPU

GPP

General
Purpose
Processor

SoC / SoB

System
on
Chip / Board

- FPGA-AP
- FPGA-MCU
- GPP-GPU
- AP
- MCU-analog

FPGA

Field
Programmable
Gate Array

LOGIC

DSP

Digital
Signal
Processor

CPU

(GP) GPU

Graphics
Processing
Unit

General
Purpose
GPU

MCU

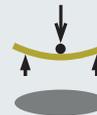
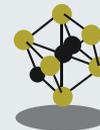
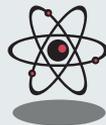
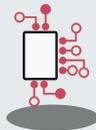
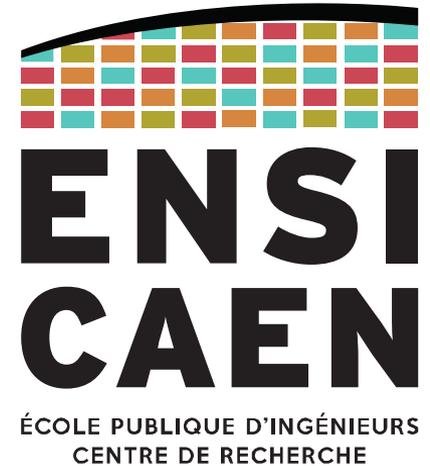
MICROCONTROLLER UNIT

Applications

Architectures

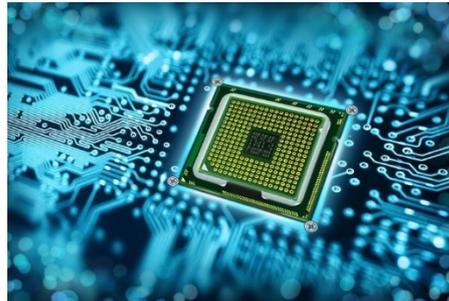
Designers and products

Market shares



MCUs (Microcontroller Units, fr: *micro-contrôleurs*) are the most common processors in our environment (talking about quantity).

We use about 200 processors every day, without even being aware!



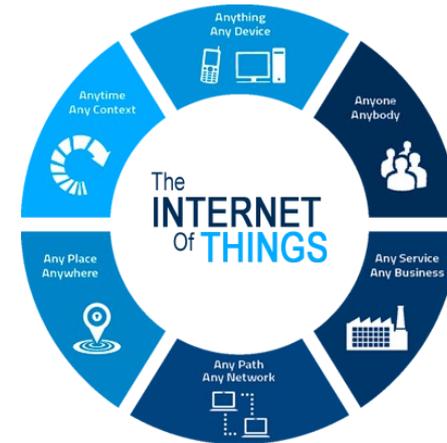
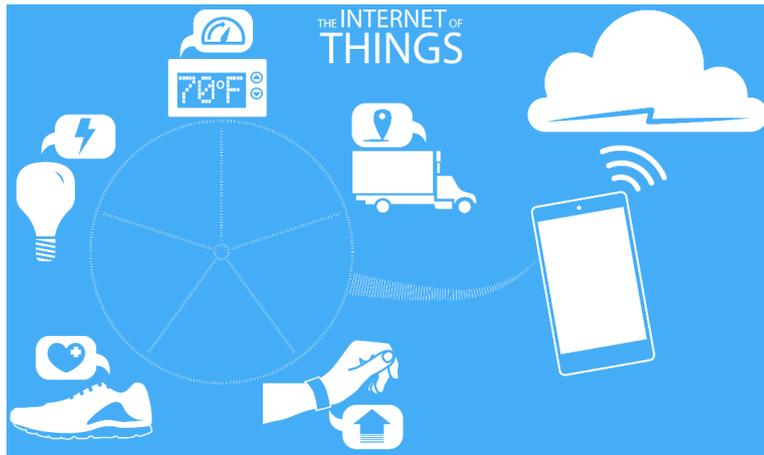
MCUs are control processors that are dedicated to the supervision of electronic processes. They control their input/output interfaces with their application-custom embedded firmware.

They aim for markets applications that require low-cost, low-consumption, small-size, and big production volumes.

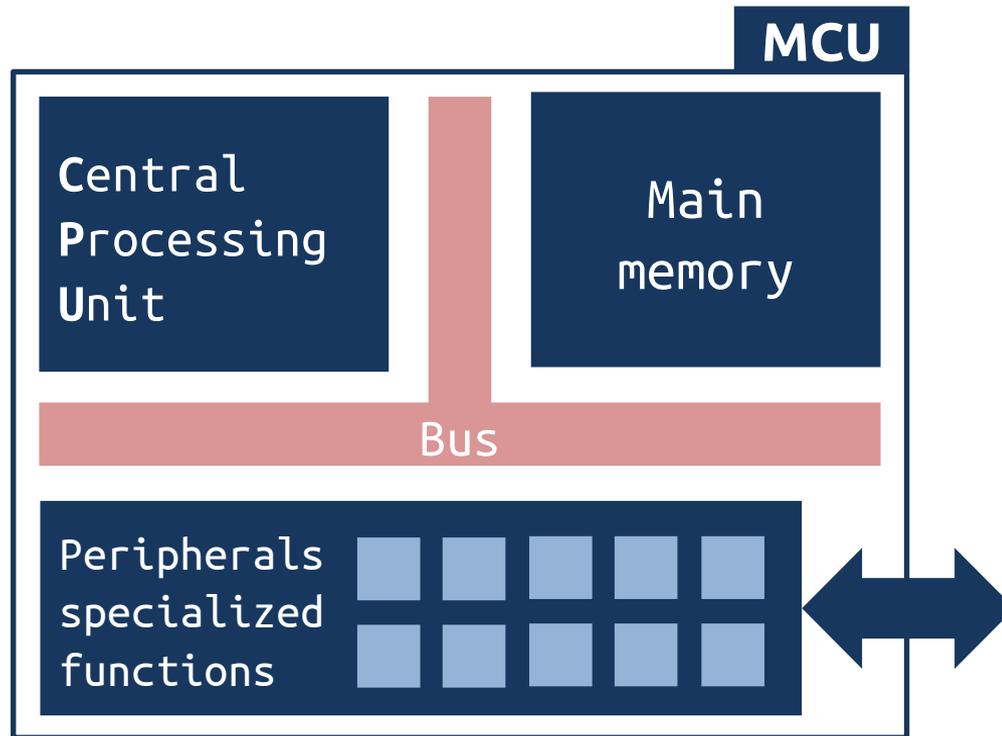


The IoT (Internet of Thing, fr: *objets connectés*) is the major market for MCUs. The IoT is the Internet extension to physical world objects and places. It is considered as the third Internet evolution and has been therefore named « Web 3.0 ».

With 3.6 billions of active connections in 2015, 11.7 billions in 2020 and 30 billions planned in 2025, the IoT counted for 18% of MCUs population in 2019 and will be around 29% in 2025.



MCU processors are digital systems integrated onto an Integrated Circuit.
They are designed to be stand-alone (no need for external RAM, HDD ...).

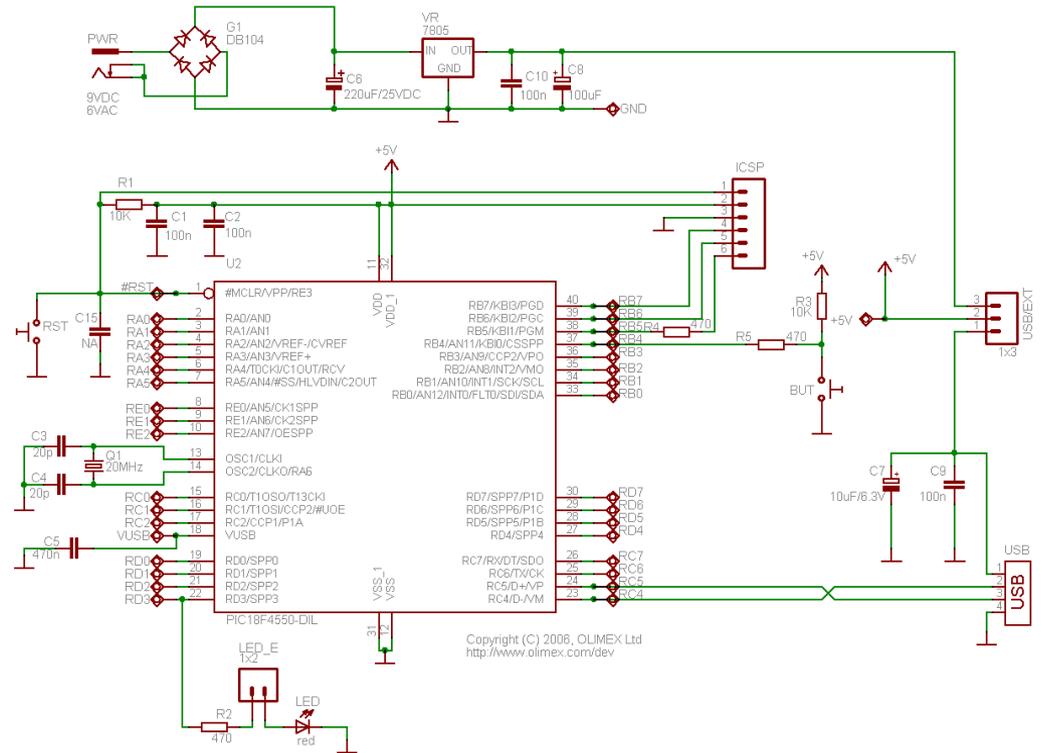
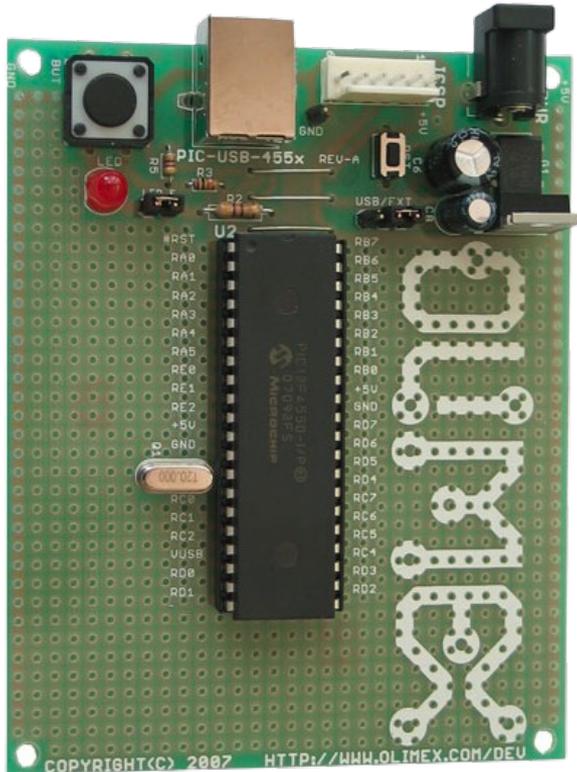


MCU – MICROCONTROLLER UNIT

Board and schematic

Example of a schematic that uses a Microchip's PIC18 MCU.

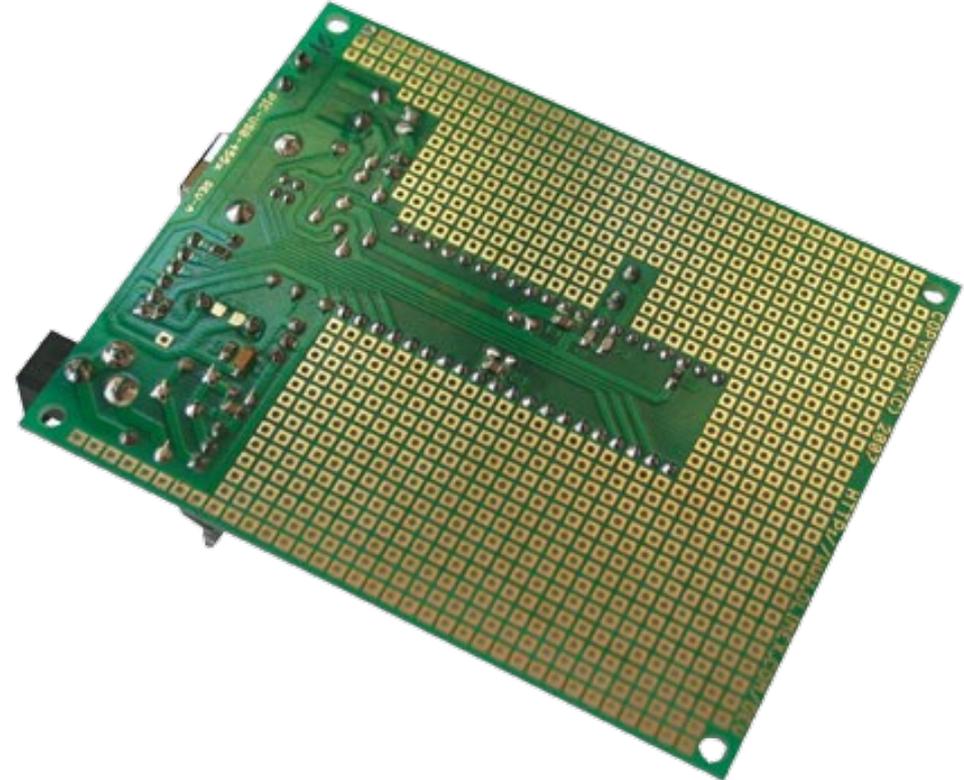
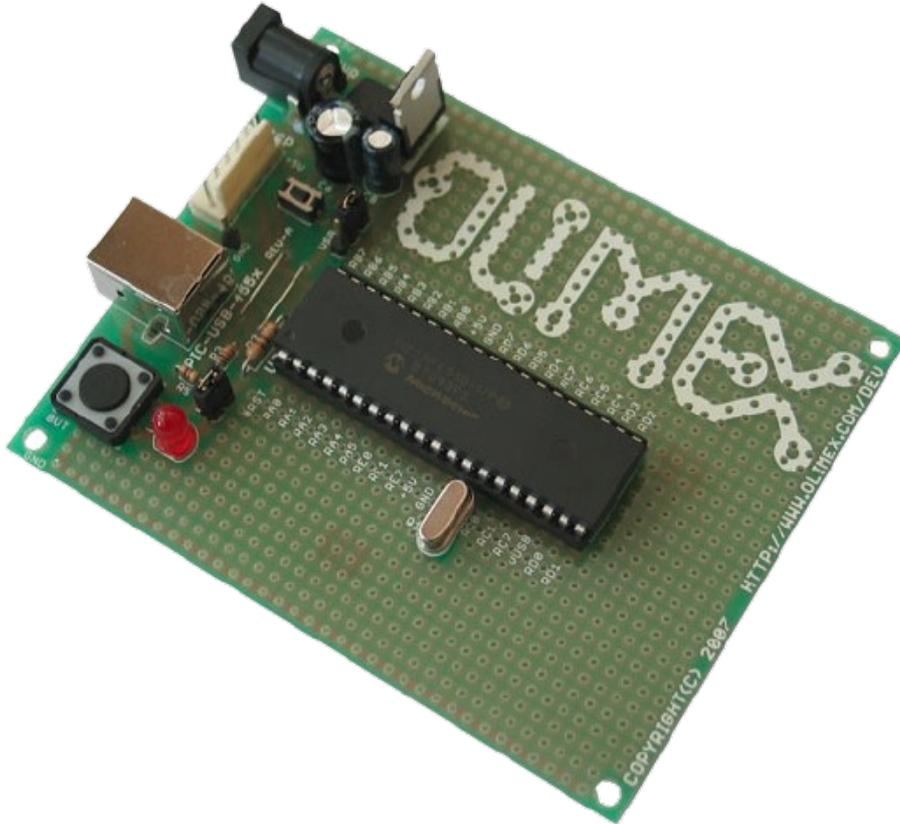
Olimex PIC-USB-4550 board.



MCU – MICROCONTROLLER UNIT

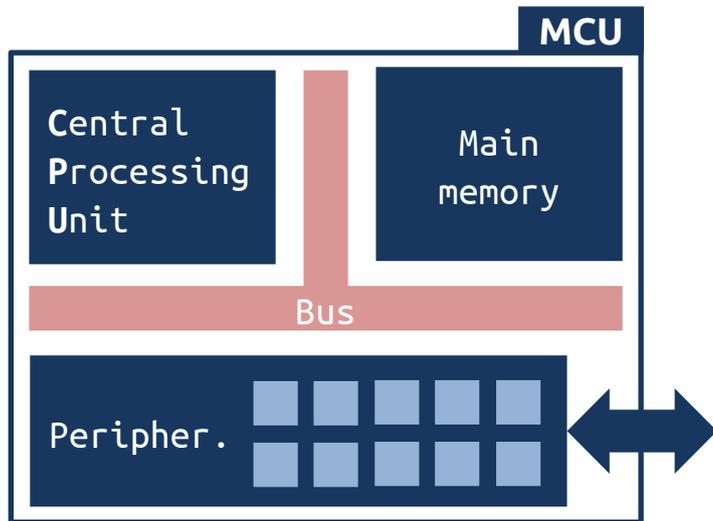
Board and schematic

Exercise: link these board devices to the schematic in the previous slide.

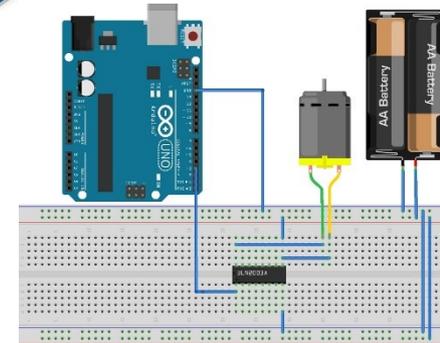
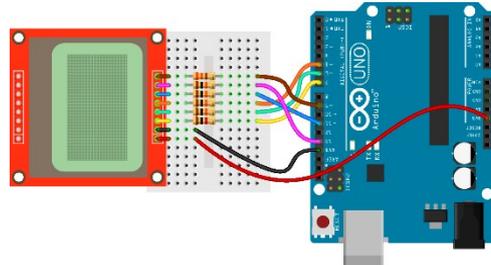
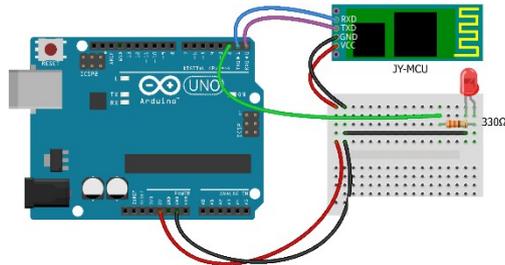
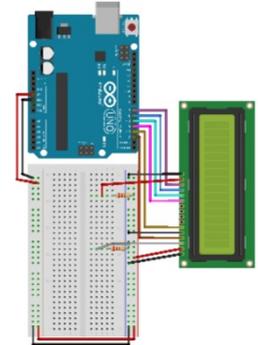


There is a big number of MCU products from various designers and foundries, each made for different uses.

MCUs from the same family possess the same CPU and associated buses. The **ISA (Instruction Set Architecture, fr: jeu d'instructions)** and the toolchain are therefore similar. The difference between same-family MCUs resides in the peripherals set and the memory resources.

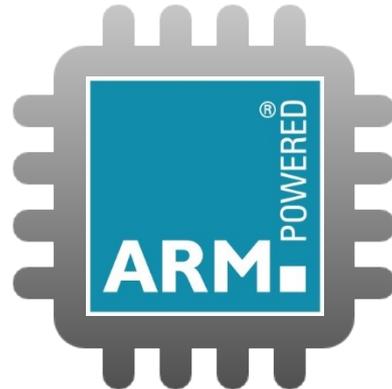


The Arduino project is certainly the most famous MCU-based electronic project. However it is too user-friendly (too magic, too many hidden things) and is not used in professional environments, which is why it is not studied in engineer schools.



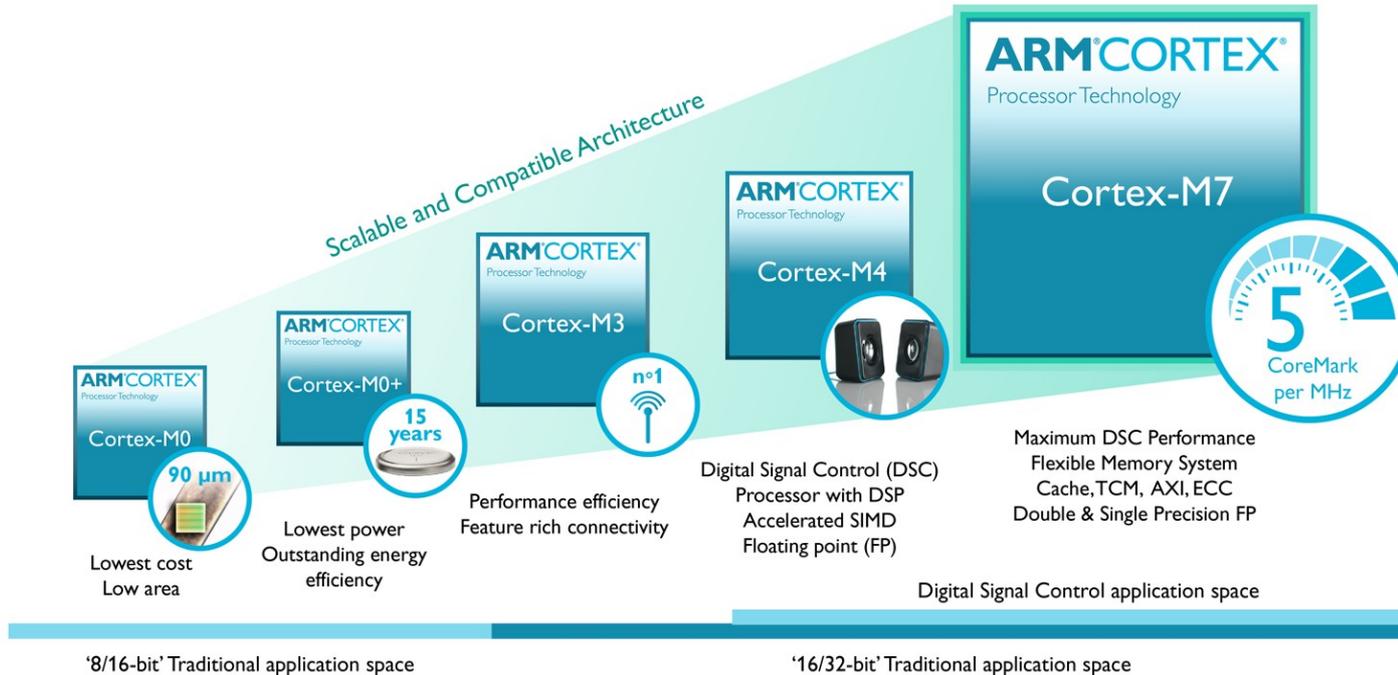
Even though the MCU market is very competitive, the vast majority of MCU founders (e.g. STMicroelectronics, Renesas, Texas Instruments, NXP, ...) use similar CPU architectures: the Cortex-M family, designed by the British company ARM

This guaranties an access to reliable development tools, libraries and software services. Some tools can also be open-source (IP / Graphical / USB / Bluetooth, stack, RTOS, ...).



ARM offers the Cortex-M series, with 'M' standing for "MCU".

This includes a whole family of MCU cores that are suitable for a wide range of applications.



arm
CORTEX® -M7

Nested vectored interrupt controller		Wake-up interrupt controller	
CPU ArmV7-M			
Memory protection unit		DSP	FPU
2x AHB-Lite	ITM trace	Data watchpoint	JTAG
	ETM trace	Breakpoint unit	Serial wire
I-cache	D-TCM	I-TCM	ECC
D-cache	AXI-M		

arm
CORTEX® -M0

Nested vectored interrupt controller		Wake-up interrupt controller	
CPU ArmV6-M			
AHB-Lite	Data watchpoint	JTAG	
	Breakpoint unit	Serial wire	

As an example let's take a look at the range of STM32. Those are 32-bit MCUs based on a Cortex-M core.

They are designed by the French-Italian company STMicroelectronics, which also is the main European manufacturer.



Common core peripherals and architecture:

Communication peripherals: USART, SPI, I ² C
Multiple general-purpose timers
Integrated reset and brown-out warning
Multiple DMA
2x watchdogs Real-time clock
Integrated regulator PLL and clock circuit
External memory interface (FSMC)
Up to 3x 12-bit DAC
Up to 4x 12-bit ADC (Up to 5 MSPS)
Main oscillator and 32 kHz oscillator
Low-speed and high-speed internal RC oscillators
-40 to +85 °C and up to 105 °C operating temperature range
Low voltage 2.0 to 3.6 V or 1.65/1.7 to 3.6 V (depending on series)
Temperature sensor

+

STM32 F4 series - High performance with DSP (STM32F405/415/407/417)

168 MHz Cortex-M4 with DSP and FPU	Up to 192-Kbyte SRAM	Up to 1-Mbyte Flash	2x USB 2.0 OTG FS/HS	3-phase MC timer	2x CAN 2.0B	SDIO 2x I ² S audio Camera IF	Ethernet IEEE 1588	Crypto/ hash processor and RNG
---	----------------------------	---------------------------	----------------------------	---------------------	----------------	---	-----------------------	---



STM32 F3 series - Mixed-signal with DSP (STM32F302/303/313/372/373/383)

72 MHz Cortex-M4 with DSP and FPU	Up to 48-Kbyte SRAM & CCM-SRAM	Up to 256-Kbyte Flash	USB 2.0 FS	2x 3-phase MC timer (144 MHz)	CAN 2.0B	Up to 7x comparator	3x 16-bit ΣΔ ADC	4x PGA
--	---	-----------------------------	---------------	--	-------------	---------------------------	---------------------	--------



STM32 F2 series - High performance (STM32F205/215/207/217)

120 MHz Cortex-M3 CPU	Up to 128-Kbyte SRAM	Up to 1-Mbyte Flash	2x USB 2.0 OTG FS/HS	3-phase MC timer	2x CAN 2.0B	SDIO 2x I ² S audio Camera IF	Ethernet IEEE 1588	Crypto/ hash processor and RNG
-----------------------------	----------------------------	---------------------------	----------------------------	---------------------	----------------	---	-----------------------	---



STM32 F1 series - Mainstream - 5 product lines (STM32F100/101/102/103 and 105/107)

Up to 72 MHz Cortex-M3 CPU	Up to 96-Kbyte SRAM	Up to 1-Mbyte Flash	USB 2.0 OTG FS	3-phase MC timer	Up to 2x CAN 2.0B	SDIO 2x I ² S audio	Ethernet IEEE 1588
-------------------------------------	---------------------------	---------------------------	-------------------	---------------------	-------------------------	--------------------------------------	-----------------------



STM32 F0 series – Entry level (STM32F050/051)

48 MHz Cortex-M0 CPU	Up to 12-Kbyte SRAM	Up to 128-Kbyte Flash	3-phase MC timer	Comparator	CEC
----------------------------	---------------------------	-----------------------------	---------------------	------------	-----



STM32 L1 series - Ultra-low-power (STM32L151/152/162)

32 MHz Cortex-M3 CPU	Up to 48-Kbyte SRAM	Up to 384-Kbyte Flash	USB FS device	Up to 12-Kbyte EEPROM	LCD 8x40 4x44	Comparator	BOR MSI VScal	AES 128-bit
----------------------------	---------------------------	-----------------------------	------------------	-----------------------------	---------------------	------------	---------------------	----------------

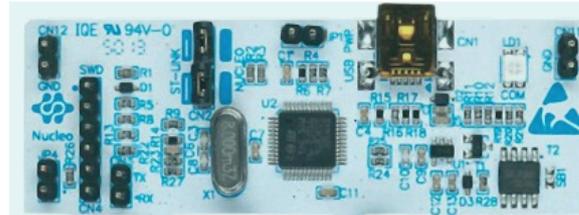
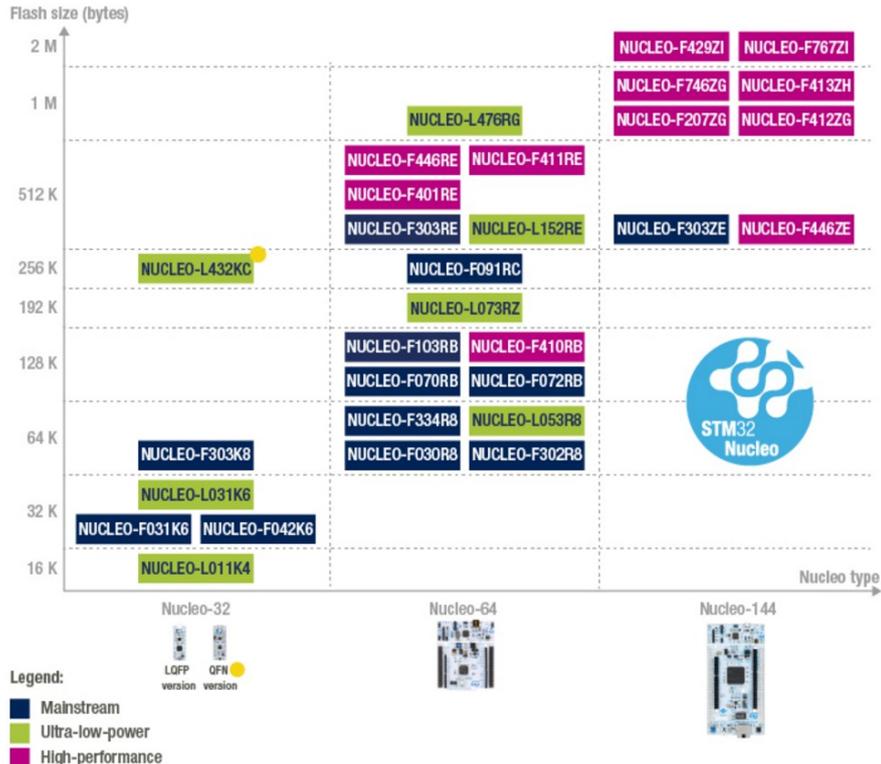


STM32 W series - Wireless (STM32W108)

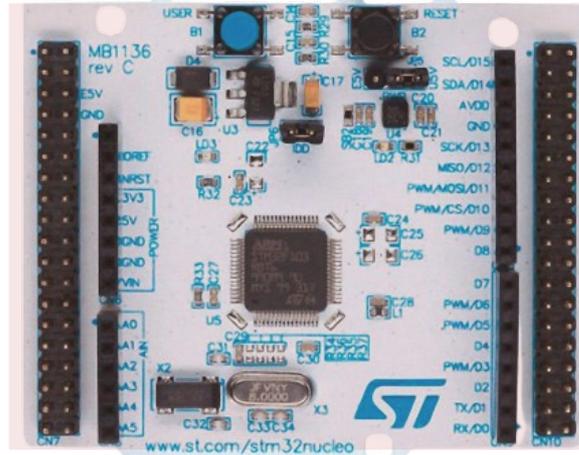
24 MHz Cortex-M3 CPU	Up to 16-Kbyte SRAM	Up to 256-Kbyte Flash	2.4 GHz IEEE 802.15.4 Transceiver	Lower MAC Digital baseband	AES 128-bit
----------------------------	---------------------------	-----------------------------	--	----------------------------------	----------------



The STMicroelectronics Nucleo project offers low-cost (\approx €10) evaluation boards that use ARM-based MCUs and industrial development tools.



- Power supply
- Programmer (JTAG emulator)



- Target MCU
- Switch and LED
- External ports
- Shields connectors
- Arduino shield connectors

Nucleo-64

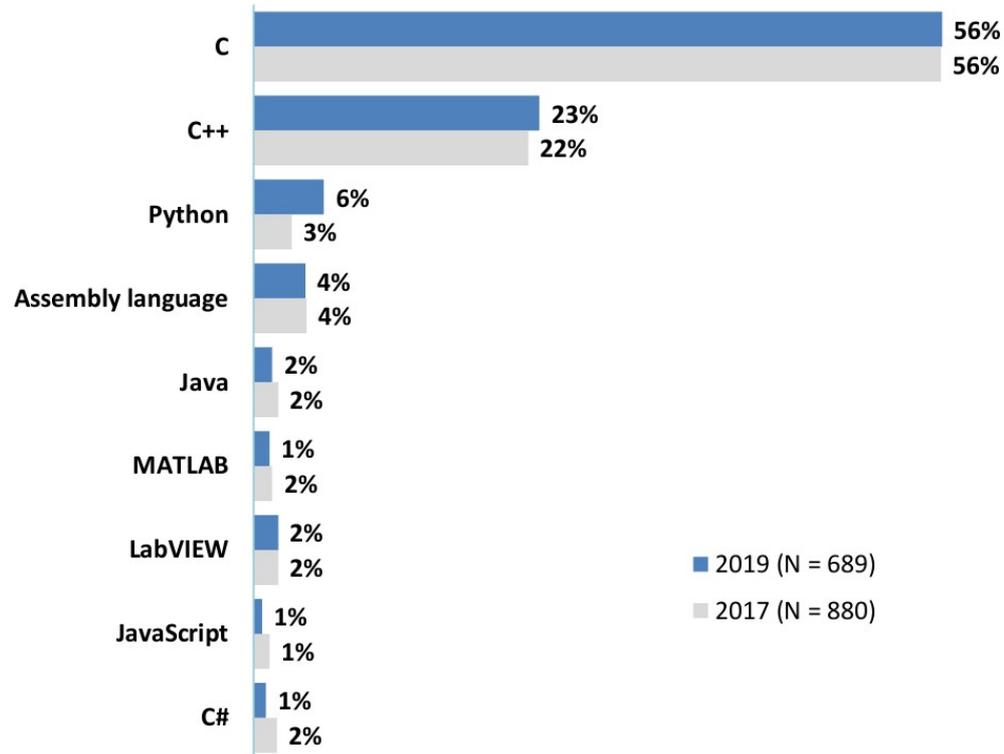
Let's take a look at an annual markets study.



2019 Embedded Markets Study
Integrating IoT and Advanced Technology Designs,
Application Development & Processing Environments
March 2019

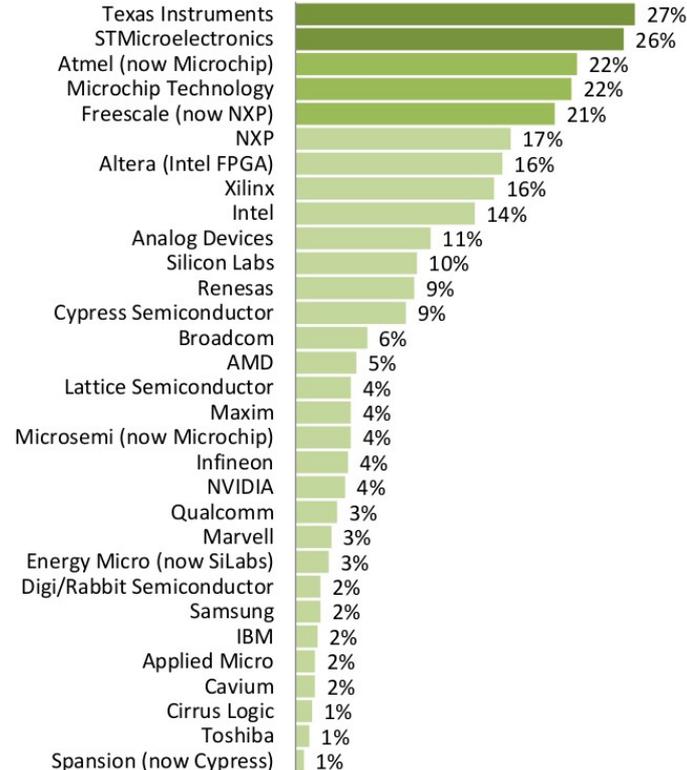
Presented By: **EE**Times embedded

My *current* embedded project is programmed mostly in:



Market shares

Please select the processor vendors you are currently using.



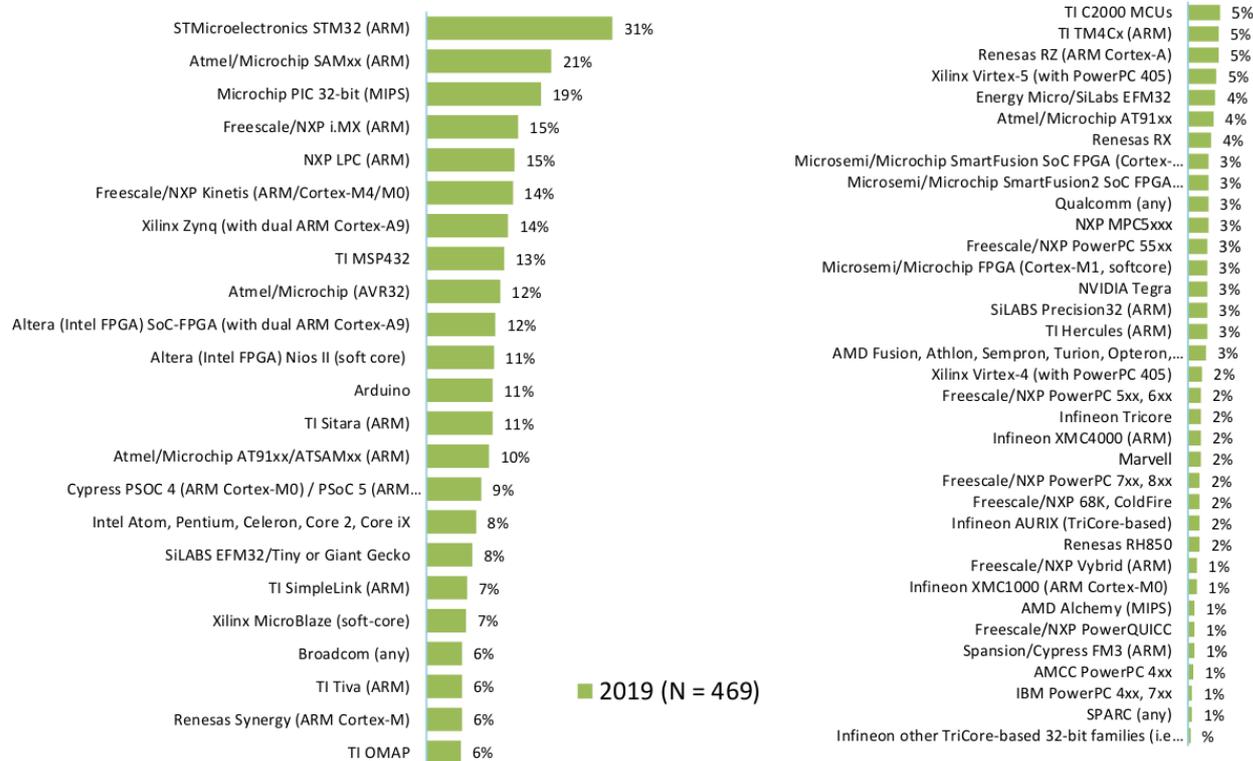
Merged Brands Combined	%
Microchip/Atmel/Microsemi (Net)	40
NXP/Freescale (Net)	28
Intel/Altera (Net)	26
Silicon Labs/Energy (Net)	10
Cypress/Spansion (Net)	9

Top Four Brands by Region:
Americas: TI, Microchip, STMicro, Atmel
EMEA: STMicro, NXP, TI, Atmel
APAC: TI, Atmel, Freescale, STMicro

2019 (N = 458)

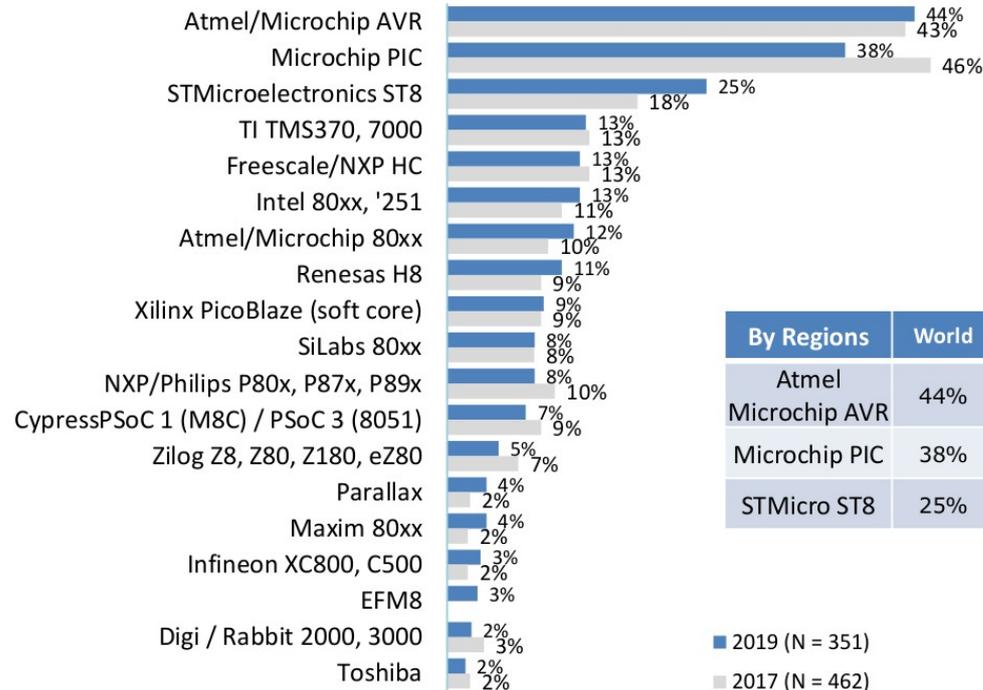


Which of the following 32-bit chip families would you consider for your next embedded project?





Which of the following 8-bit chip families would you consider for your next embedded project?



By Regions	World	Americas	EMEA	APAC
Atmel	44%	44%	52%	39%
Microchip AVR	38%	41%	43%	23%
STMicro ST8	25%	22%	31%	28%

■ 2019 (N = 351)

■ 2017 (N = 462)

GPP

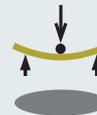
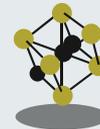
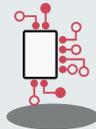
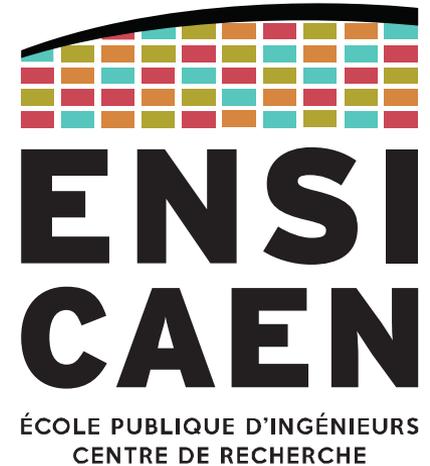
GENERAL PURPOSE PROCESSORS

Applications

Architecture

Motherboards

Superscalar processor



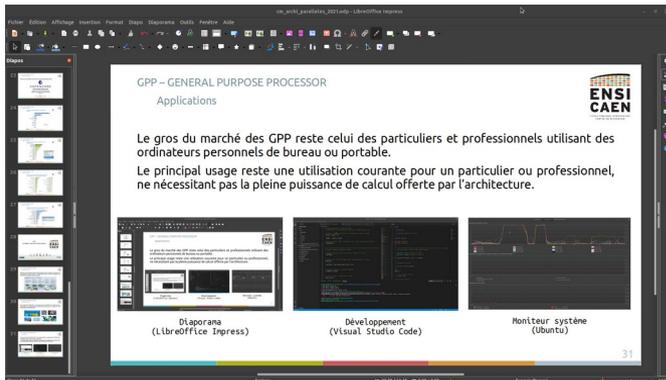
GPP (General Purpose Processors) have a complex CPU architecture that gives them a **great adaptability** especially for executing non-optimised programs.

Most of the time, those programs contain sequential code with a lot of tests and function calls, which are difficult to accelerate.

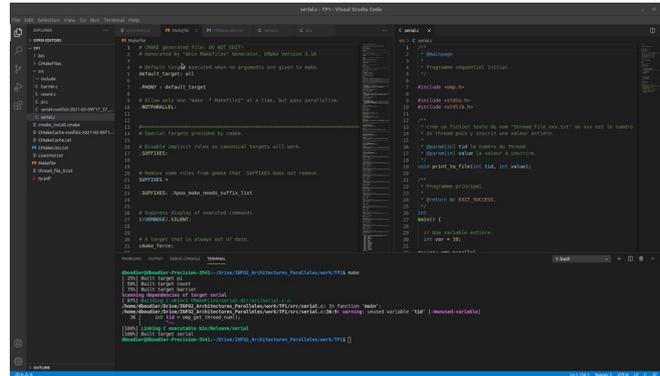
```
444     prev = NULL;
445     for (mpnt = oldmm->mmap; mpnt; mpnt = mpnt->vm_next) {
446         struct file *file;
447
448         if (mpnt->vm_flags & VM_DONTCOPY) {
449             vm_stat_account(mm, mpnt->vm_flags, -vma_pages(mpnt));
450             continue;
451         }
452         charge = 0;
453         if (mpnt->vm_flags & VM_ACCOUNT) {
454             unsigned long len = vma_pages(mpnt);
455
456             if (security_vm_enough_memory_mm(oldmm, len)) /* sic */
457                 goto fail_nomem;
458             charge = len;
459         }
460         tmp = kmem_cache_alloc(vm_area_cachep, GFP_KERNEL);
461         if (!tmp)
462             goto fail_nomem;
463         *tmp = *mpnt;
464         INIT_LIST_HEAD(&tmp->anon_vma_chain);
465         retval = vma_dup_policy(mpnt, tmp);
466         if (retval)
467             goto fail_nomem_policy;
```

Their target market are personal and professional computer and laptops.

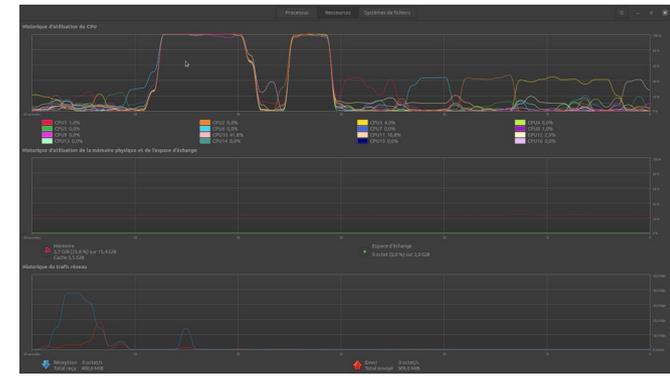
Thus their main usage is for general applications (i.e. not specific) for personal and professional uses. Most of the time that does not require all the computing power that is really available



Slideshow
(LibreOffice Impress)



Development
(Visual Studio Code)

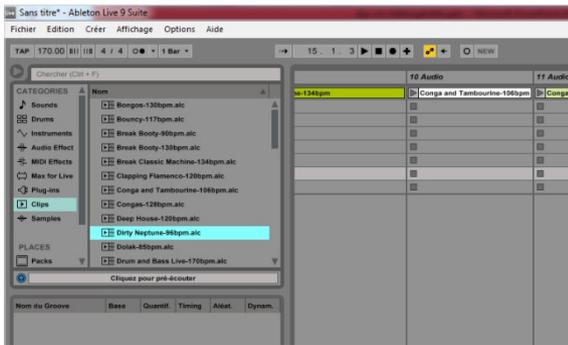


System monitor
(Ubuntu)

Applications

Of course some applications are likely to need full capability of the hardware, even though they are not the most common ones.

One can think of audio, image and video processing or software development as well-known examples.



Audio editing (Ableton)



Audio processing

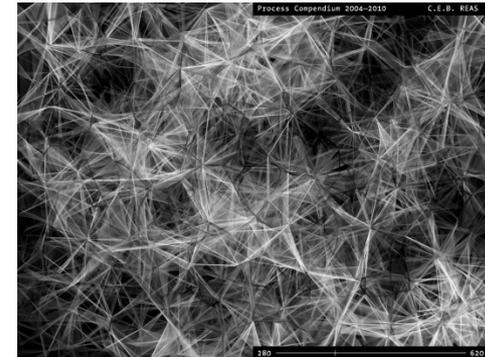


Image processing

Industrial applications are a historical part of GPP uses.

They are typically encountered on control tasks or specialised calculus functions. This market tends to use integrated solutions, such as AP (Application Processor), SoC (System on Chip), DSP (Digital Signal Processor), FPGA (Field Programmable Gate Array)

...



Radar GM400
(Thalès)

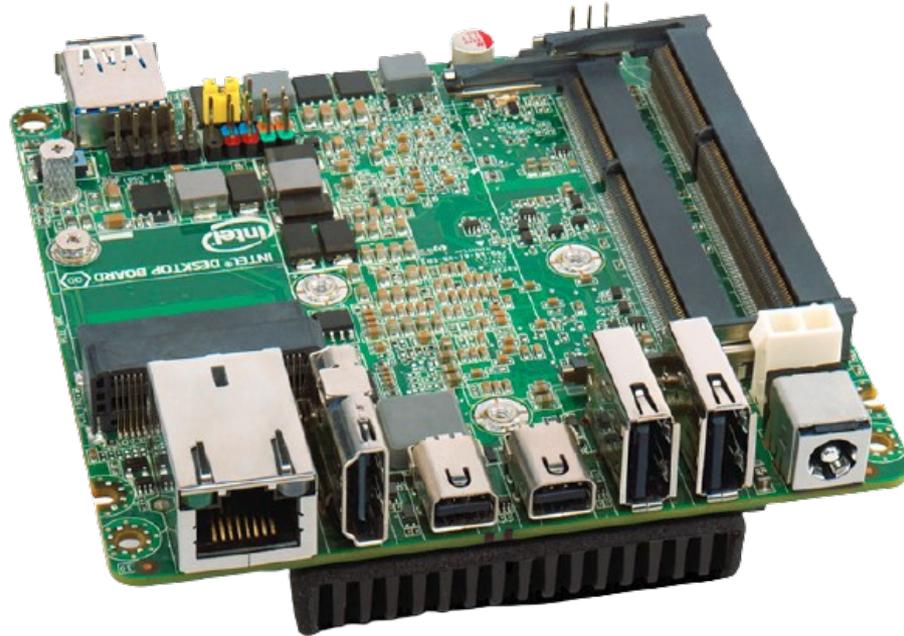


Rafale
(Dassault)

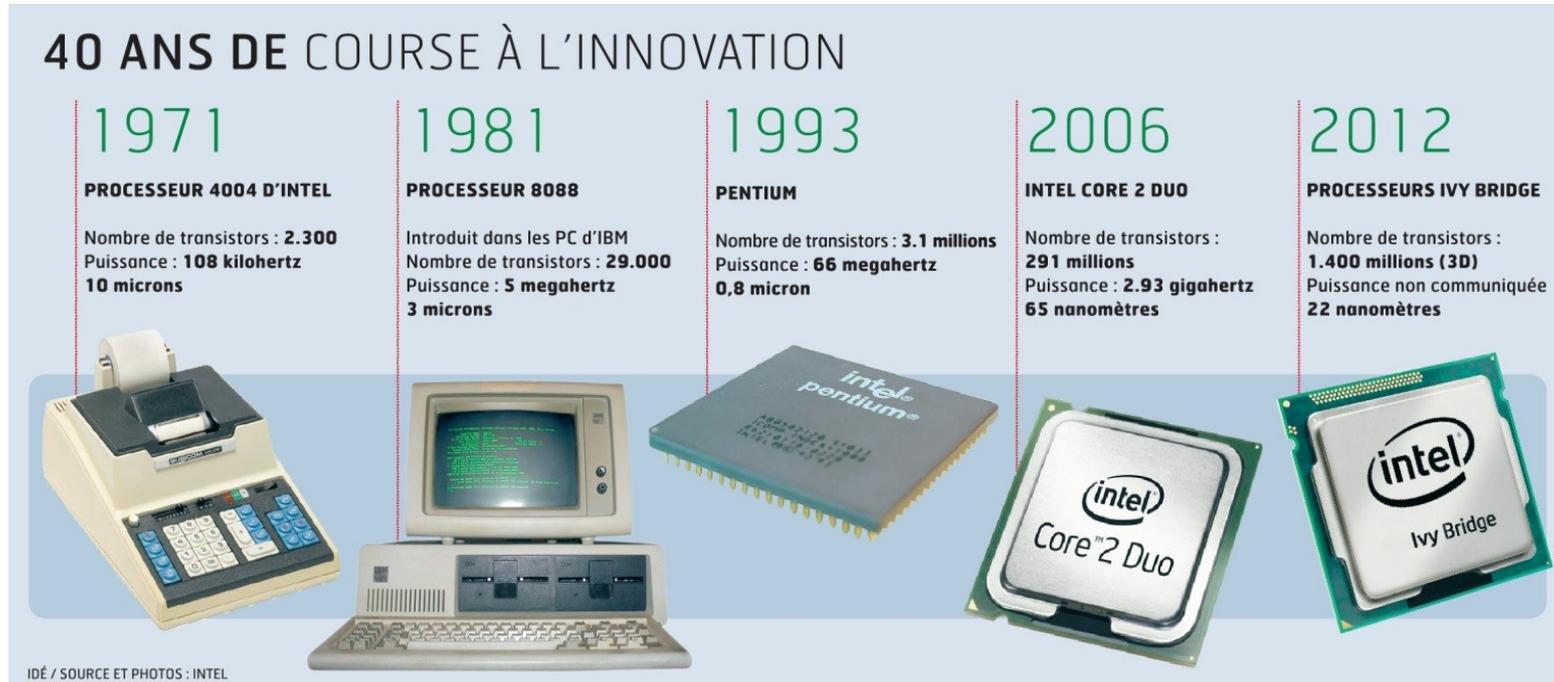


Automatic bollard
Box j200

Please note that GPPs can also be used in embedded systems applications.
For instance this is the NUC Core i5, an Intel motherboard.

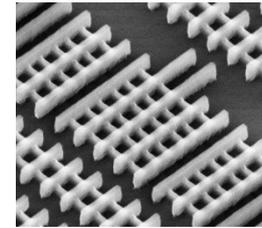
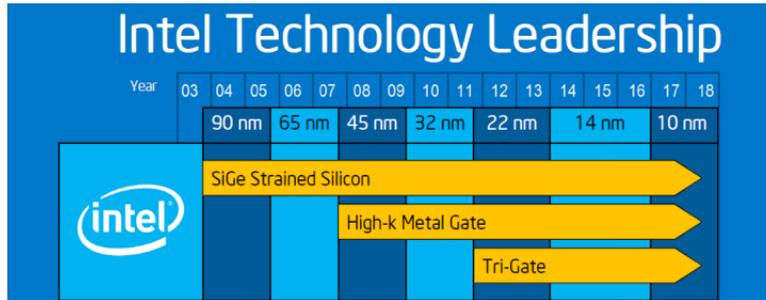


Let's have a look on major Intel architectures. Note that Intel is the historical and current leader of GPP market, but it is also the leader of semi-conductors market.



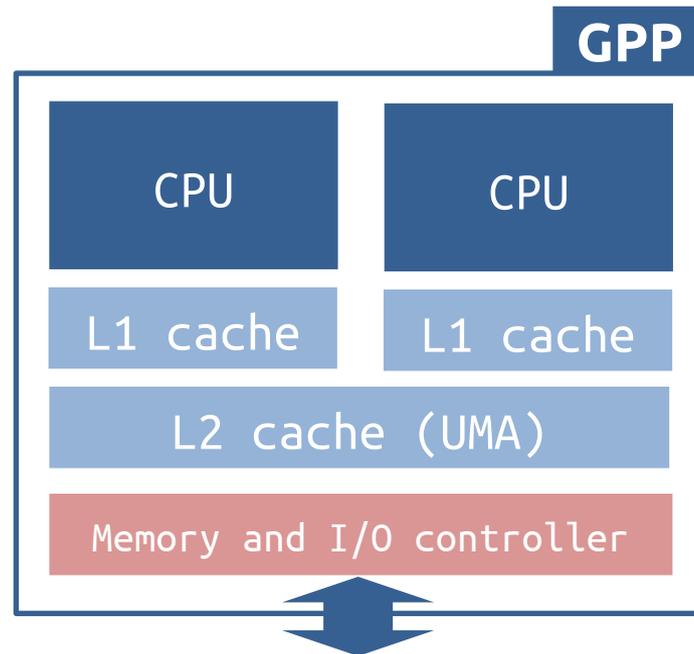
Today's leading GPP architectures are the Intel Core i3/i5/i7/i9 families.

However there are many other actors and manufacturers aiming for different markets.



A GPP consists of a are processing element, with no main memory.

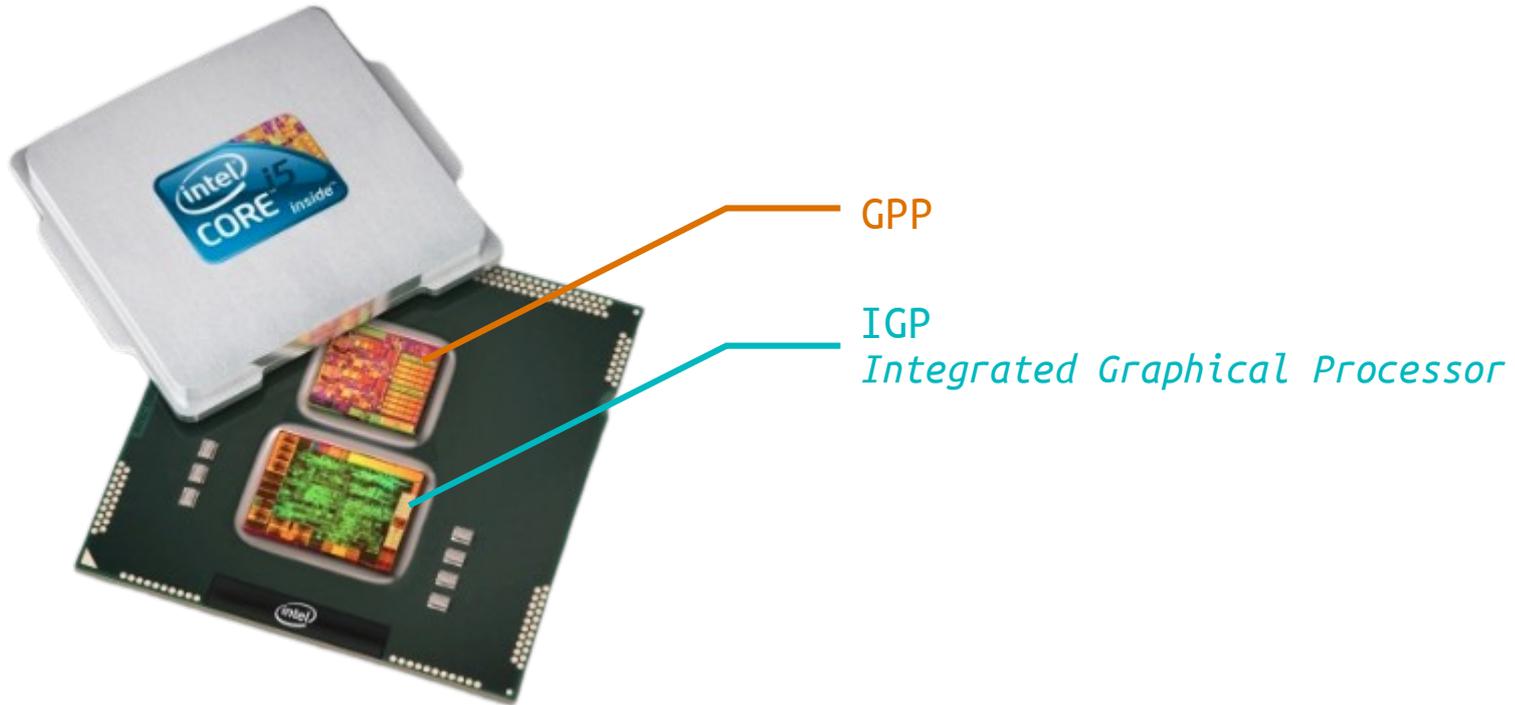
A GPP possesses one or several CPU (of same architecture) that are associated with their cache memories. They use an UMA (Uniform Memory Access) and and interface controller.



GPP – GENERAL PURPOSE PROCESSOR

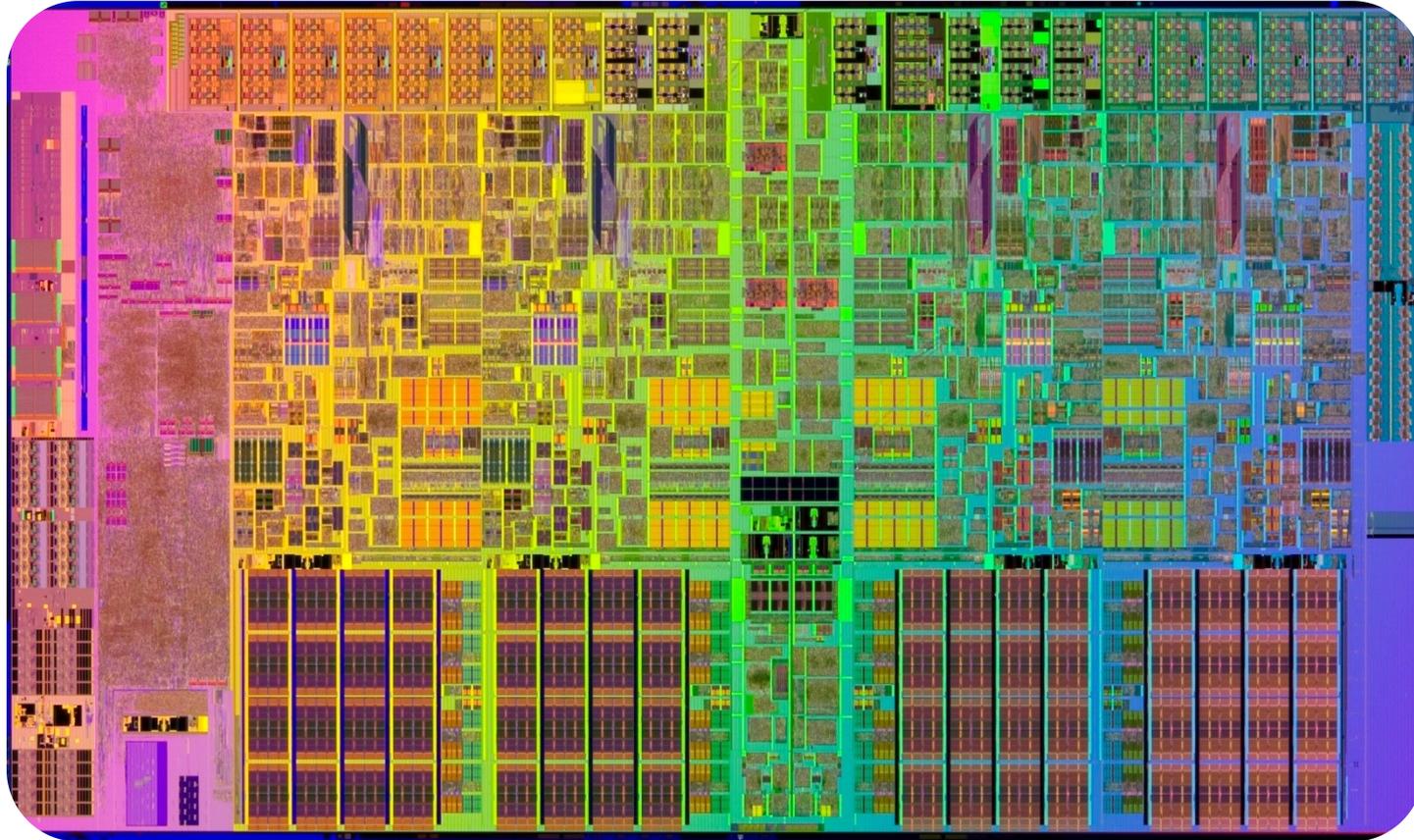
Example: Intel Core i5

Example of the Intel Core i5 family.



GPP – GENERAL PURPOSE PROCESSOR

Example: Intel Core i5

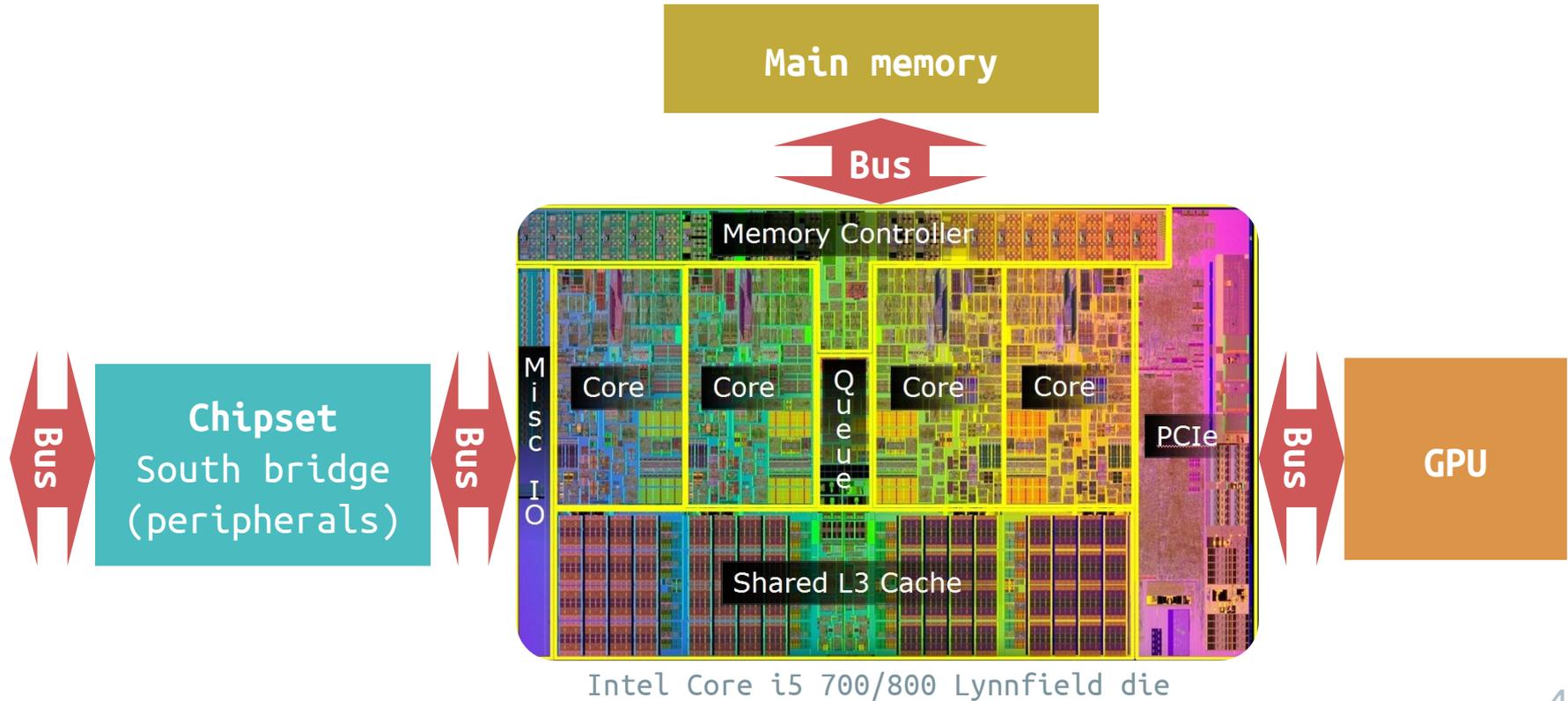


Intel Core i5 700/800 Lynnfield die

GPP – GENERAL PURPOSE PROCESSOR

Example: Intel Core i5

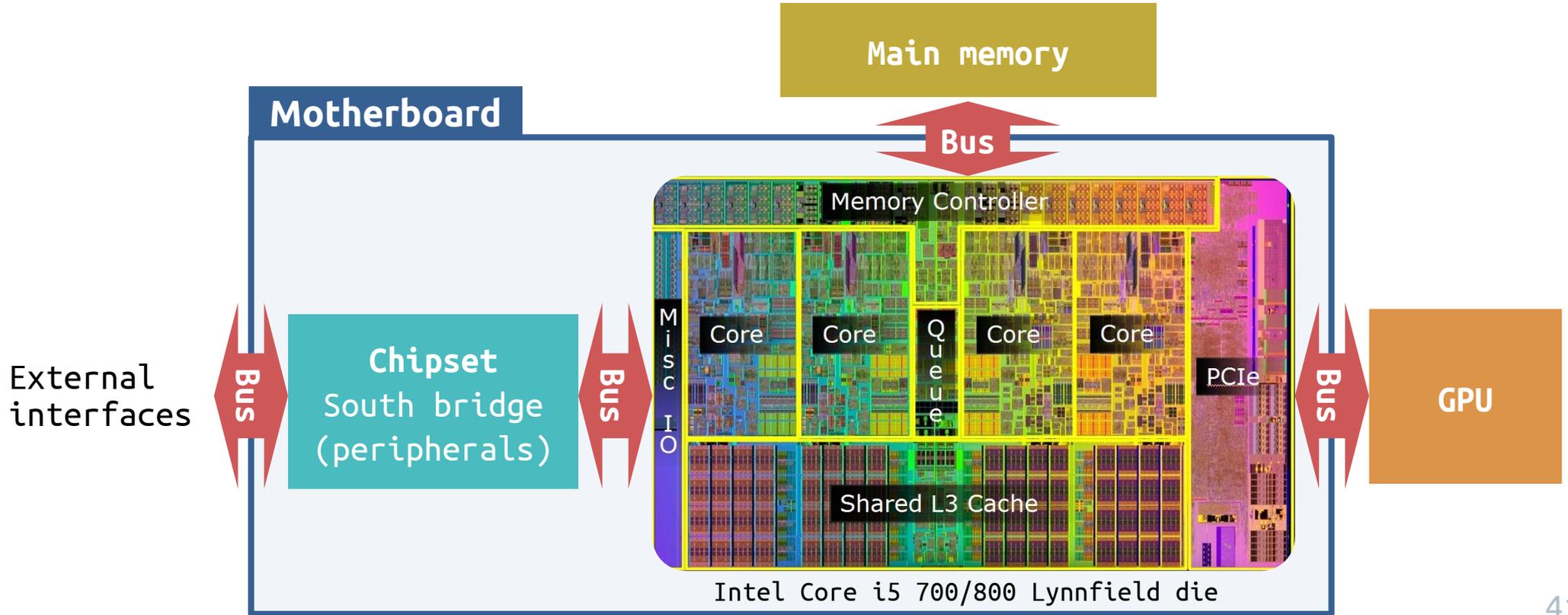
GPP integrated into a motherboard



GPP – GENERAL PURPOSE PROCESSOR

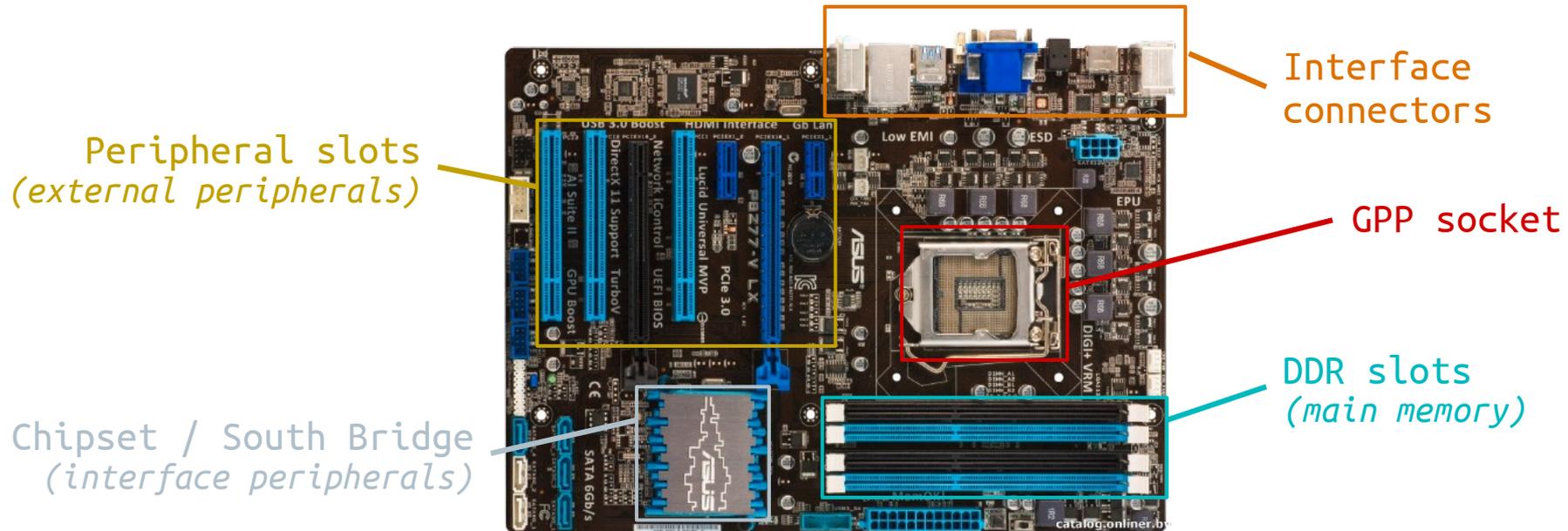
Example: Intel Core i5

GPP integrated into a motherboard



A GPP must be carried onto a motherboard, on which main memory (RAM) and external interface peripherals will be placed.

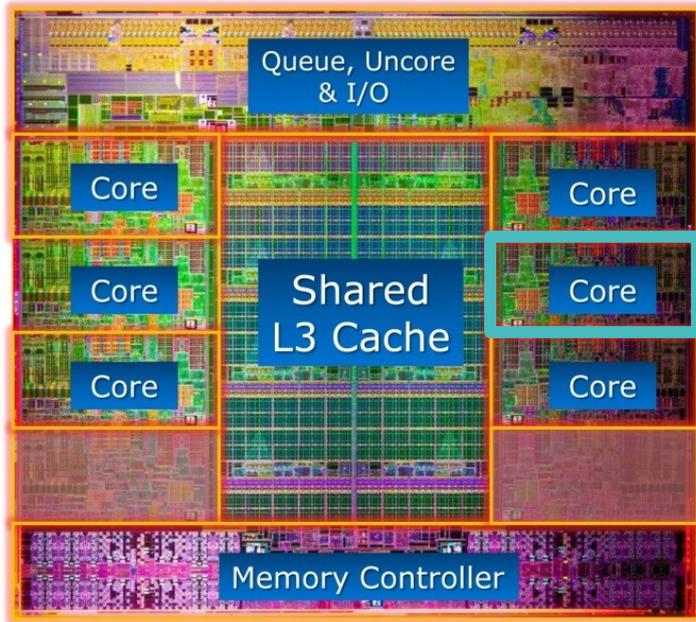
Example of a motherboard from ASUS, second leader of world market in 2016.



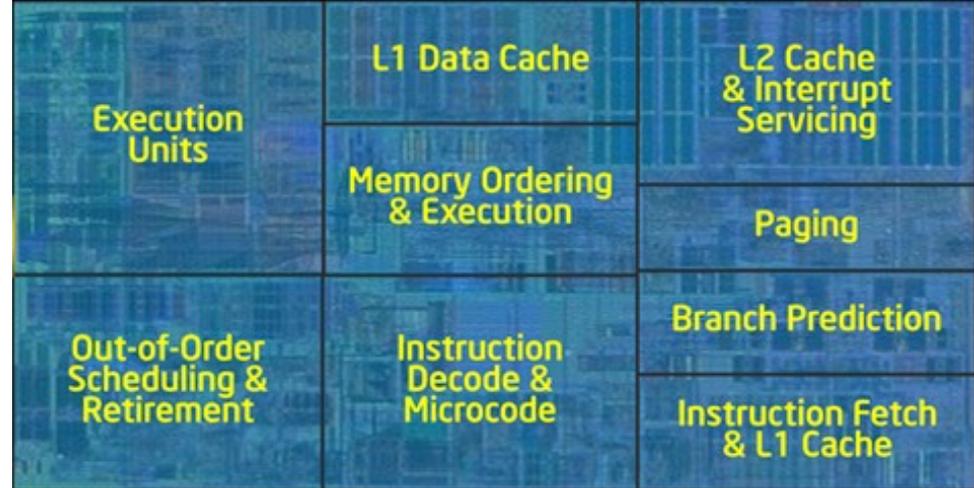
GPPs have **CPU said to be superscalar**. Processors with this type of CPU pipeline are generally characterised by the implementation of the following hardware accelerating mechanisms:

- **Out-Of-Order execution stage**: instructions are not executed in the programmed order. A hardware scheduler looks for dependencies on data, the intermediate results are stored in other registers and instructions are executed in another order (in comparison to the “programmed” order).
- **Branch-prediction stage**: use statistics and counters to estimate the success rate of a test statement (if, else, for, while, ...)
- **RISC-like execution stage**: even if the ISA (Instruction Set Architecture) is CISC.

Die of a Core i7 CPU (Intel Sandy Bridge generation).



Intel Core i7



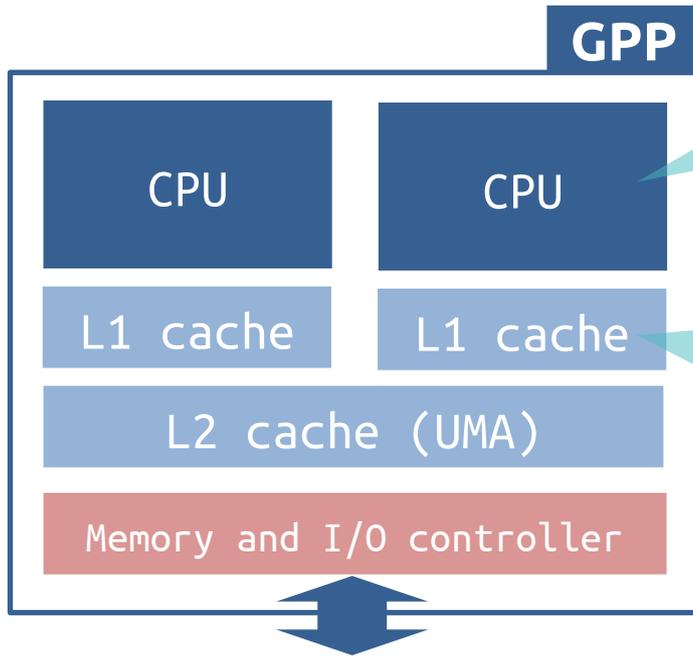
Sandy Bridge CPU/Core

However, GPP's great adaptability and hardware complexity leads to a lack of determinism and performance when it comes to the execution of specific algorithms.

For GPPs, the calculation power is simply not good when compared to the power consumption and the price.

GPPs are designed to support an high-end OS (Operating System, fr: *Système d'exploitation*) and to execute application code. As already mentioned, they are not specialised for signal, image, audio and video processing for instance.





Superscalar CPU

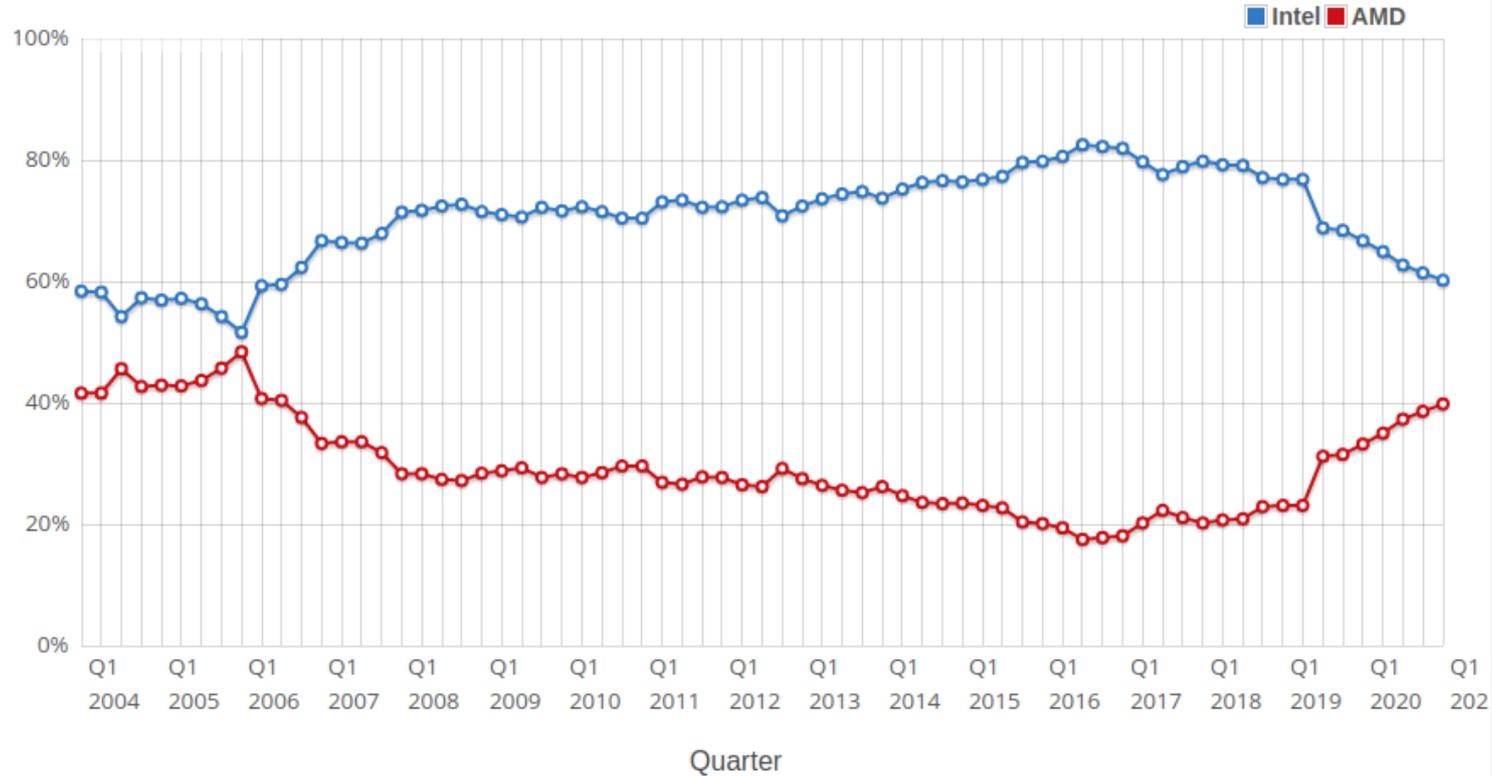
- Out-Of-Order execution
- Branch prediction
- Not determinist
- Bad (calculus power) / (Watt x Cost) ratio

Memory

- Uniform Memory Access (UMA)
- Cache memory:
 - Fast transfer technologies
 - Copy information from main memory (DATA or INST.)
 - Cache controllers for keeping data up to date
 - Not determinist

AMD vs Intel Market Share (All CPUs)

Last updated on the 11th of February 2021



AP

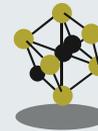
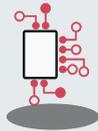
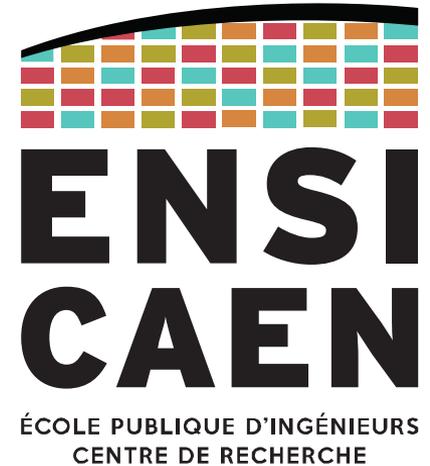
APPLICATION PROCESSOR

Applications

Architecture

Qualcomm

ARM



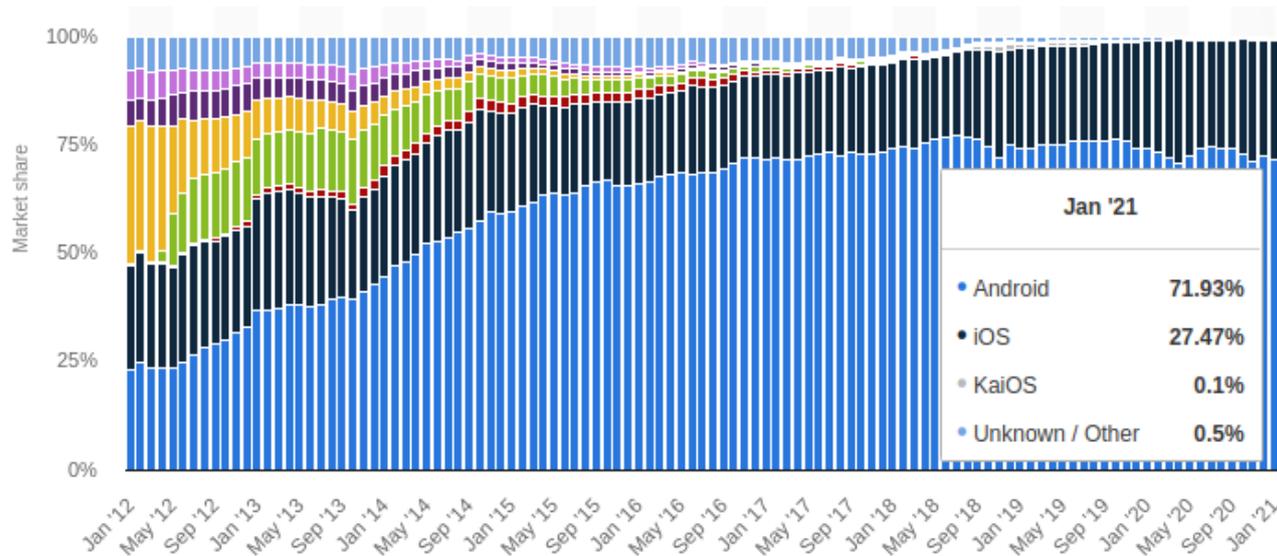
The AP (Application Processor) market is recent and has started with mobile phones and tablets.

APs embed many functionalities and hardware services, and even SoC (System on Chip).



Mobile phones is the main target market for APs.

This market has led to an overwhelming use of the Android operating system in 2016 (Android is a Linux-kernel based OS).



Source:
Statista 2021

● Android ● iOS ● KaiOS ● Windows Phone ● Series 40 (Nokia)* ● Symbian OS* ● Samsung
● BlackBerry OS ● Unknown / Other

However application processors are seen in many other embedded systems as well, whatever the final application: consumer, defence, transport, ...

In those cases they are usually embedded with an operating system and a graphical interface.



Freebox Revolution



Sony X94C 4K television



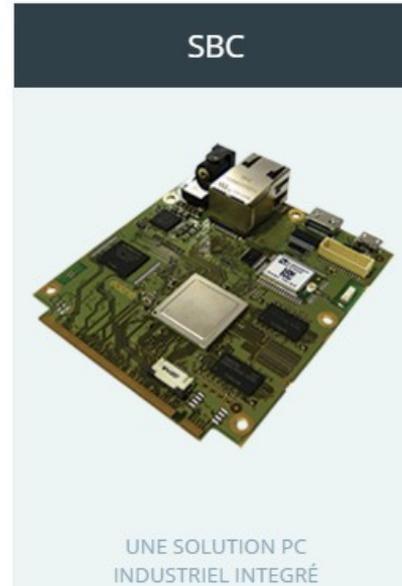
Cook tablet
(EOLANE, made in Caen)

In most cases, APs are used by high-level operating systems.

On those markets, GNU/Linux systems and customs versions reign supreme.



Example of EOLANE (French, #2 in Europe): industrial platform working with a Freescale iMX6 SoC/AP based on a GNU/Linux system.

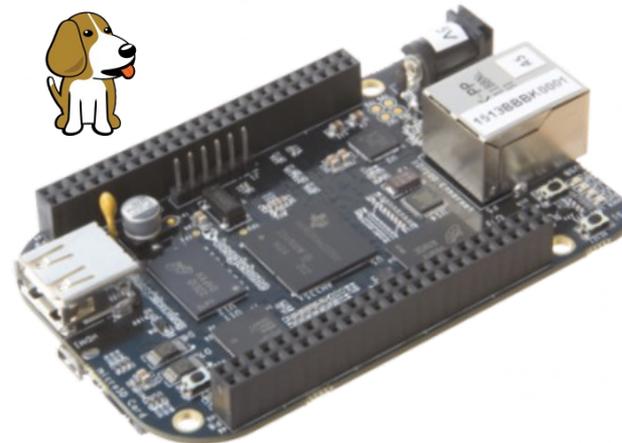


Here are the two major solutions of user-oriented AP-based boards:

Raspberry Pi (Broadcom BCMxxxx SoC) and **Beaglebone** (TI AM335x SoC) projects.

These solutions are also based on GNU/Linux operating systems.

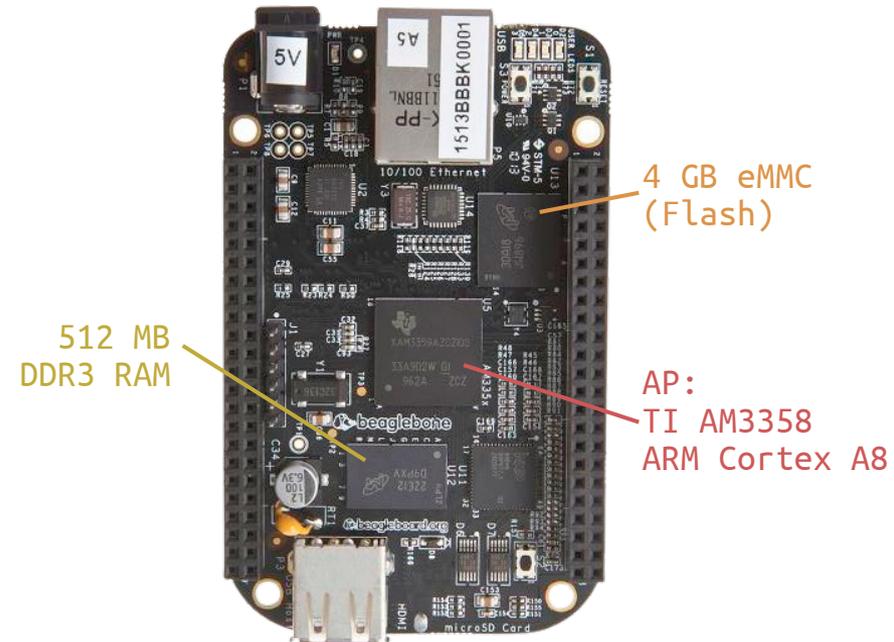
They are more likely to be used for prototyping stages or in a teaching environment, but cannot be industrialised. However hardened versions exist.



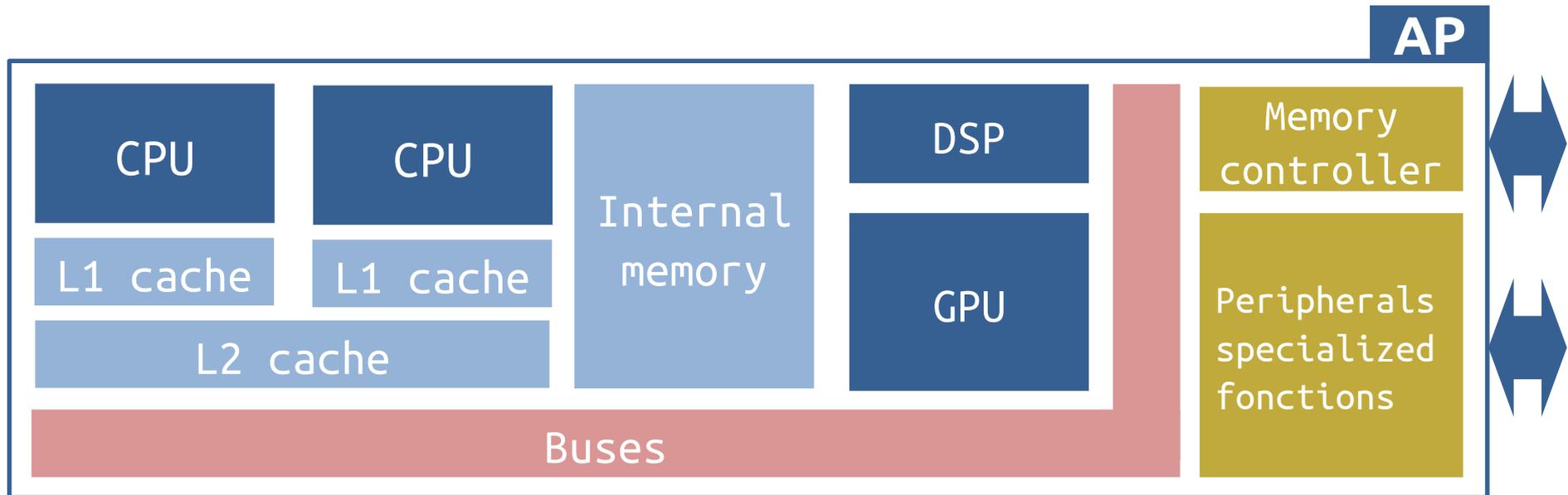
An **application processor** has one or several superscalar generalist CPUs. Their work is to execute the high-level operating systems (virtual or real) and application codes.

An **AP** may also have many calculus specialised functions (such as GPU, DSP, cryptography, ...), an evolved peripheral set and an internal memory. However the latter is not capable of containing the operating system but has a bootloader instead.

As a consequence a **DDR volatile main memory** and a **remanent mass storage** (MMC, eMMC, SDcard) must both be added as external components.

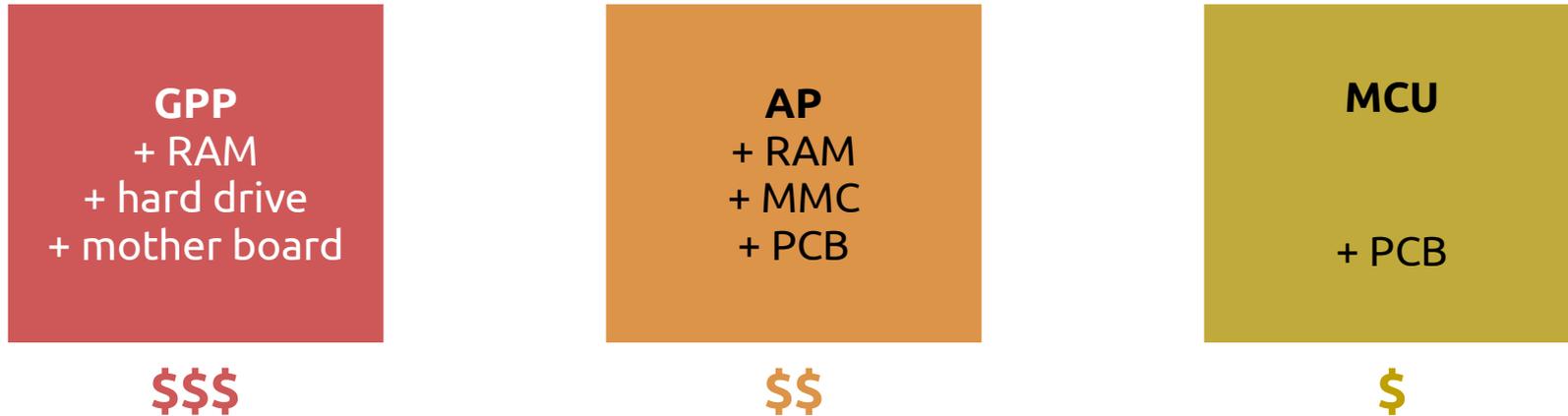


APs are fully operational systems in an integrated circuit (heterogeneous architecture).
Nonetheless main memory must be added as an external component.

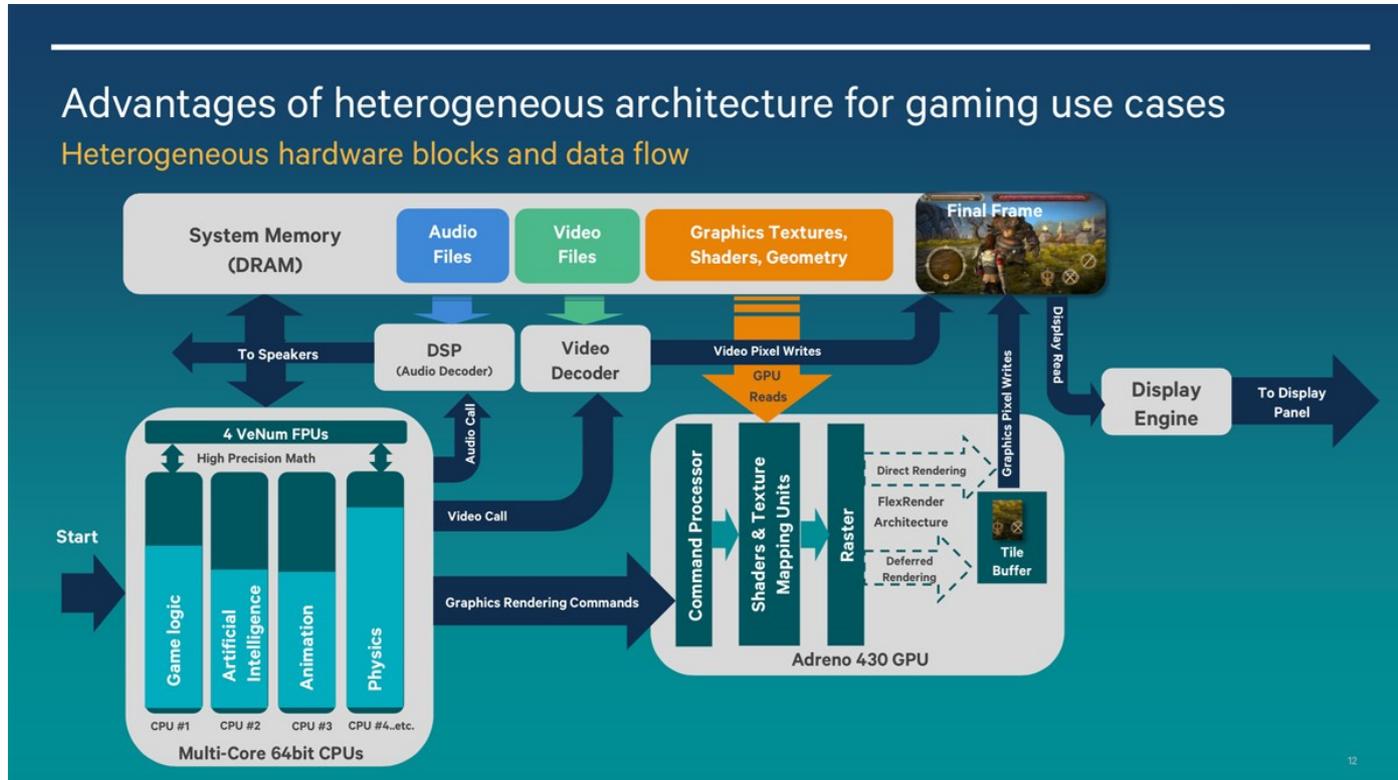


Contrary to MCUs, which contain all hardware services in a single chip, application processors require an important unitary cost and are therefore not the best solution for low-cost or large-quantities productions.

Yet if the application needs evolved interface and/or connectivities, MCUs are not suitable any more because of their low performances. APs then become the best solution.



Observe the point of a heterogeneous architecture for video games applications.



AP – APPLICATION PROCESSOR

Qualcomm Snapdragon solution

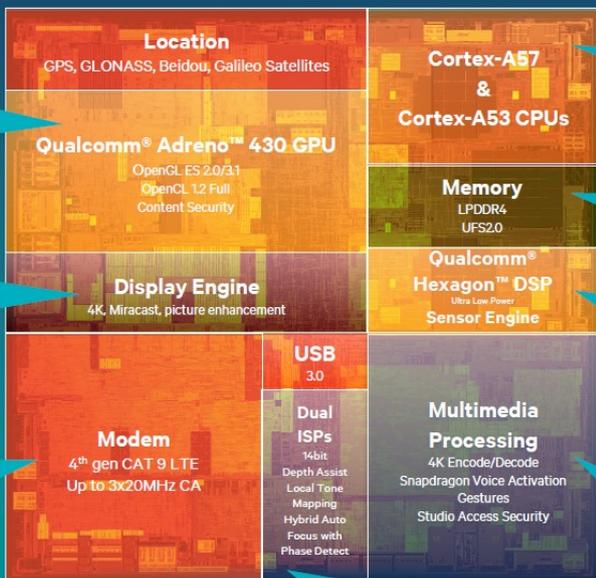
The market leader is Qualcomm.

This is due to its Snapdragon family dedicated to mobile phones market.



Internal architecture and hardware functionalities of the Qualcomm Snapdragon 810.

Introducing the Snapdragon 810 Processor



Location GPS, GLONASS, Beidou, Galileo Satellites	Cortex-A57 & Cortex-A53 CPUs
Qualcomm® Adreno™ 430 GPU OpenGL ES 2.0/3.1 OpenCL 1.2 Full Content Security	Memory LPDDR4 UFS2.0
Display Engine 4K, Miracast, picture enhancement	Qualcomm® Hexagon™ DSP Sensor Engine Ultra Low Power
Modem 4 th gen CAT 9 LTE Up to 3x20MHz CA	USB 3.0 Dual ISP 14bit Depth Assist Local Tone Mapping Hybrid Auto Focus with Phase Detect
	Multimedia Processing 4K Encode/Decode Snapdragon Voice Activation Gestures Studio Access Security

Not drawn to scale.

Qualcomm Adreno and Qualcomm Hexagon are pro

Qualcomm Technologies' FIRST 14-bit Dual ISP for highest quality, depth enabled photography. Up to 21MP for main camera with depth assist, phase detect, for sharper dual camera user experiences

Advanced Graphics & Compute with the Adreno 430 – the best GPU Qualcomm Technologies' has ever made

4K primary & external display support with ecoPix and TruPalette and 3:1 pixel compression

Mobile industry's FIRST announced multi-channel 4G LTE SoC supporting Category 9 Carrier Aggregation

FIRST Announced ARM®v8-A/64-bit using Cortex®-A57+ Cortex®-A53

Mobile industry's FIRST announced dual channel 1600 MHz LPDDR4 memory

Qualcomm Technologies' FIRST UFS 2.0 Support

Greatly improved power management for DSP/Sensor Engine, Low Power Snapdragon Voice Activation (SVA), 12-channel surround sound decode

Qualcomm Technologies' FIRST hardware implementation of 4K HEVC/H.265 video encode. HEVC designed to deliver up to 50% better video compression

Les deux leaders du marché hors terminaux mobiles sont Texas Instruments et Freescale, deux fondateurs offrant de larges communautés d'utilisateurs.

Observons la famille i.MX6 de Freescale :

i.MX 6 Series At a Glance
Scalable series of six ARM Cortex A9-based SoC families

Red indicates change from column to the left

i.MX 6SoloLite	i.MX 6SoloX	i.MX 6Solo	i.MX 6DualLite	i.MX 6Dual	i.MX 6Quad
<ul style="list-style-type: none"> Single ARM® Cortex™-A9 at 1GHz 256KB L2 cache, Neon, VFPv16, Trustzone 2D graphics 32-bit DDR3 and LPDDR2 at 400MHz 10/100 Ethernet EPD controller 	<ul style="list-style-type: none"> Single ARM Cortex-A9 up to 1GHz Single Cortex-M4 at 166Mhz 256KB L2 cache, Neon, VFP, Trustzone 3D and 2D Graphics 32-bit DDR3 and LPDDR2 at 400MHz Dual Gigabit Ethernet PCIe (x1 lane) 	<ul style="list-style-type: none"> Single ARM Cortex-A9 up to 1GHz 512KB L2 cache, Neon, VFPv16, Trustzone 3D graphics with 1 shader 2D graphics 32-bit DDR3 and LPDDR2 at 400MHz 1080p30 video Gigabit Ethernet PCIe (x1 lane) EPD controller 	<ul style="list-style-type: none"> Dual ARM Cortex-A9 up to 1GHz 512KB L2 cache, Neon, VFPv16, Trustzone 3D graphics with 1 shader 2D graphics 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 400MHz 1080p30 video Gigabit Ethernet PCIe (x1 lane) EPD controller 	<ul style="list-style-type: none"> Dual ARM Cortex-A9 up to 1/1.2GHz 1 MB L2 cache, Neon, VFPv16, Trustzone 3D graphics with 4 shaders Two 2D GFX engines 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 533MHz 1080p60 video PCIe (x1 lane) Gigabit Ethernet SATA-II 	<ul style="list-style-type: none"> Quad ARM Cortex-A9 up to 1/1.2GHz 1 MB L2 cache, Neon, VFPv16, Trustzone 3D graphics with 4 shaders Two 2D GFX engines 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 533MHz 1080p60 video PCIe (x1 lane) Gigabit Ethernet SATA-II

Pin-to-pin and Power Compatible

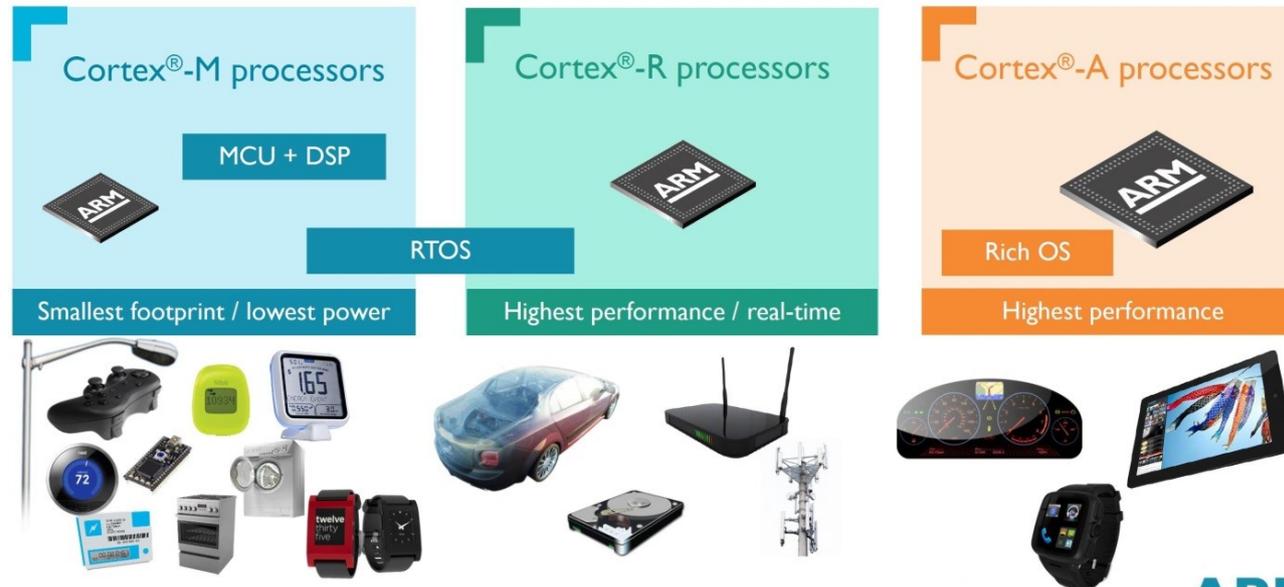
Software Compatible

- ARM Cortex-A9 based solutions ranging up to 1.2GHz
- HD 1080p encode and decode (except 6SoloLite/6SoloX)
- 3D video playback in high definition (except 6SoloLite/6SoloX)
- Integrated IO's may include HDMI v1.4, MIPI and LVDS, display ports, MIPI camera, Gigabit Ethernet, multiple USB 2.0 and PCI-Express
- SW support: Google Android™, Linux®, QNX, Windows® Embedded CE

freescale™ External Use | 24

Outside of the mobile phones market, the ARM Cortex-A is the leading architecture in embedded markets. The 'A' stands for "Application".

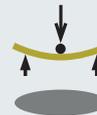
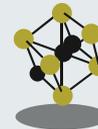
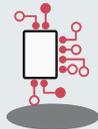
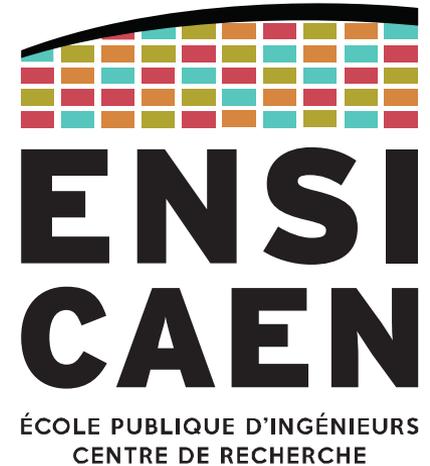
ARM® Cortex® Processors across the Embedded Market



GPU

GRAPHICS PROCESSING UNIT

Applications
Architecture
Nvidia products
Markets

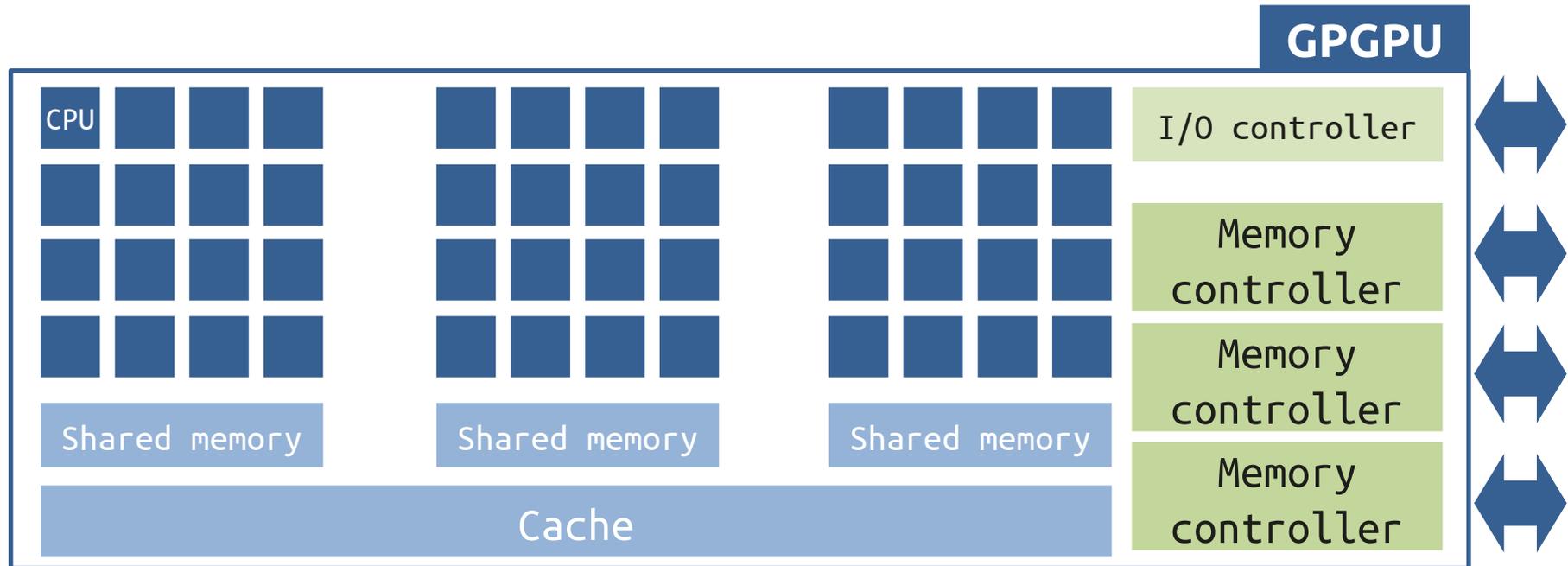


GPUs (*Graphics Processing Unit*) are specialised **co-processors** dedicated for high intensity calculus and processing.

The term of GPGPU (General Purpose GPU) appeared in the last few years. It relates to massive computing in very sense. Applications are diverse: finance, research, science, medical imagery, video games, ...



GPU possess a shared NUMA (Non Uniform Memory Access), allowing a cloning of data to be processed and a execution parallelism. They integrated a massively parallel architecture.



Let's take a look at the Tesla P100 board characteristics. It has been produced by Nvidia in 2016 and it is dedicated to the then most advanced data centres.

The GPU is a Nvidia GP100.



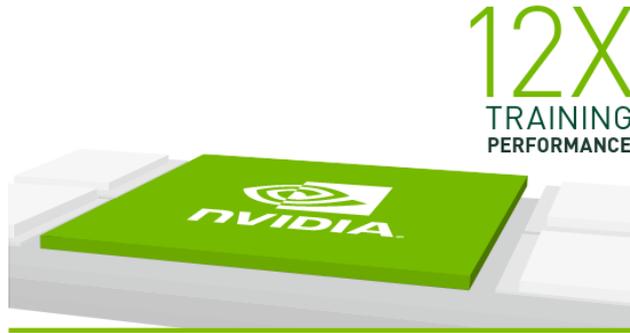
SPECIFICATIONS

GPU Architecture	NVIDIA Pascal
NVIDIA CUDA® Cores	3584
Double-Precision Performance	5.3 TeraFLOPS
Single-Precision Performance	10.6 TeraFLOPS
Half-Precision Performance	21.2 TeraFLOPS
GPU Memory	16 GB CoWoS HBM2
Memory Bandwidth	732 GB/s
Interconnect	NVIDIA NVLink
Max Power Consumption	300 W
ECC	Native support with no capacity or performance overhead
Thermal Solution	Passive
Form Factor	SXM2
Compute APIs	NVIDIA CUDA, DirectCompute, OpenCL™, OpenACC

TeraFLOPS measurements with NVIDIA GPU Boost™ technology

GPU – GRAPHICS PROCESSING UNITS

Nvidia products: Pascal architecture



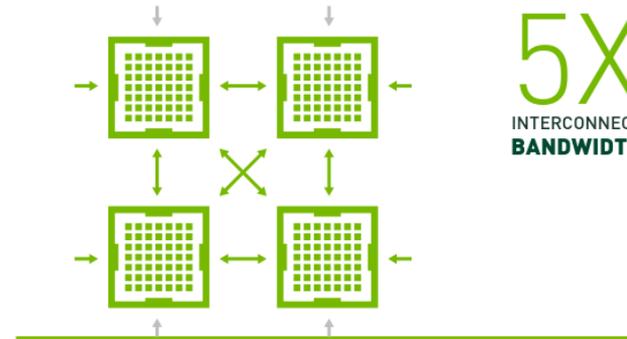
LEAP IN NEURAL NETWORK TRAINING PERFORMANCE
WITH NEW **NVIDIA PASCAL ARCHITECTURE**



FABRICATED WITH **16 NANOMETER FINFET**
FOR UNPRECEDENTED ENERGY EFFICIENCY



WITH **CoWoS® WITH HBM2** COMPARED TO
NVIDIA MAXWELL™ ARCHITECTURE FOR
BIG DATA WORKLOADS



WITH **NVIDIA NVLINK™** FOR MAXIMUM
APPLICATION SCALABILITY

GPU – GRAPHICS PROCESSING UNITS

Nvidia products: GP100 GPU architecture



The Nvidia GP100 GPU in a nutshell

- 6 Graphics Processing Clusters
- 30 Texture Processing Clusters (5 / GPC)
- 60 Streaming Multiprocessors (2 / TPC)
- 3840 single precision cores (64 / SM)
- 1920 double precision units (32 / SM)
- 240 texture units (4 / SM)
- 8 memory controllers
 - 8 x 512 KB = 4096 KB L2 cache
 - 4 pairs that control HBM2 DRAM

Note : the Tesla P100 board uses only 56 SMs out of the 60 available in the GP100 GPU.

Tesla Products	Tesla K40	Tesla M40	Tesla P100
GPU	GK110 (Kepler)	GM200 (Maxwell)	GP100 (Pascal)
SMs	15	24	56
TPCs	15	24	28
FP32 CUDA Cores / SM	192	128	64
FP32 CUDA Cores / GPU	2880	3072	3584
FP64 CUDA Cores / SM	64	4	32
FP64 CUDA Cores / GPU	960	96	1792
Base Clock	745 MHz	948 MHz	1328 MHz
GPU Boost Clock	810/875 MHz	1114 MHz	1480 MHz
Peak FP32 GFLOPs ¹	5040	6840	10600
Peak FP64 GFLOPs ¹	1680	210	5300
Texture Units	240	192	224
Memory Interface	384-bit GDDR5	384-bit GDDR5	4096-bit HBM2
Memory Size	Up to 12 GB	Up to 24 GB	16 GB
L2 Cache Size	1536 KB	3072 KB	4096 KB
Register File Size / SM	256 KB	256 KB	256 KB
Register File Size / GPU	3840 KB	6144 KB	14336 KB
TDP	235 Watts	250 Watts	300 Watts
Transistors	7.1 billion	8 billion	15.3 billion
GPU Die Size	551 mm ²	601 mm ²	610 mm ²
Manufacturing Process	28-nm	28-nm	16-nm FinFET

¹ The GFLOPS in this chart are based on GPU Boost Clocks.

GPUs integrate a large number of classical pipeline CPUs but with vectorial SIMD execution units.

EU = Execution Unit

SIMD = Single Instruction Multiple Data

GPC = Graphics Processing Cluster

TCP = Texture Processing Cluster

SM = Streaming Multiprocessor
(multithreaded processor)

Warp = thread of SIMD instructions

DP = Double Precision

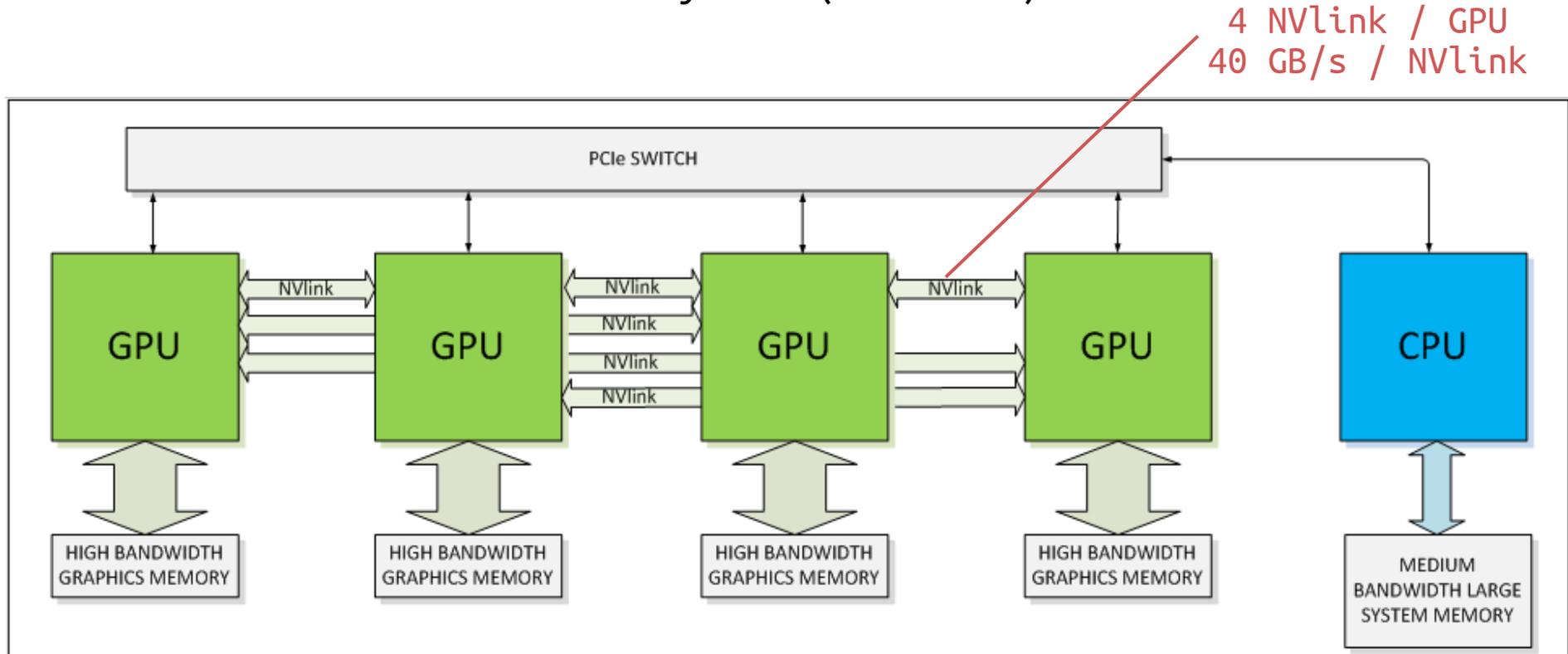
LD/ST = Load/Store

SFU = Special Function Unit

Tex = Texture



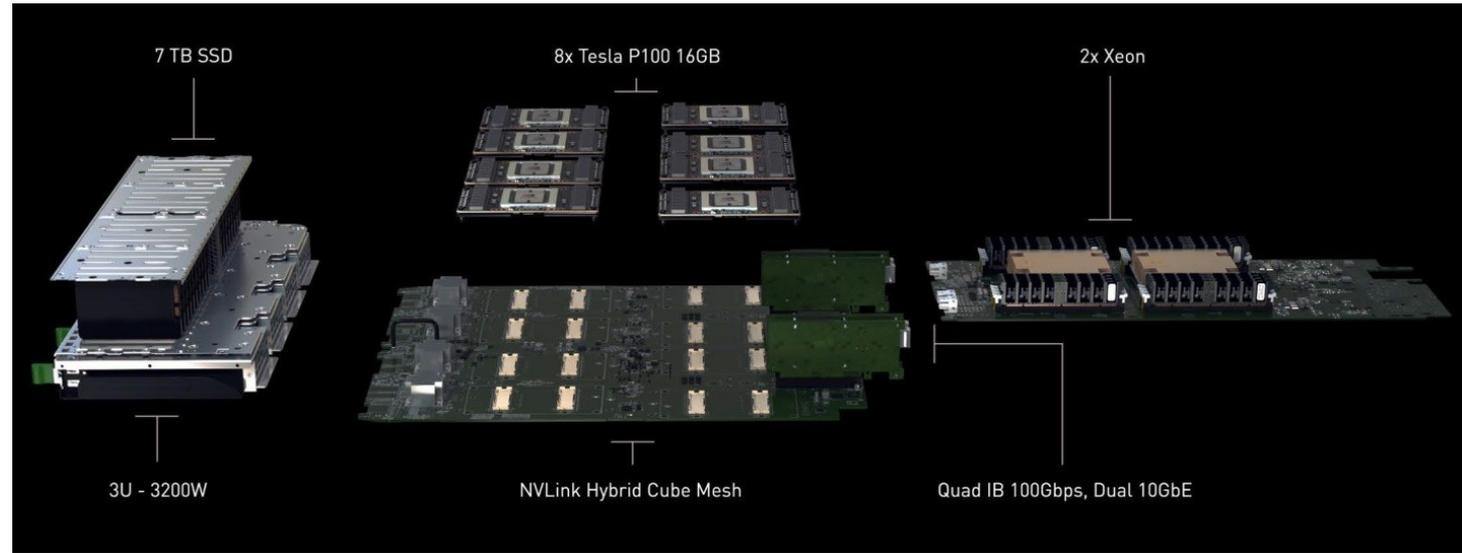
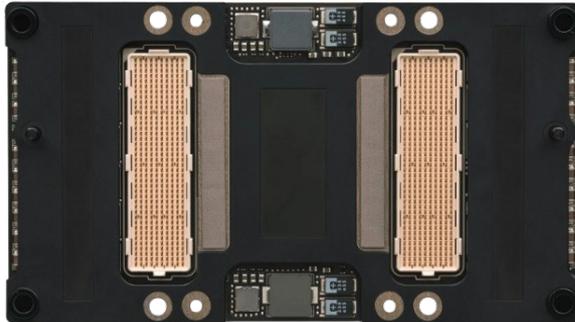
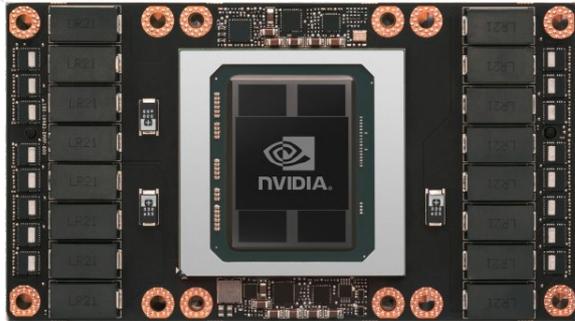
Communication and interconnection systems (Tesla P100)



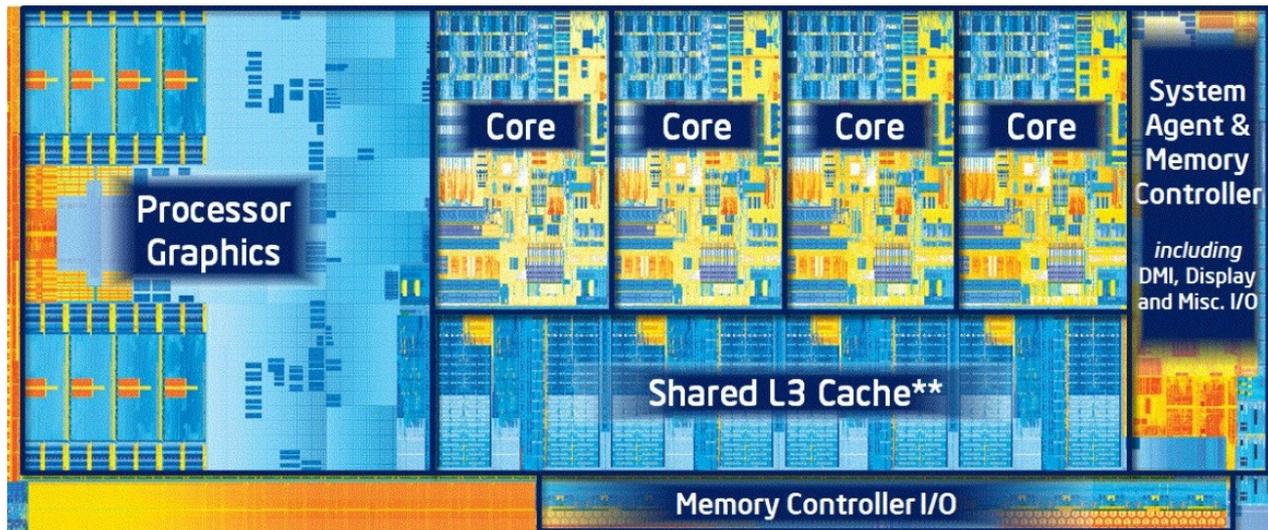
GPU – GRAPHICS PROCESSING UNITS

Nvidia products: application example

Example of an application using the Nvidia Tesla P100 board.



The undisputed leader of the GPU/IGP market is Intel, thanks to their graphics co-processors IGP (Integrated Graphics Units) embedded in a wide range of their GPPs (more than 70% of market shares in 2016).



Nonetheless the leader of high-performance external solutions in the American company Nvidia.



Tesla K20C



Tesla P100



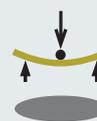
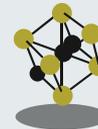
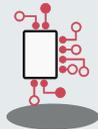
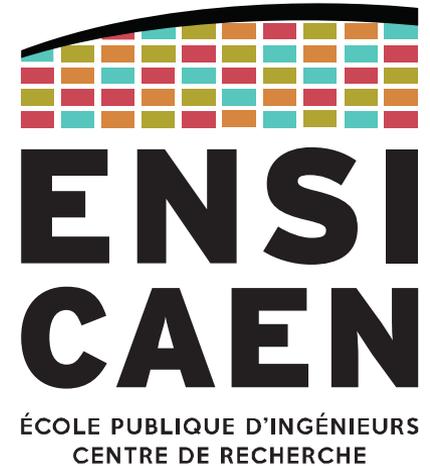
DSP

DIGITAL SIGNAL PROCESSOR

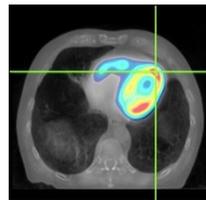
Applications

Architecture

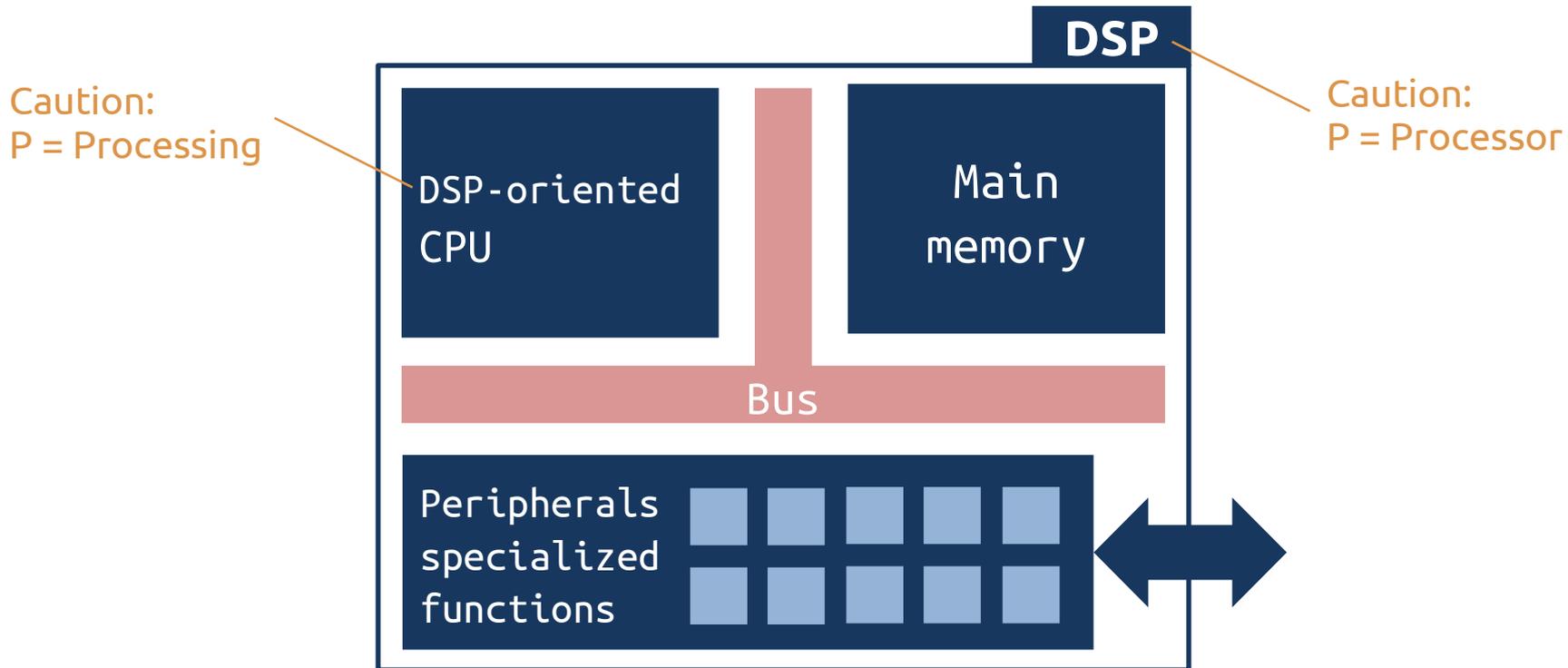
Texas Instruments



DSPs (Digital Signal Processors) are dedicated to applications with Digital Signal Processing (fr: Traitement numérique du signal).



DSPs are very close to MCUs: they are autonomous systems.
However their CPU is specialised for signal processing and calculus.

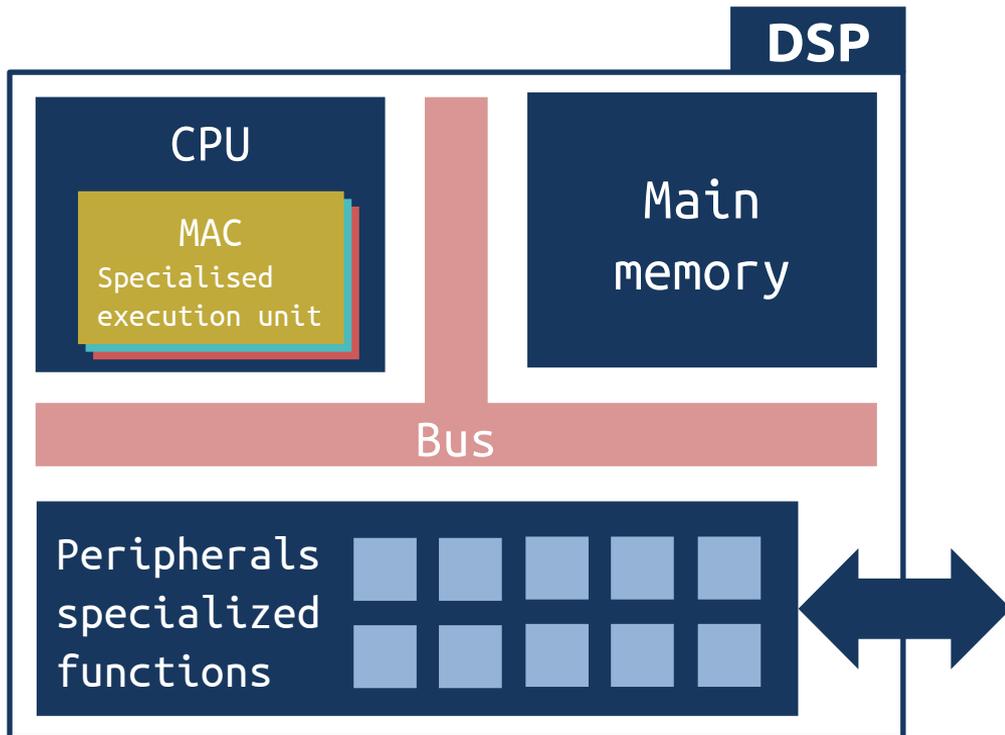


DSP's CPUs possess execution units dedicated for MAC (Multiply Accumulate) or SOP (Som Of Products) operations. These are elementary operations met in almost every signal processing algorithm.

Expansion of the Danielson-Lanczos Lemma to 8 terms:

$$\begin{aligned}
 F(n) = & \sum_{k=0}^{N/8-1} x(8k)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + W_N^{\frac{n}{4}} \sum_{k=0}^{N/8-1} x(8k+4)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + \\
 & W_N^{\frac{n}{2}} \sum_{k=0}^{N/8-1} x(8k+2)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + W_N^{\frac{n}{2}} W_N^{\frac{n}{4}} \sum_{k=0}^{N/8-1} x(8k+6)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + \\
 & W_N^{\frac{n}{4}} \sum_{k=0}^{N/8-1} x(8k+1)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + W_N^{\frac{n}{4}} W_N^{\frac{n}{4}} \sum_{k=0}^{N/8-1} x(8k+5)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + \\
 & W_N^{\frac{n}{2}} W_N^{\frac{n}{2}} \sum_{k=0}^{N/8-1} x(8k+3)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}} + W_N^{\frac{n}{2}} W_N^{\frac{n}{2}} W_N^{\frac{n}{4}} \sum_{k=0}^{N/8-1} x(8k+7)e^{\frac{-j2\pi kn}{\left(\frac{N}{8}\right)}}
 \end{aligned}$$

CPU with MAC/SOP dedicated execution units. The ISA (Instruction Set Architecture) contains specific instructions for working with these EUs.

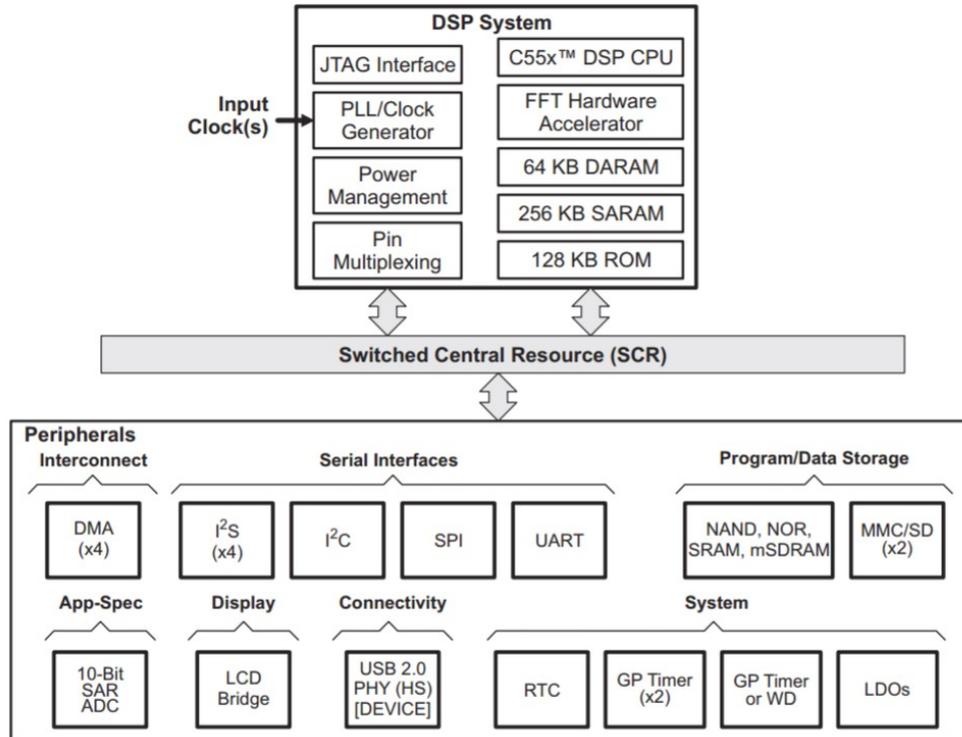


MAC = SOP

MAC : Multiply-Accumulate
SOP : Som of Products

ISA : Instruction Set Architecture
EU : Execution Unit

This is the Texas Instruments C5500 DSP, one of the leading DSP solutions.

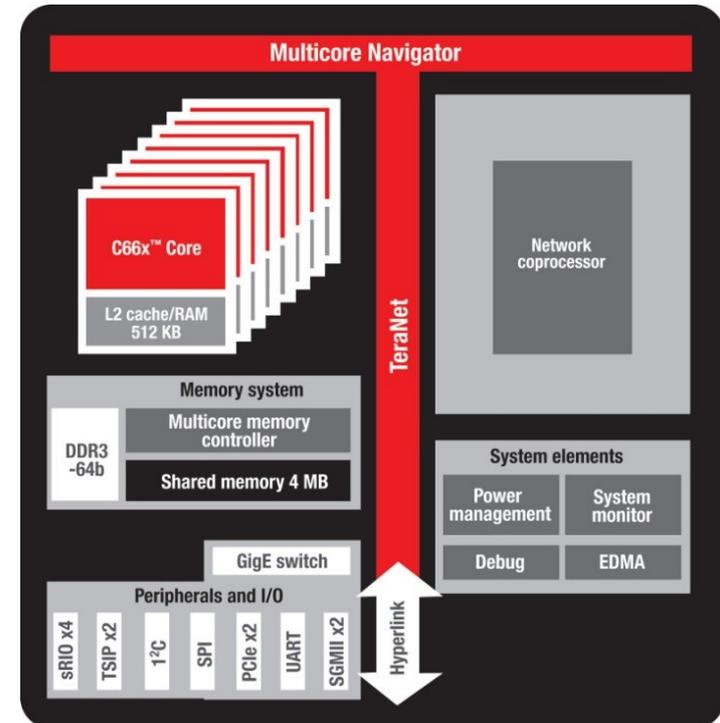
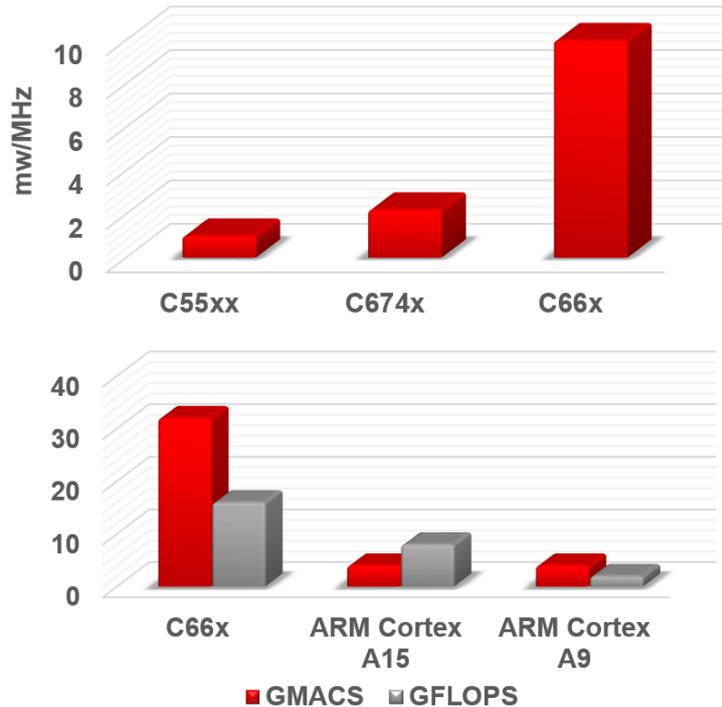


Here is an extract of the C5500 datasheet, with a summary of its characteristics.

1.1 Features

- CORE:
 - High-Performance, Low-Power, TMS320C55x Fixed-Point Digital Signal Processor
 - 20-, 10-ns Instruction Cycle Time
 - 50-, 100-MHz Clock Rate
 - One or Two Instructions Executed per Cycle
 - Dual Multiply-and-Accumulate Units (Up to 200 Million Multiply-Accumulates per Second [MMACS])
 - Two Arithmetic and Logic Units (ALUs)
 - Three Internal Data and Operand Read Buses and Two Internal Data and Operand Write Buses
 - Software-Compatible with C55x Devices
 - Industrial Temperature Devices Available
 - 320KB of Zero-Wait State On-Chip RAM, Composed of:
 - 64KB of Dual-Access RAM (DARAM), 8 Blocks of 4K x 16-Bit
 - 256KB of Single-Access RAM (SARAM), 32 Blocks of 4K x 16-Bit
 - 128KB of Zero Wait-State On-Chip ROM (4 Blocks of 16K x 16-Bit)
 - Tightly Coupled FFT Hardware Accelerator

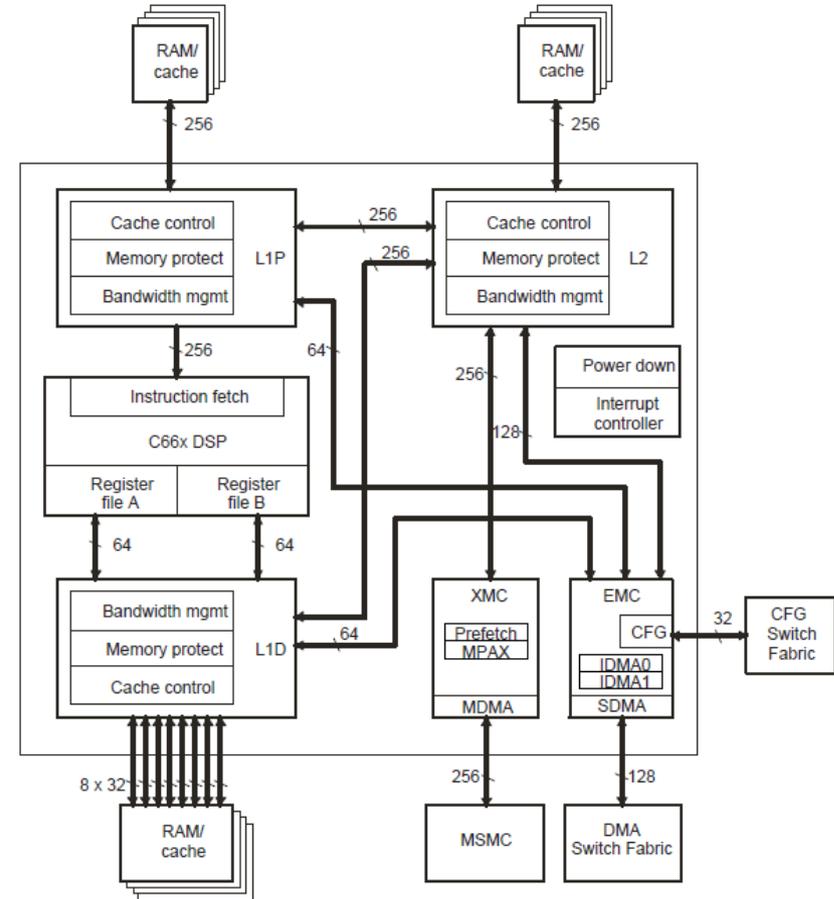
Let's switch to the Keystone C6600. This Texas Instruments DSP is one of the highest performances in the current market.



Texas Instruments C6600 CorePac.

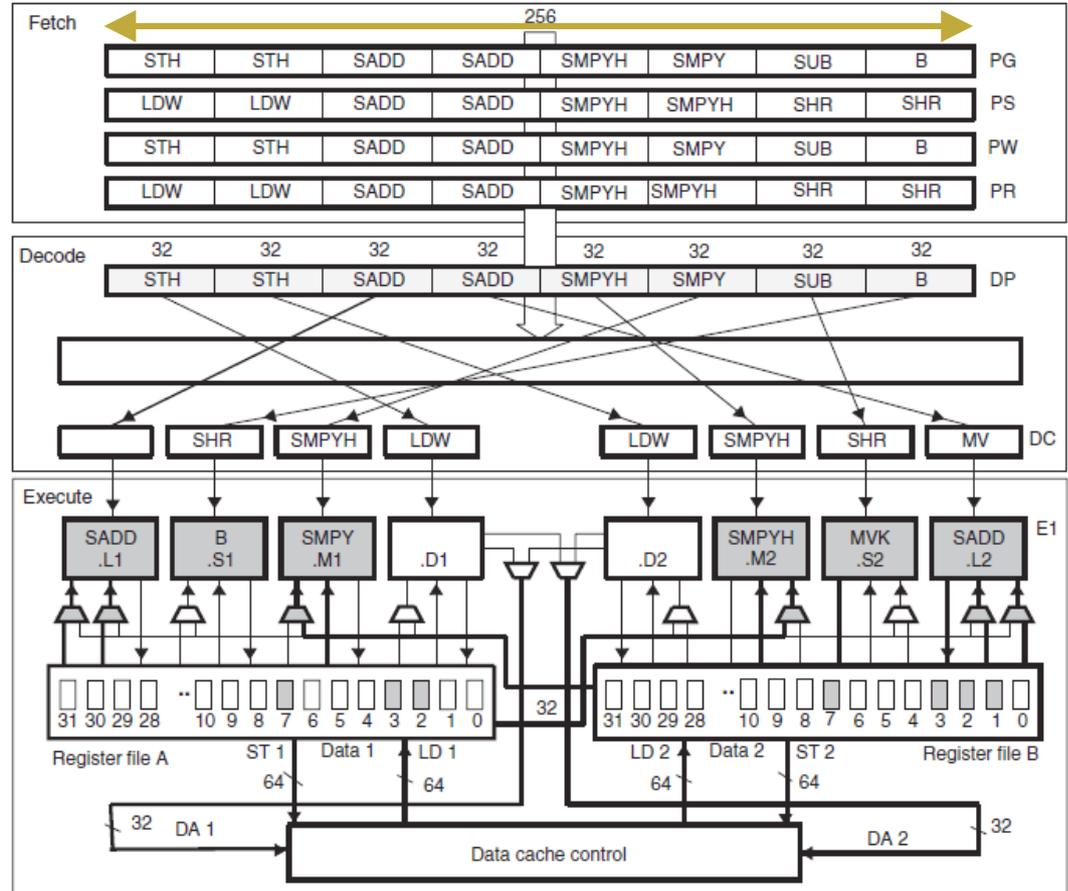
Memory configurable as cache memory or addressable SRAM with no bandwidth loss.

UMA or NUMA models configurable for each core.



C6600 core with:

- 14-stage VLIW hardware pipeline (*Very Long Instruction Word*)
- software pipeline with a max width of 8 instructions



These DSPs are designed for both parallel and daisy-chain work.

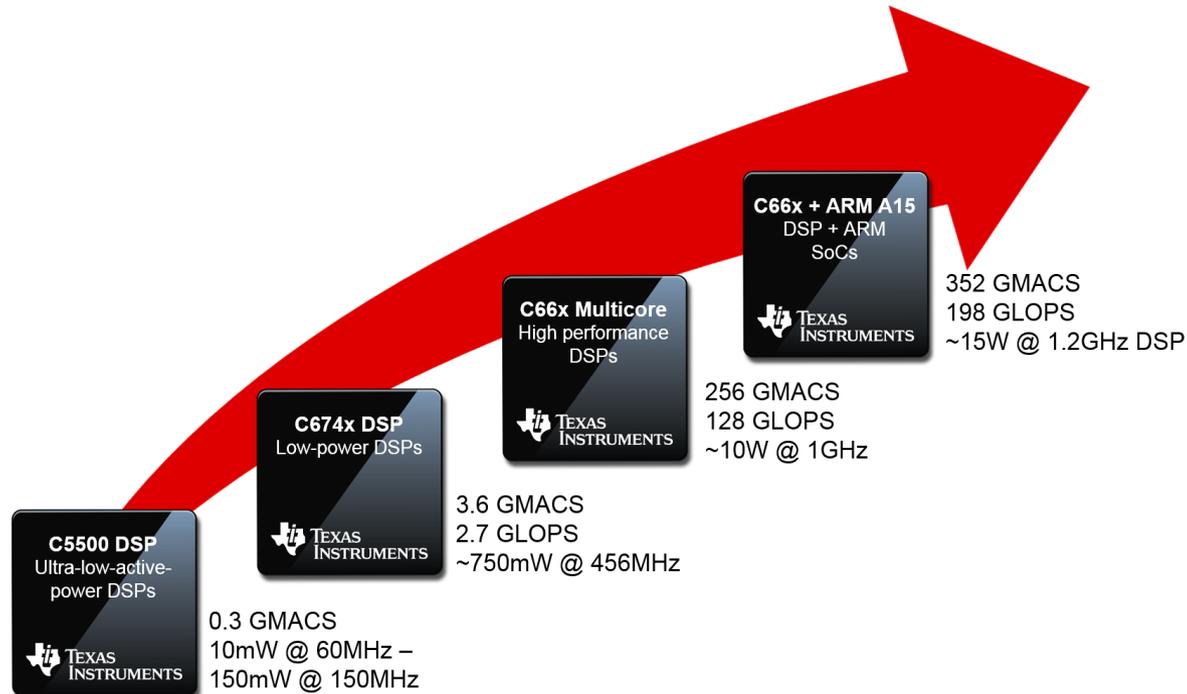
Parallel configuration is suitable for massive parallel processing whereas daisy-chain configuration is more suitable for deep processes algorithms.



Advantage of using daisy-chain configuration:

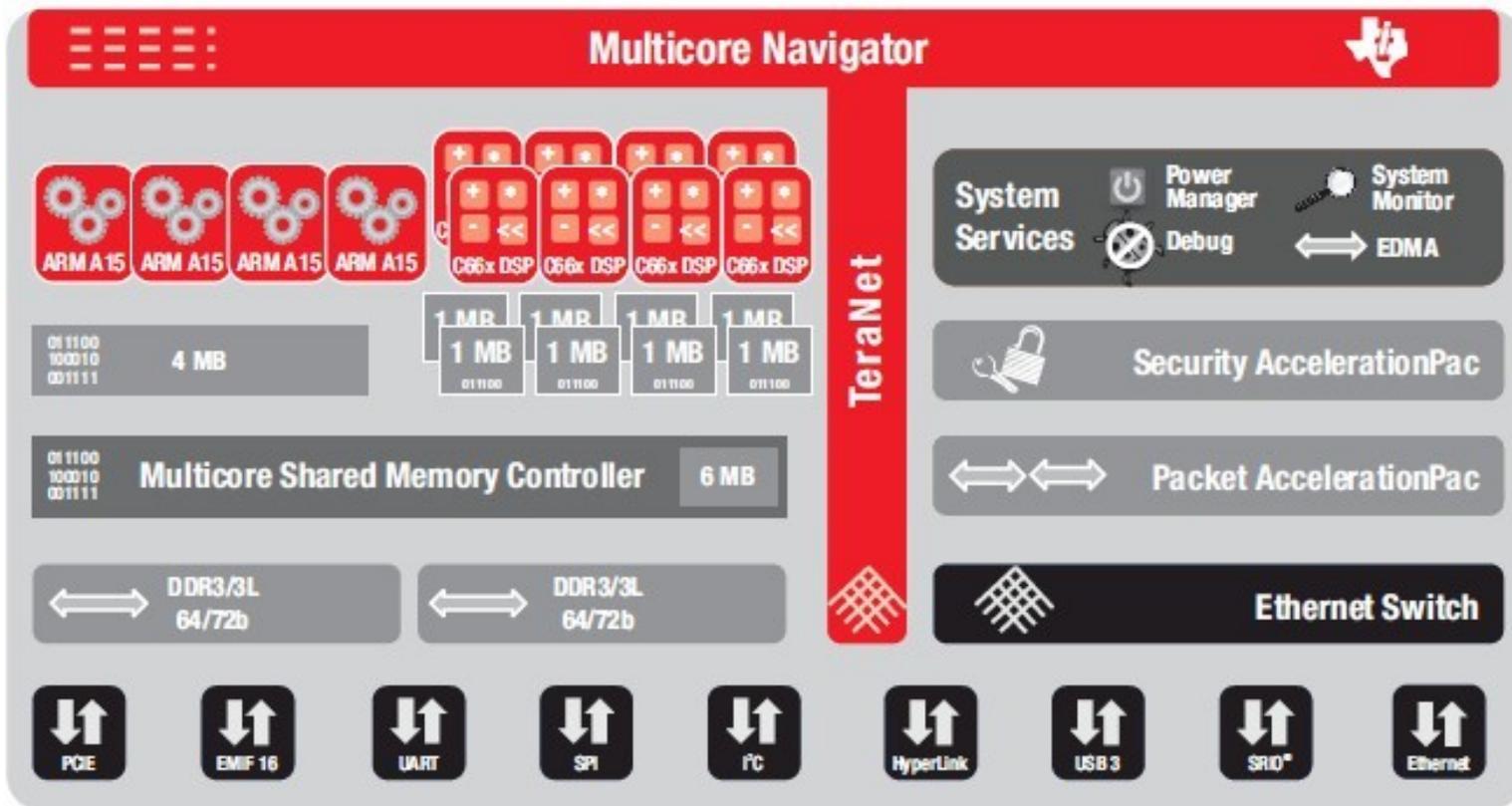
That's not all, TI also offers the Keystone II family. It consists of an AP-SoC with application processors dedicated for digital signal processing applications.

The main target is the telecommunications area.

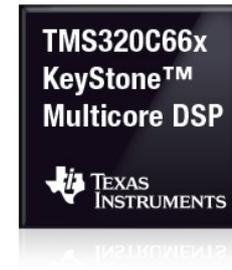


DSP – DIGITAL SIGNAL PROCESSOR

Texas Instruments products: Keystone II



The historical and current leader is by far Texas Instruments.
TI was the first company to design DSP in 1982.

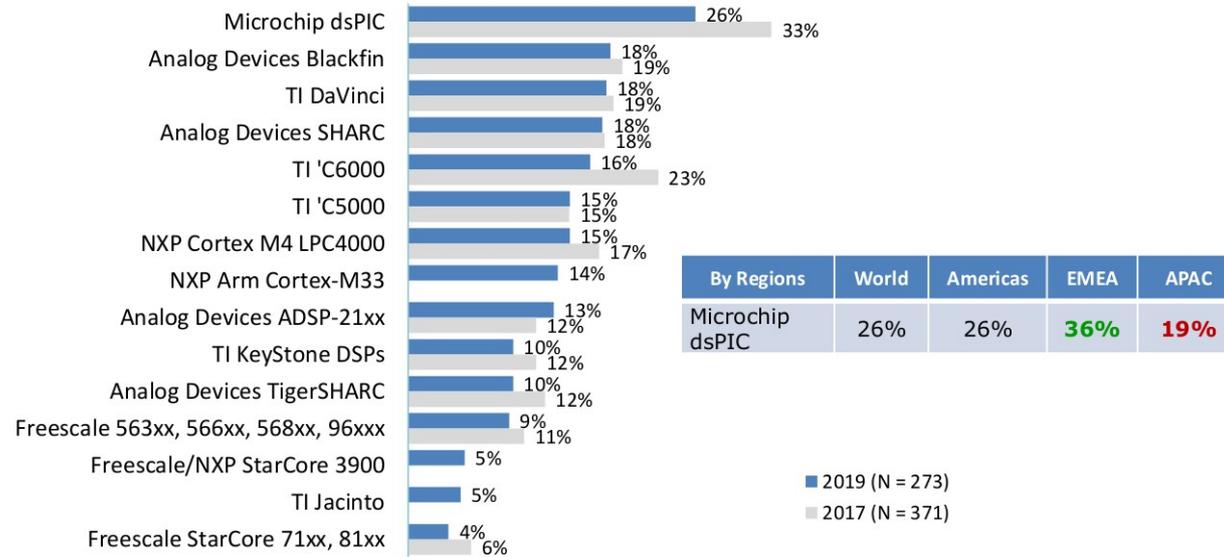


Here is the range of Texas Instruments processors.

Microcontrollers (MCUs)		ARM®-based Processors			Digital Signal Processors		
16-bit Ultra Low Power MCU	32-bit Real-Time MCU	32-bit ARM MCU	32-bit ARM Processors for Performance Applications	Application Processors	Singlecore DSP	Multicore DSP	Ultra Low Power DSP
<ul style="list-style-type: none"> • MSP430™ 	<ul style="list-style-type: none"> • C2000™ 	<ul style="list-style-type: none"> • TMS570 Cortex® R4 • RM4 Cortex® R4F • TMS470M Cortex® M3 Automotive 	<ul style="list-style-type: none"> • Sitara™ Cortex A and ARM9 • KeyStone Cortex® A15 and Cortex® A15 + DSP 	<ul style="list-style-type: none"> • OMAP™ Processors • DaVinci™ Video Processors 	<ul style="list-style-type: none"> • C6000™ Power Optimized 	<ul style="list-style-type: none"> • KeyStone Multicore DSP+ARM • C6000™ Multicore 	<ul style="list-style-type: none"> • C5000™



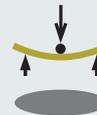
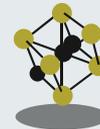
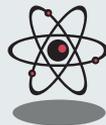
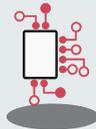
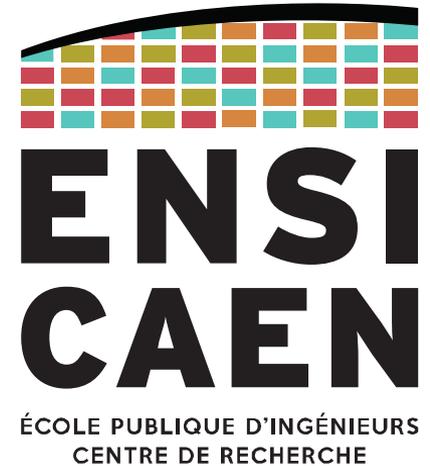
Which of the following DSP chip families would you consider for your next embedded project?



EXECUTION MODELS

Classifying processors according to their execution model

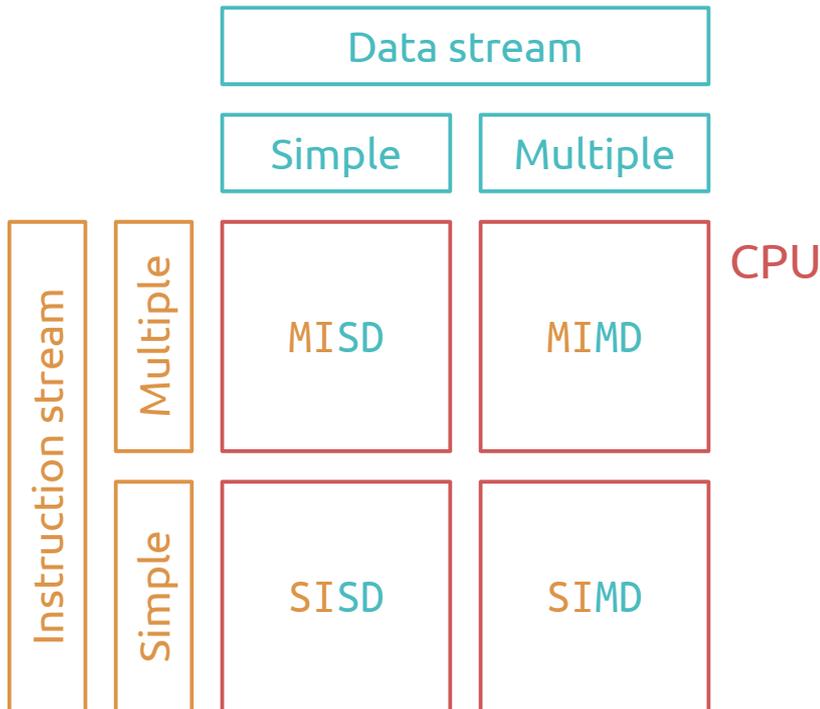
SISD – SIMD – MISD – MIMD



The next slides are not intended for proper lecturing.
However you'll hear those terms quite a lot, so here are a few slides about execution models.



Flynn's classification (1972)



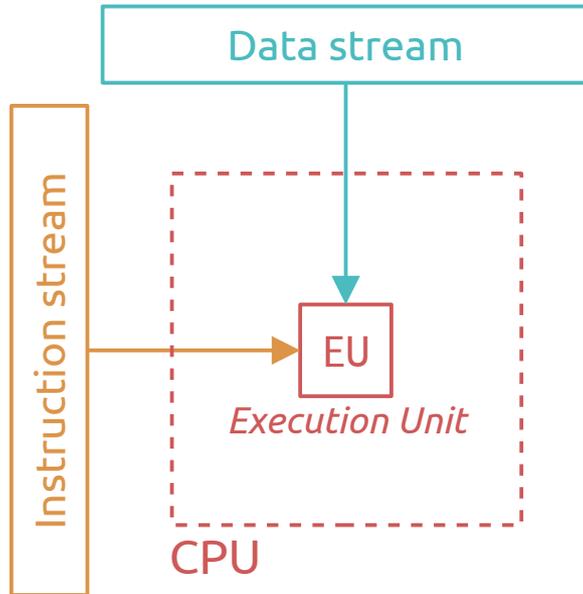
Simple data stream: each operand contains only one piece of data (one memory cell per operand).

Multiple data streams: each operand contains multiple pieces of data (a fixed-size array per operand).

Single instruction stream: the CPU can execute one instruction at once (sequential execution).

Multiple instruction streams: the CPU can execute multiple instructions at once, either using data parallelism (e.g. *forall* loop) or using control parallelism (e.g. parallel sections).

SISD – Single Instruction stream, Single Data stream



The processor execute one instruction at once, each instruction operand containing a single memory cell.

This is the typical mono-processor architecture:

→ Von Neumann architecture

→ MCUs and old GPP generations

→ Sequential processor (no parallelism)

→ **Scalar processor**

→ A single piece of data (a single memory cell) for each operand

SISD – Single Instruction stream, Single Data stream

*Example: TI C6600 assembly language
Adding two floats*

```
; Single Precision ADD  
ADDSP    A17, A5, A5
```

```
; Result:  
; A5 = A5 + A17
```

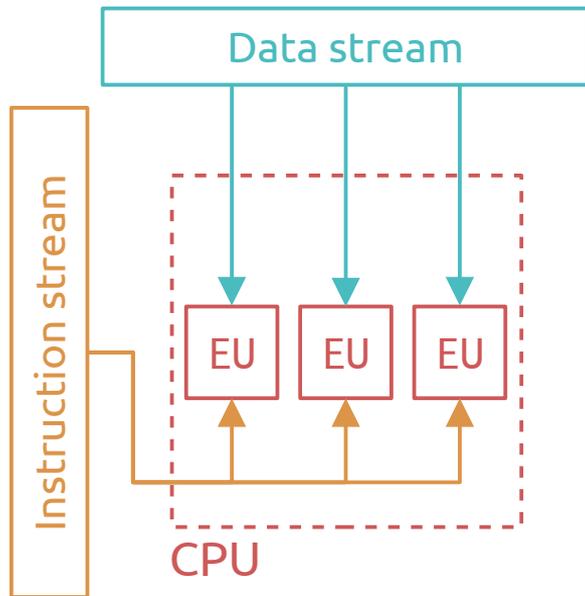
*Example canonical C:
Adding two floats*

```
float a, b ;
```

```
// Initialising a and b ...
```

```
a = a + b ;
```

SIMD – Single Instruction stream, Multiple Data streams



The same instruction will be executed by multiple EUs, each processing its own piece of data. It means the whole CPU will execute a single instruction on multiple pieces of data.

Parallel architecture with centralised control unit:

→ Vectorial processor

→ GPU

→ Intel SSE and AVR instructions set architecture for x86

SSE = Streaming SIMD Extension (SSE, SSE2, SSE3, SSE4)

AVR = Advanced Vector Extensions (AVX, AVX2, AVX512)

SIMD – Single Instruction stream, Multiple Data streams

Example: TI C6600 assembly language
Adding two couples of floats

```
; Dual ADD Single Precision
DADDSP    A21:A20, A25:A24, A25:A24
```

```
; Result:
; A25 = A25 + A21
; A24 = A24 + A20
```

```
; Just like the SSE for Intel, the C6600
; DSP has a C extension (C functions)
; for vectorial instructions
```

Example: x86 SSE C, adding four couples of floats

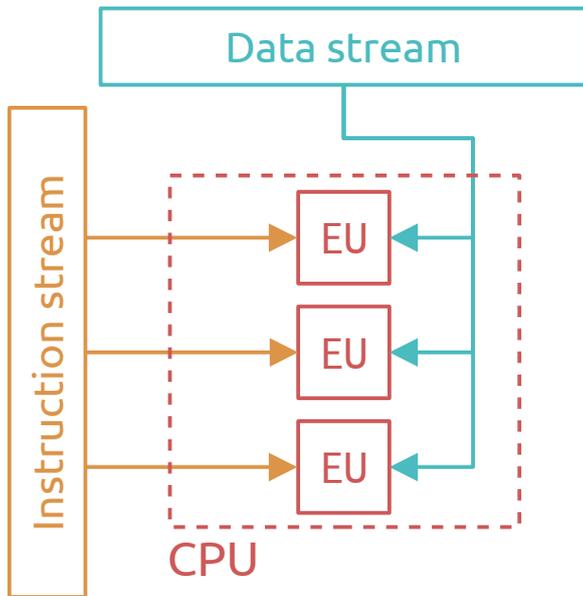
```
float A[N], B[N], C[N] ;

for( int i = 0 ; i < N ; i += 4 ) {
    __m128 reg_b = _mm_load_ps( &B[i] );
    __m128 reg_c = _mm_load_ps( &C[i] );
    __m128 reg_a = _mm_add_ps( reg_b , reg_c ) ;
    __mm_store_pd( &A[i] , reg_a );
}
```

Lanes per type in a 128-bit SIMD register

int8x16	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15
int16x8	s0		s1		s2		s3		s4		s5		s6		s7	
int32x4 / float32x4	s0 (x)				s1 (y)				s2 (z)				s3 (w)			
float64x2	s0 (x)								s1 (y)							

MISD – Multiple Instruction streams, Single Data stream



Each EU execute its own instruction, with single pieces of data.

Few practical applications

→ code redundancy (for detection of execution errors)

→ **VLIW processors** (*Very Long Instruction Word*)

e.g. C66xx Texas Instruments DSP

MISD – Multiple Instruction streams, Single Data stream

*Example: TI C6600 assembly language
Simultaneously adding and multiplying*

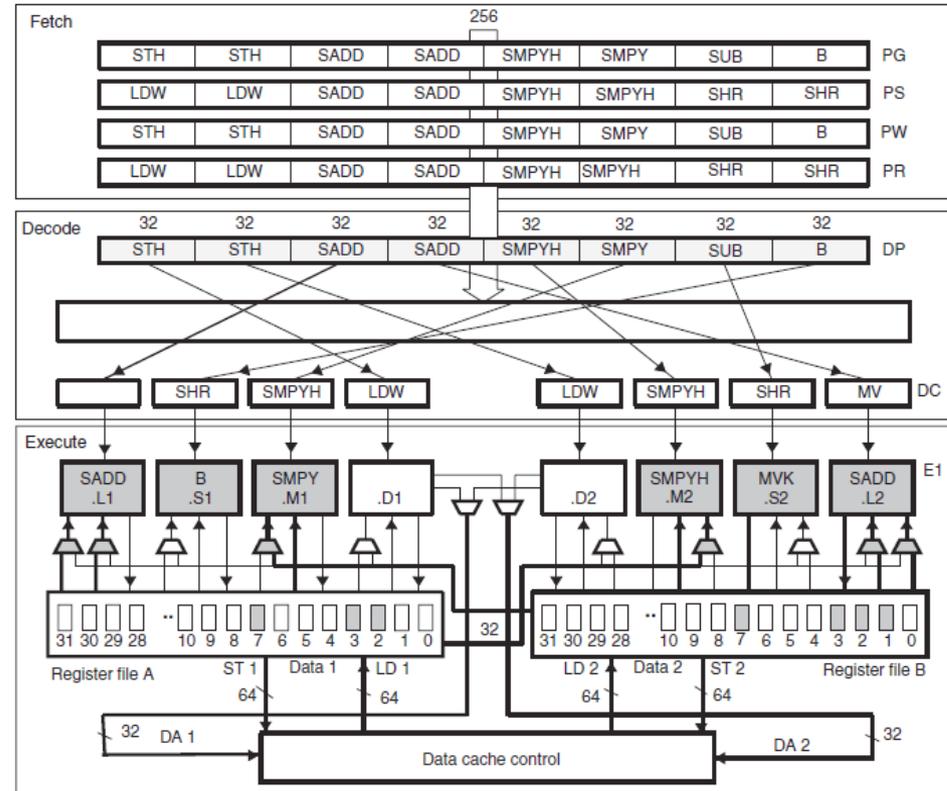
```

; ADD Single Precision
; MULTIPLY Single Precision
||  ADDSP  A3, A9, A3
||  MPYSP  B3, B9, B3
    
```

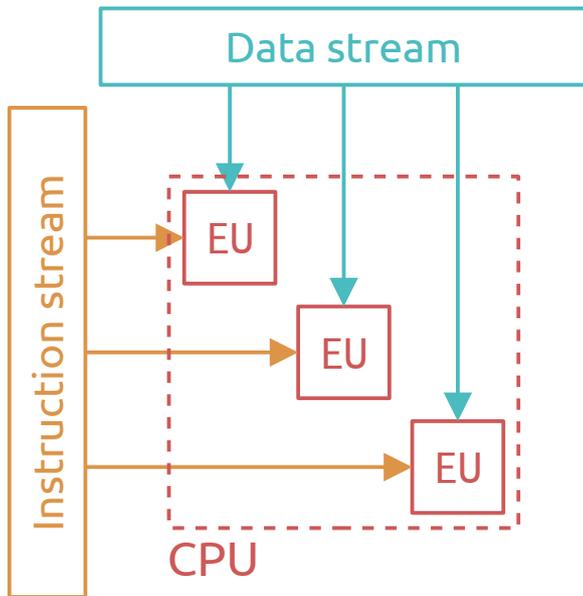
; The pipes (||) explicitly indicate that
; instructions must be executed in parallel
; (use of software pipeline)

```

; Result
; A3 = A9 + A3
; B3 = B9 + B3
    
```



MIMD – Multiple Instruction streams, Multiple Data streams



Each EU executes its own instructions flow on their own data flow.

Execution Unit can be grouped as a cluster.

Parallel architectures with independent control units

→ Super-scalar processors

→ Any modern GPP: x86-x64 (CISC), Cortex-A (RISC)

→ Includes use of SPMD (*Single Program, Multiple Data*)

MIMD – Multiple Instruction streams, Multiple Data streams

*Example: TI C6600 assembly language
Simultaneously adding and multiplying two
different couples of data*

```

; Dual ADD Single Precision
; Dual SUBTRACT Single Precision
  DADDSP    A21:A20, A25:A24, A25:A24
||  DSUBSP   B25:B24, B23:B22, B23:B22

; The pipes (||) explicitly indicate that
; instructions must be executed in parallel
; (use of software pipeline)

; Result
; A25 = A25 + A21
; A24 = A24 + A20
; B23 = B25 - B23
; B22 = B24 - B22

```

*Example: C and OpenMP
Parallelisation of for loop*

```

#pragma omp parallel reduction(+:acc)
{
    #pragma omp for schedule(static)
    for( k = 0; k < size; k ++ )
    {
        acc += A[i * size + k] * x[k];
    }
}

```