

Universidade Tecnológica Federal do Paraná – Toledo
Engenharia da Computação – COENC

Sistemas Embarcados

Arquitetura de Microprocessadores

Tiago Piovesan Vendruscolo

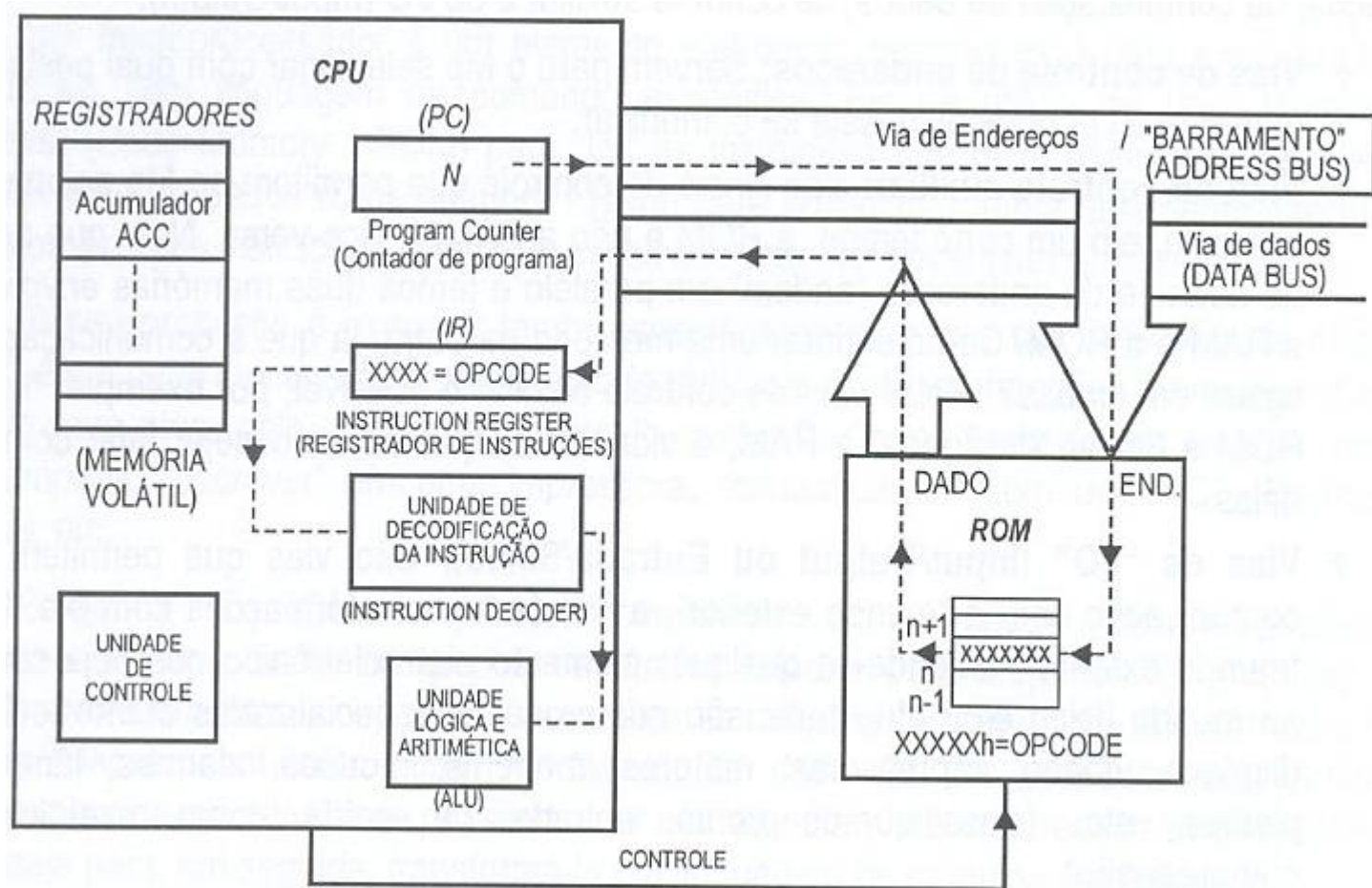


Esta licença permite que outros remixem, adaptem e criem a partir do trabalho para fins não comerciais, desde que atribuam o devido crédito aos autores originais. [4.0 international](https://creativecommons.org/licenses/by-nc-nd/4.0/)

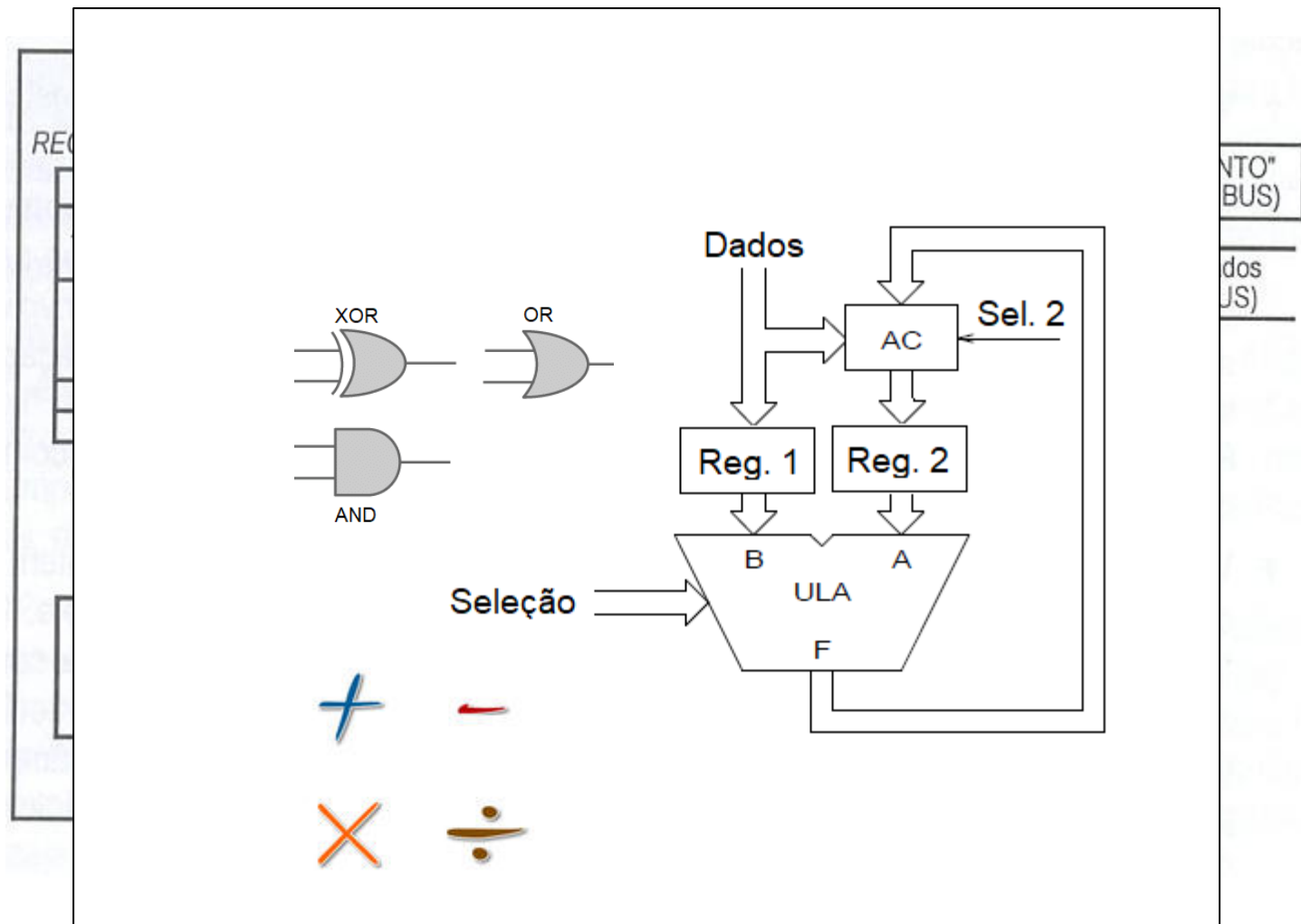
CISC x RISC

- No começo da década de 80, a tendência era construir processadores com conjuntos de instruções cada vez mais complexos. Alguns fabricantes porém, resolveram seguir o caminho oposto, criando o padrão RISC.
- Atualmente, 2 arquitetura são mais comumente usadas em processadores: RISC e CISC.
- CISC (Complex Instruction Set Computer) x RISC (Reduced Instruction Set Computer).
- CISC: Capaz de executar várias centenas de instruções complexas. Sendo extremamente versátil. Exemplos: Alguns dos primeiros processadores para PC: 386 e 486.

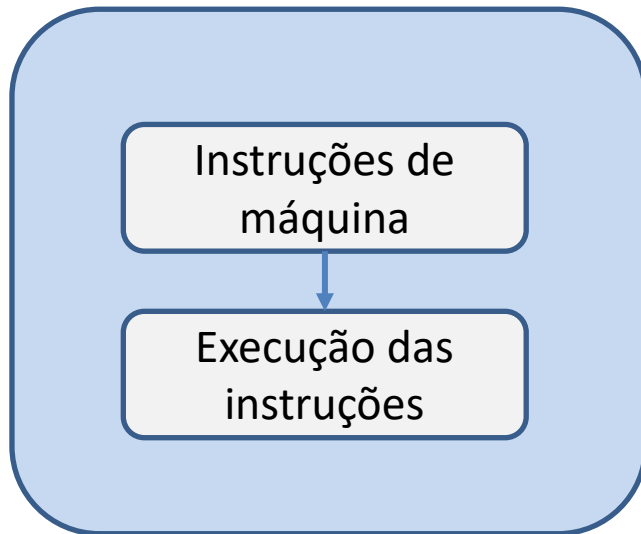
Arquitetura Conceitual - Processador



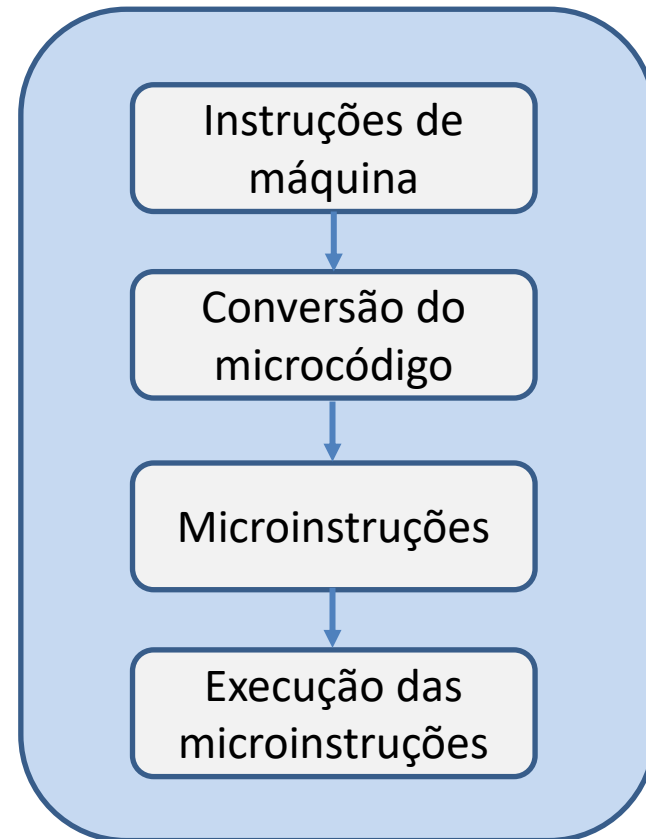
Arquitetura Conceitual - Processador



RISC

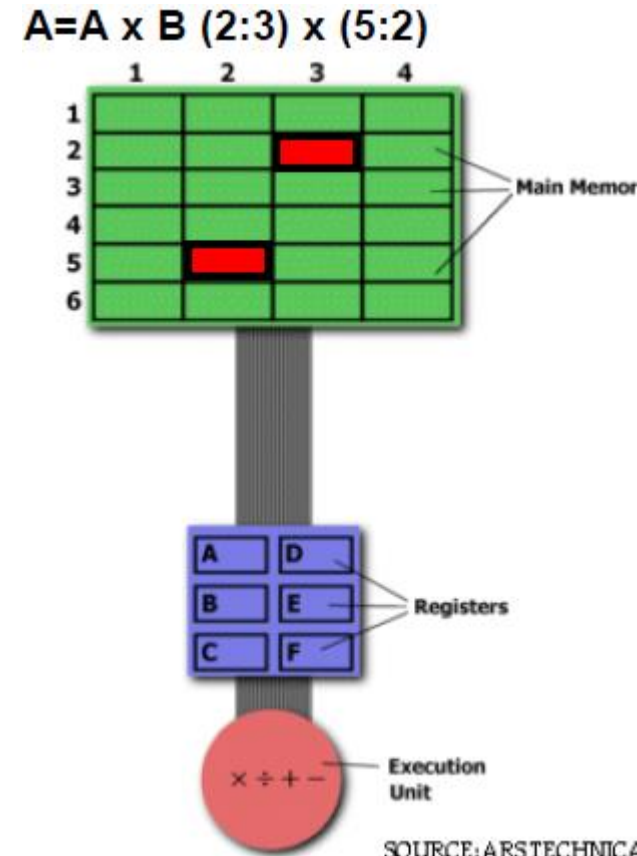


CISC



- RISC: Processador com um pequeno número de instruções → Apenas as instruções utilizadas mais frequentemente. Com o uso de softwares apropriados, é possível executar quase todas as instruções de um processador CISC.
- Problema: Como realizar a multiplicação de 2 números na memória: Dividir a instrução complexa em instruções simples mais rápidas (LOAD, PROD, STORE)

```
LOAD A, 2:3  
LOAD B, 5:2  
PROD A, B  
STORE 2:3, A
```



- Requer mais linhas de código e mais memória RAM para guardar as instruções.
- Maior velocidade (frequência);
 - Maior desempenho pode ser obtido se cada instrução ocupar exatamente uma palavra de memória.
 - Mais ciclos de clock para processar instruções complexas → precisa ser quebrada em instruções simples.
 - Circuitos mais simples → baixo custo.
 - Menor consumo de energia.
 - Qual a arquitetura de um processador ARM? RISC
 - Qual a arquitetura dos processadores atuais? Híbridos – núcleo RISC com decodificador CISC.

- Premissas da abordagem RISC
 - Os operandos de instruções aritméticas e lógicas devem estar nos registradores do processador. Instruções possuem tamanho fixo.
 - Possibilita a execução de instruções em *pipeline*.
 - Não há microprograma para interpretar as instruções.

- Lavanderia

1. Roupas de A, B, C e D



2. Lavadora (30 minutos)



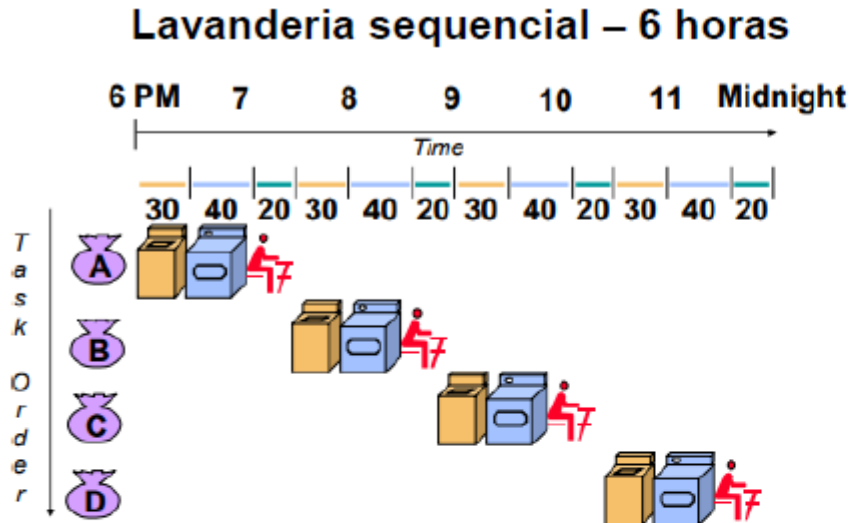
3. Secadora (40 minutos)



4. Passar roupa (20 minutos)

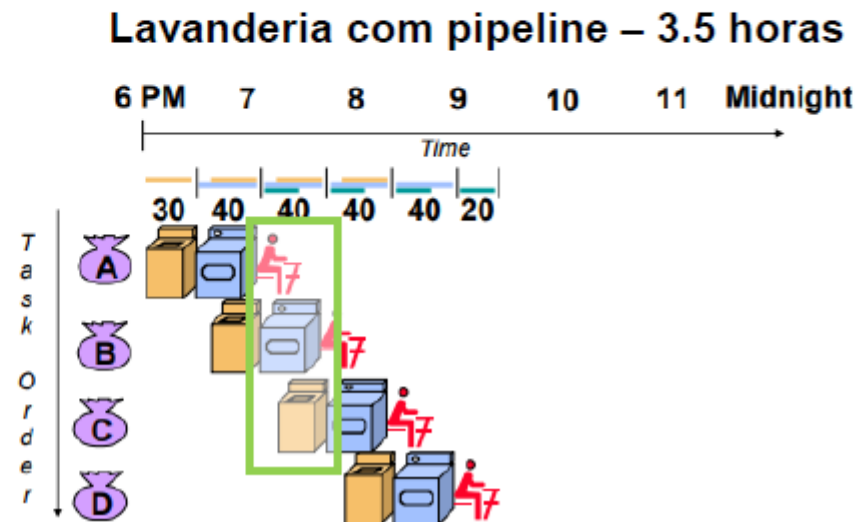


Analogia de PIPELINE



Algumas considerações:

- O pipeline não reduz o tempo de execução de uma tarefa, porém reduz o tempo de execução de toda a carga de trabalho.
- A velocidade do pipeline é limitado pelo estágio mais lento.
- O pipeline trava de houver conflito de recursos (registradores, cache, barramentos...).
- O pipeline trava de houver dependência de dados (é necessário aguardar uma operação antes de utilizar determinada variável...).



CISC

- Problema: Como realizar a multiplicação de 2 números na memória.

Operação destino, fonte

$C = A * B$

MOVE C, B

MULT C, A

Executa as operações
Load e Store

- “MULT” é uma instrução complexa que opera na memória diretamente (programador não precisa explicitamente carregar ou salvar os dados em memória)
- As instruções complexas são implementadas em Hardware.
- Compilador tem sua tarefa de traduzir a linguagem de alto nível para assembly simplificada.

- Núcleos complexos devido a
 - *Instruções mais complexas que demandam de vários ciclos de clock para serem executadas*
 - *Dezenas de modos de endereçamento*
 - *Instruções de tamanhos variados (no RISC é fixo)*

- Resultado
 - *É mais difícil implementar o pipeline*
 - *A taxa média de execução de uma instrução normalmente é bem maior do que 1*
 - *A unidade de controle é microprogramada*
 - *Instruções complexas significam um maior tempo para decodificar e executar, sendo que muito delas raramente são usadas (gerando custo desnecessário).*

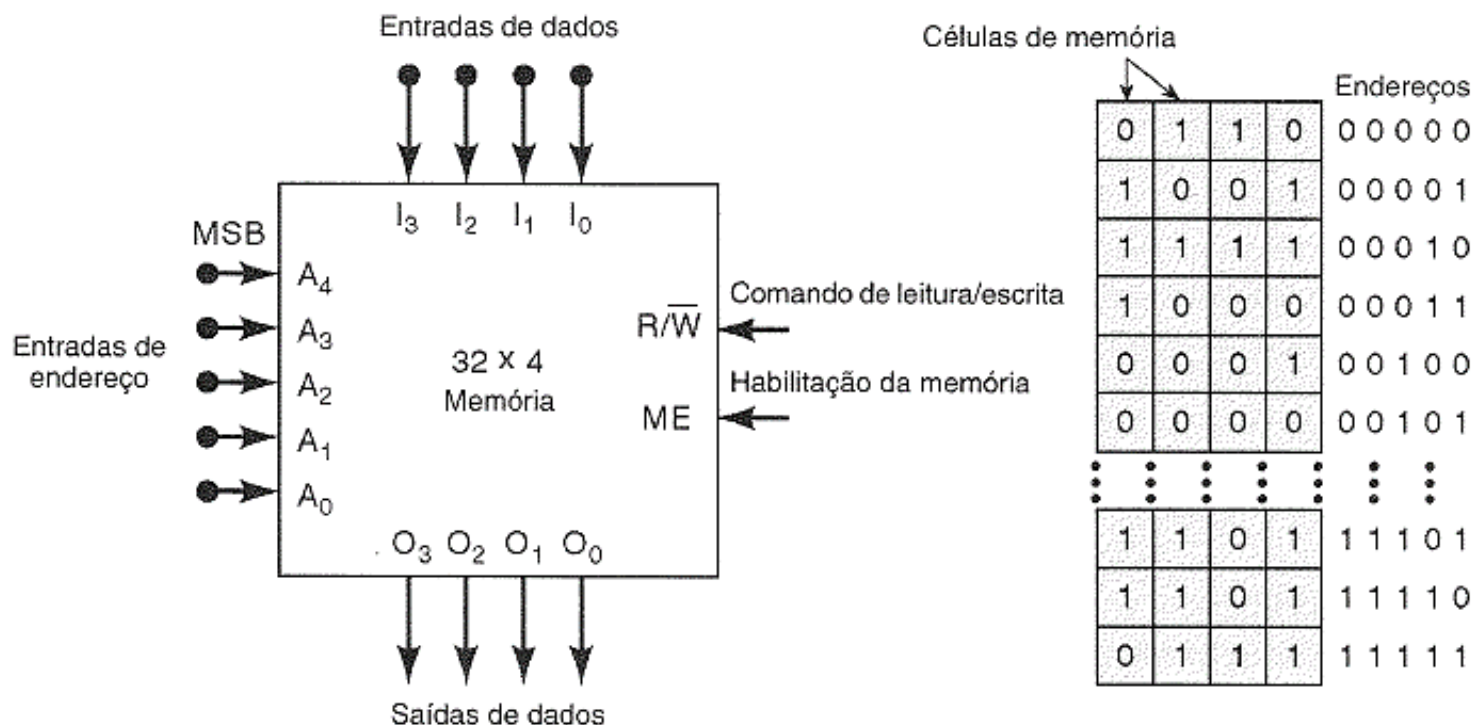
- Vantagens:
 - Processamento de instruções complexas de forma direta;
 - Menor consumo de RAM.
- Desvantagens:
 - Maior número de circuitos internos ➔ Custo elevado;
 - Maior consumo de energia.

- Resumo

- CISC: Facilitam a construção dos compiladores, dessa forma, programas complexos são compilados em programas de máquina mais curtos. Para trabalhar com um grande número de instruções, precisam de uma unidade de controle (hardware!) mais sofisticada, gerando circuitos mais complexos.
- RISC: Unidade de controle simples, rápida e barata. Possui poucos endereços de instruções. Por utilizar poucas instruções, a unidade de controle pode ter previsibilidade em relação ao número de ciclos utilizados, facilitando e aumentando a velocidade no controle dos estágios (*pipeline*). Necessita de compiladores mais sofisticados.

Arquitetura Conceitual - Memória

- Funcionamento:



Fonte: (TOCCI; WIDMER; MOSS, 2011)

- Por que possui 4 bits de entrada/saída e 5 bits de endereço?

- Pontos importantes sobre a Memória:
 - Tempo de acesso: é o tempo necessário para acessar a memória e realizar uma operação de leitura ou gravação.
 - Capacidade: Quantidade efetiva de dados que podem ser armazenados no interior da memória.
 - Não volatilidade: Capacidade de a memória manter seus dados mesmo quando não houver energia elétrica (RAM x ROM).
 - Memória de programa (firmware): ROM.
 - Memória de dados: RAM.
 - Tempo de Latência: É o intervalo mínimo entre cada operação de leitura ou escrita na memória.

Arquitetura Conceitual - Memória

Exemplo: Descreva as condições de cada entrada e saída quando a palavra 1 1 1 0 deve ser escrita na posição de endereço 0 1 1 0 1.

Entrada de endereço: 0 1 1 0 1

Entrada de dados: 1 1 1 0

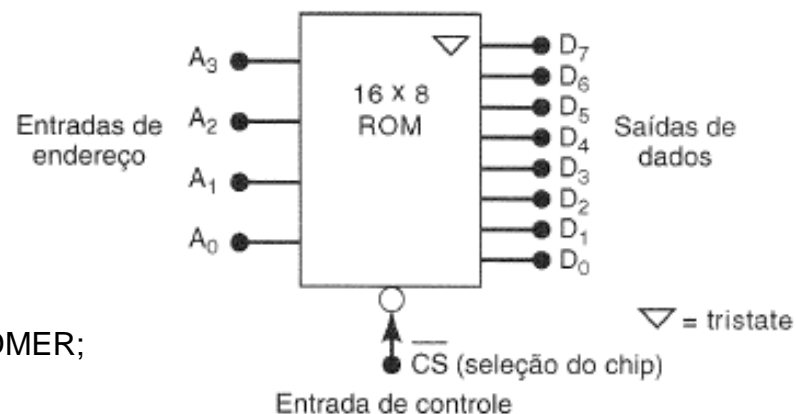
R/\overline{W} : Nível Baixo

Habilitação de memória: Alto

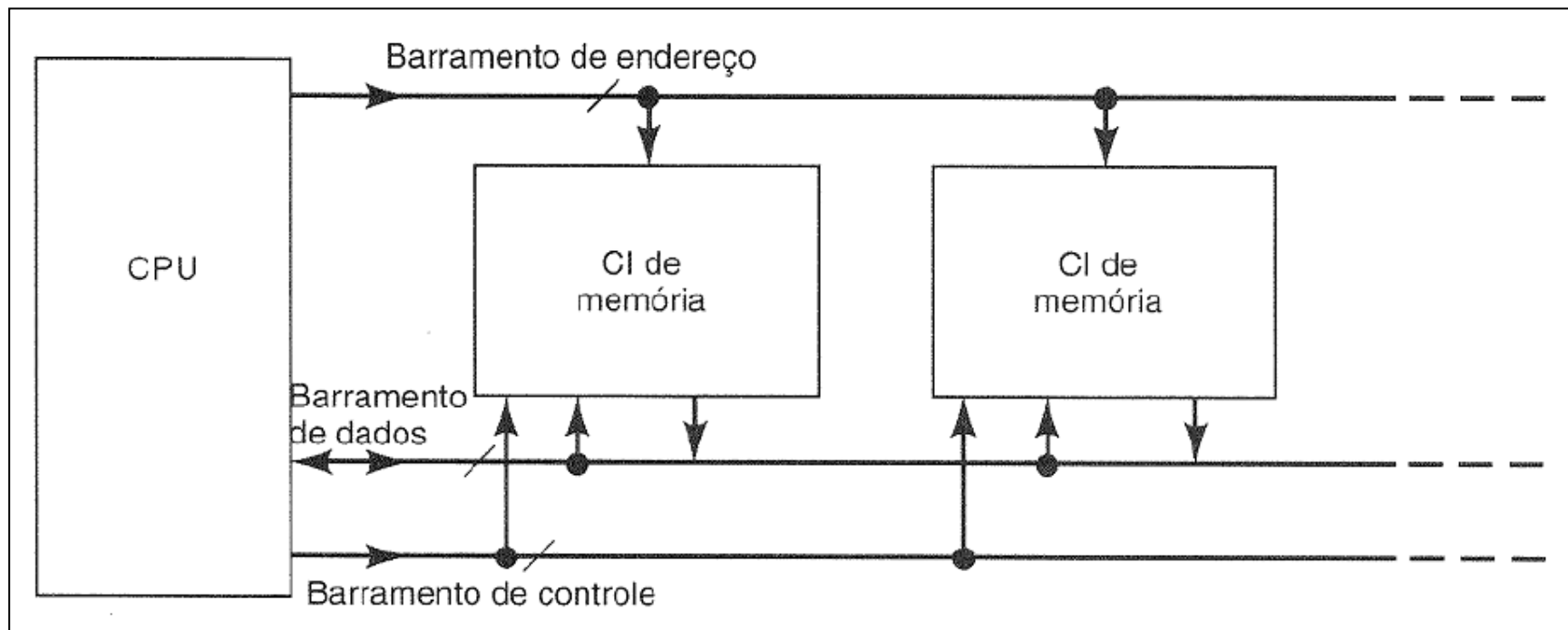
Saída de dados: Xxxx (não utilizado – processo de escrita)

Arquitetura Conceitual - Memória

■ Conexões CPU-Memória



Fonte: (TOCCI; WIDMER;
MOSS, 2011)

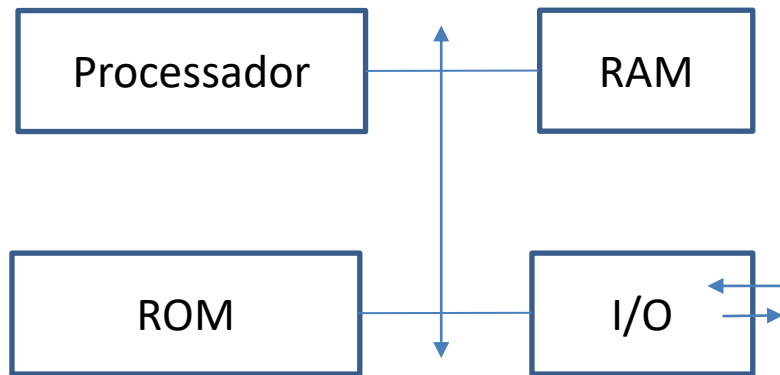


Arquitetura Conceitual – Memória SDRAM e SRAM

- SRAM – *Static Random Access Memory*
 - Não necessita *refresh*
 - Volátil
 - Relativamente imune a ruídos elétricos e outros distúrbios
 - Mais rápida, e mais cara que memórias do tipo DRAM.
 - Utilizada em memória cache L1/L2
 - Trabalha na velocidade do processador
 - Consome menos energia – aquece menos
- SDRAM - *Synchronous Dynamic Random Access Memory*
 - Memória DDR é SDRAM
 - Volátil
 - É necessário um transistor e um capacitor para cada bit – necessita *refresh*
 - Mais sensível a distúrbios
 - Mais lenta e mais barata que as memórias SRAM

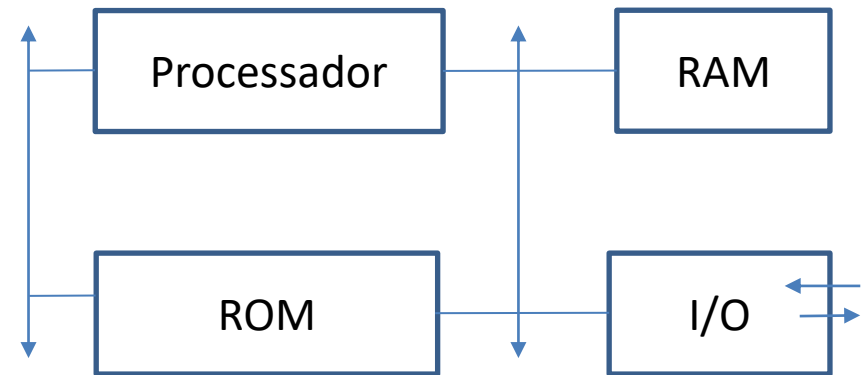
- FLASH
 - Não volátil
 - Tempo de acesso típico de 20ns → máx 50MHz
 - Como utilizar se o processador operar acima de 50 MHz:
 - Inserir wait-cycles durante o acesso a FLASH – perde desempenho
 - Utilizar memória cache para instruções
 - Copiar o código do programa da FLASH para a RAM

Von Neumann



Barramento

Harvard

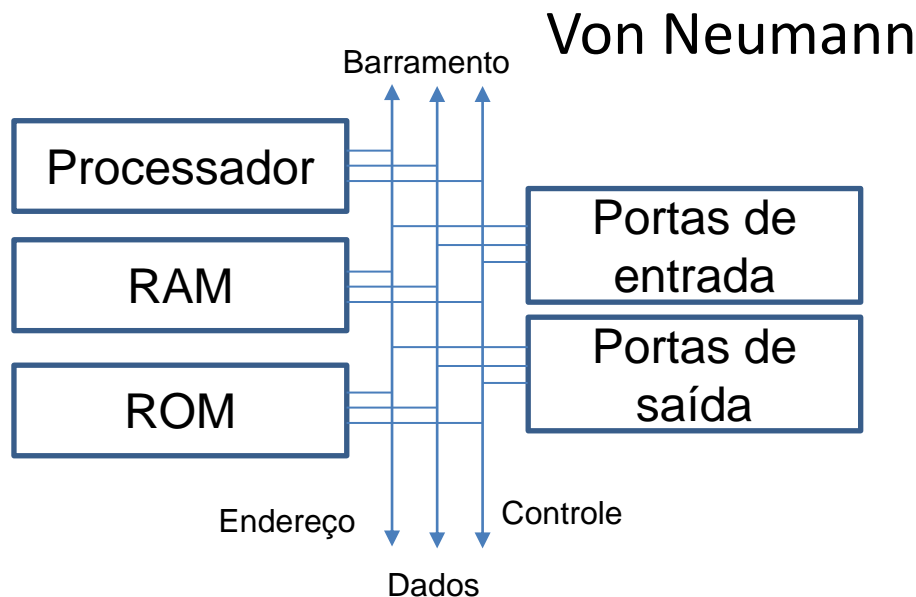


Barramento
de "busca"

Barramento
de sistema

Diversos processos de leitura e escrita
podem ser feitos ao mesmo tempo

Arquitetura Conceitual



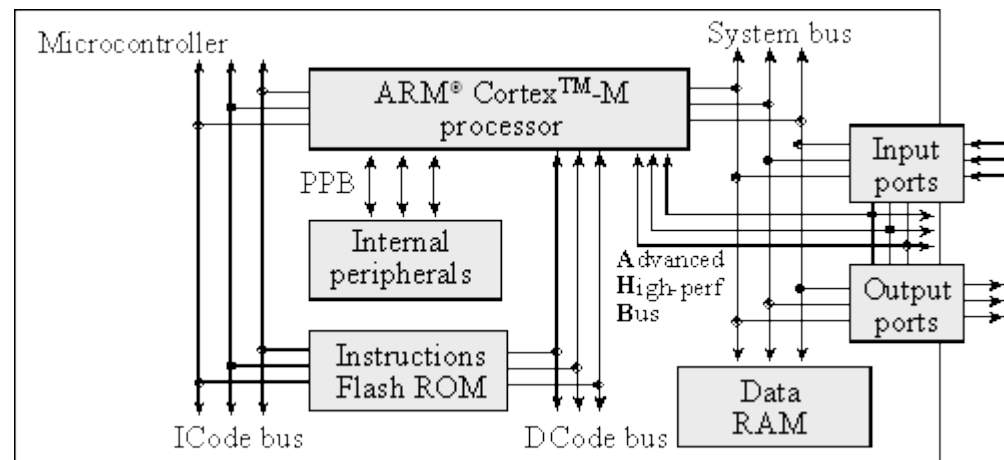
Instruções são obtidas da ROM pelo barramento Icode.

Dados interagem entre a RAM e I/O pelo barramento System.

Barramento Dcode é utilizado para funções debugging.

Periféricos internos se comunicam pelo private peripheral bus (PPB).

Harvard (ARM M4)



Fonte: Dogan Ibrahim, ARM-Based microcontroller projects using MBED, Elsevier, 2019

Arquitetura Conceitual – 8 bits vs 32 bits

- Processadores de 8 bits fazem uso mais eficiente da memória
 - *Também tem menos desperdício de recursos*
- Muitas aplicações utilizam pouca capacidade de processamento
- Microcontroladores de 8 bits são mais acessíveis
- Qual a melhor escolha?
 - *Ambos → Depende da aplicação*

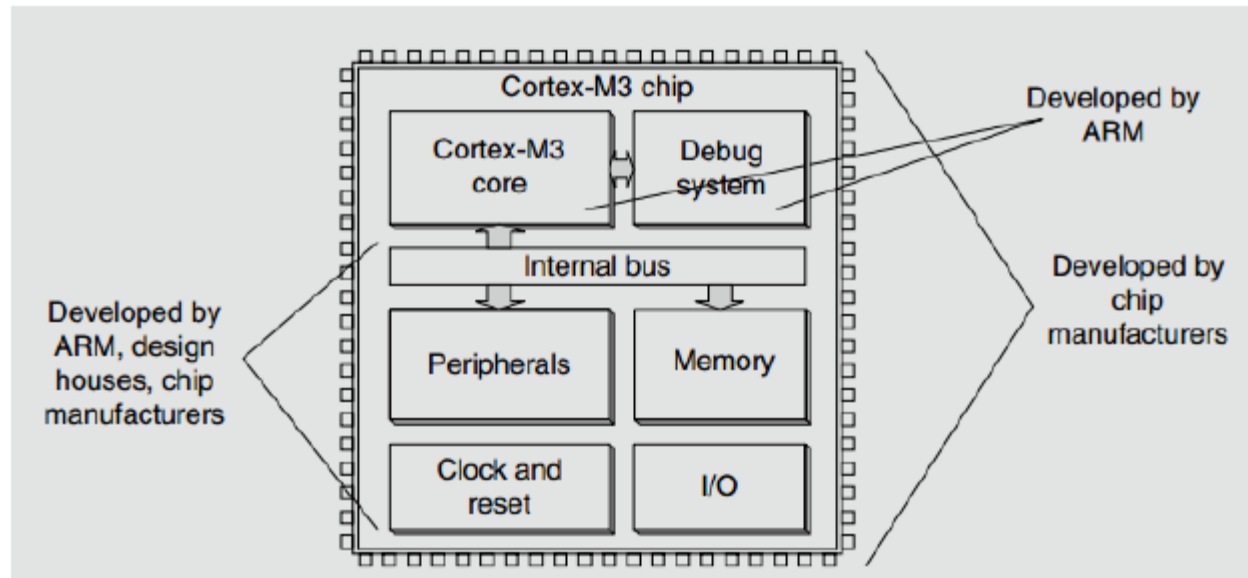
- ARM (Advanced RISC Machine)
 - Arquitetura de 32 bits utilizada principalmente em sistemas embarcados. Em torno de 90% dos processadores RISC embarcados de 32 bits são ARM.
- Características
 - Arquitetura Load-Store: As instruções somente processarão (soma, subtração, etc) valores que estiverem nos registradores e sempre armazenarão os resultados em algum registrador.
 - Formato de instruções de 3 endereços (isto é, os dois registradores operandos e o registrador de resultado são independentemente especificados).



- Arquitetura → documento de especificação
 - Instruções
 - Exceções
 - Registradores
 - Memória
- Não tem custo, pode ser obtido diretamente do website da ARM.
- *Architectural licence*: baseada nas necessidades dos clientes, as empresas podem obter uma licença arquitetural para projetar a própria CPU baseada no conjunto de instruções ARM

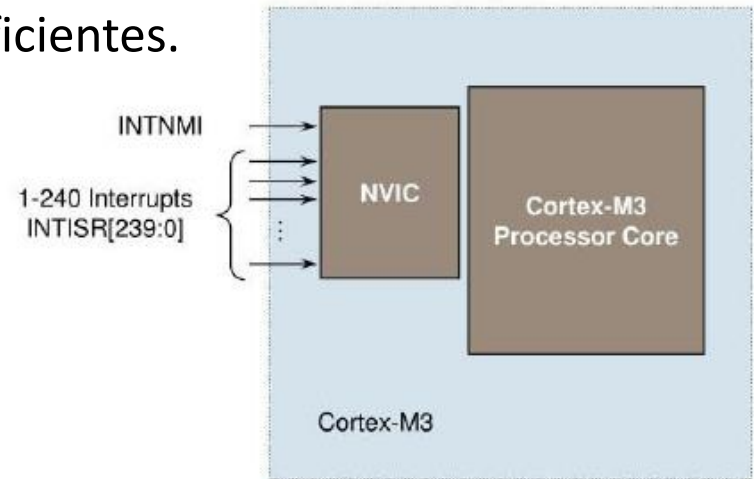
Arquitetura x Organização

- Organização → Implementação física (silício)
- Quantos Chips a ARM fabrica por ano?
- A ARM não fabrica chips, ela trabalha vendendo o licenciamento baseado em direitos de propriedade intelectual.
- *Core licence*: Venda de IP cores (bloco de lógica ou dados que é usado para produzir um FPGA ou um SOC)



Interrupções

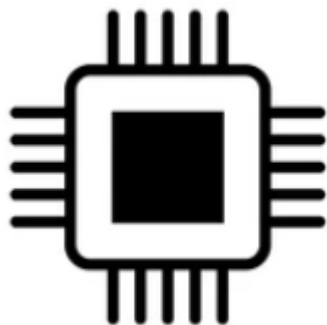
- 1 interrupção não-mascarável (NMI): Sinalizada por periférico ou acionada por software. Interrupção mascarável: Tipo de interrupção de hardware que pode ser desativada (mascarada) temporariamente quando um programa necessita de toda a atenção do microprocessador.
- 1 a 240 interrupções com prioridade
 - Interrupções são mascaráveis
 - Implementação define número de interrupções
- Processador de interrupções (NVIC): Introduzido na família Cortex-M, torna o uso de interrupções mais simples e mais eficientes.



Microprocessadores e Processadores Digitais de Sinais (DSP)

- É o “cérebro” de cada sistema embarcado. Um sistema embarcado tem um ou mais microprocessadores.
- Possuem versões desde as mais simples, com baixo consumo e poder de processamento, utilizadas por exemplo, em sensores simples IoT, até versões mais poderosas, responsáveis por realizar todo o processamento do sistema.
- Na fase de projeto do sistema embarcado, o projetista deve selecionar qual a melhor opção que atenda os requisitos, mantendo o custo x benefício.





- Microprocessador
- Microcontrolador
- FPGA
- ASIC
- PLD / CPLD
- SoC
- DSP

Processador

- 1971 – Primeiro microprocessador – INTEL 4004 (4 bits)
- 1978 - Intel lança o 8086 (16 bits) com clock de 5 MHz.
- São utilizados em situações que demandam alto poder de processamento sem restrições de consumo de energia.
- Como o microprocessador é composto apenas pela unidade de processamento, é necessário fazer o projeto de todo o conjunto do sistema (placa mãe, memória, fonte, etc).
- De acordo com a aplicação, é necessária a montagem de um computador industrial (mais resistente para ambientes hostis).
- Dependendo do computador, ele foge da definição de “sistema embarcado”.



Computador industrial.

Fonte: <https://www.ato.com/embedded-industrial-pc-fanless>



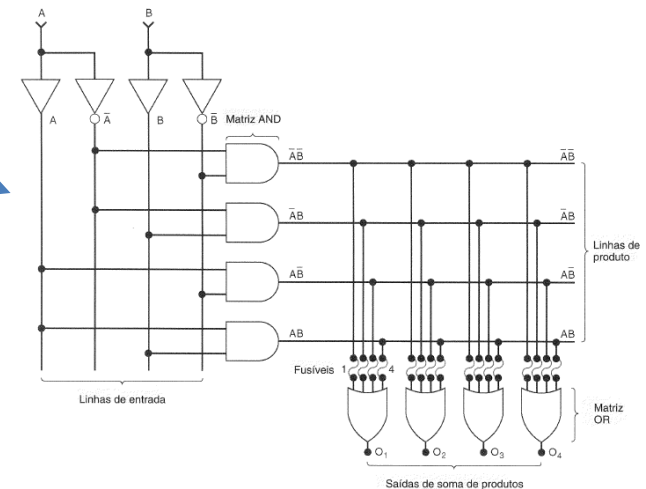
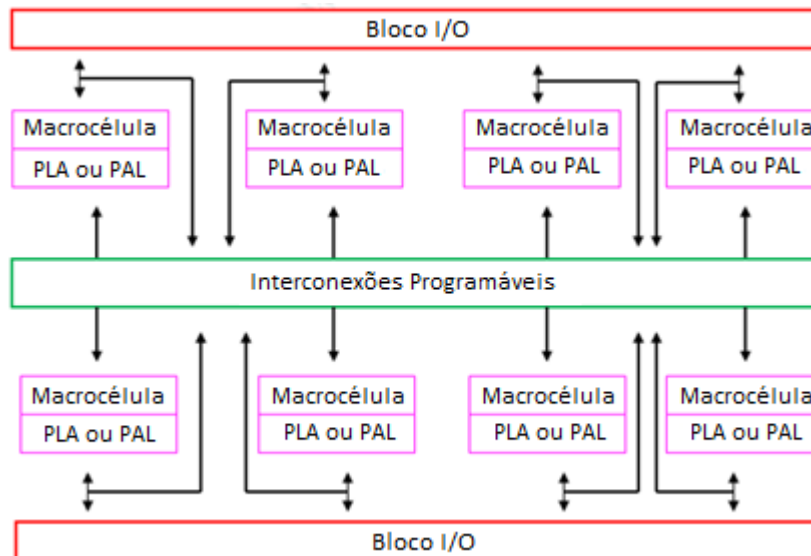
Computador industrial.

Fonte: TinkerTouch S 10.1 - Industrial Panel PC



CPLD (COMPLEX PLD)

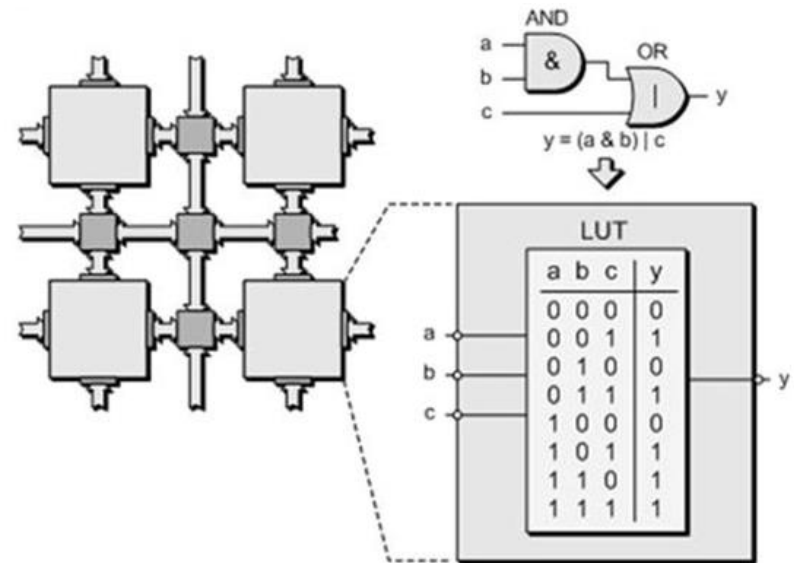
- Dispositivo que possui seu hardware reconfigurável
- Normalmente programado utilizando linguagem de descrição de máquina em VHDL, Verilog, SystemVerilog.
- Principais fabricantes: Xilinx e Altera.



Fonte: (TOCCI; WIDMER; MOSS, 2011)

FPGA – Field-programmable Gate Array

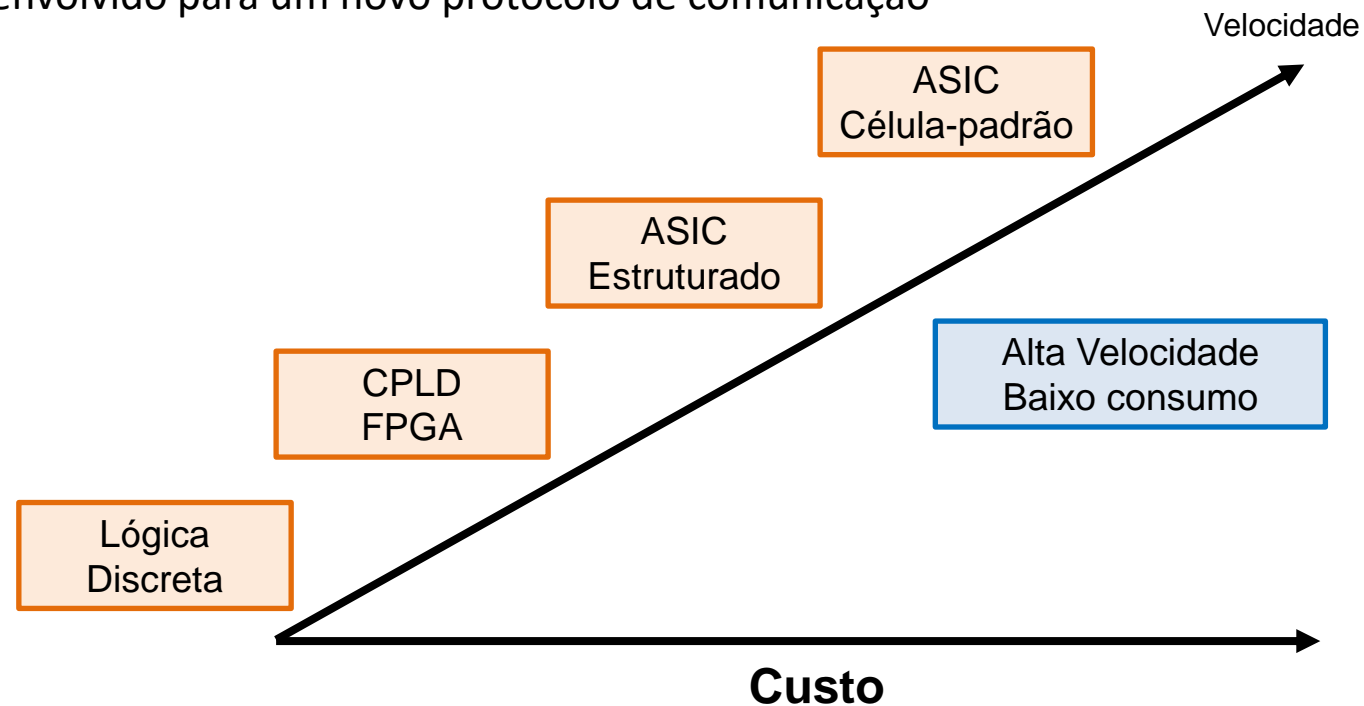
- Dispositivo que possui seu hardware reconfigurável
- Normalmente programado utilizando linguagem de descrição de máquina em VHDL, Verilog, SystemVerilog.
- Bastante útil quando é necessário um alto número de processamento paralelo (real).
- Muito utilizado em sistemas de tempo real que necessite alta confiabilidade sem restrição de custo (sistema de controle de mísseis balísticos).
- Largura de banda (bits) variável.
- Principais fabricantes: Xilinx e Altera.



Fonte: [https://tadviser.com/index.php/Article:Field-Programmable_Gate_Array_\(FPGA\)](https://tadviser.com/index.php/Article:Field-Programmable_Gate_Array_(FPGA))

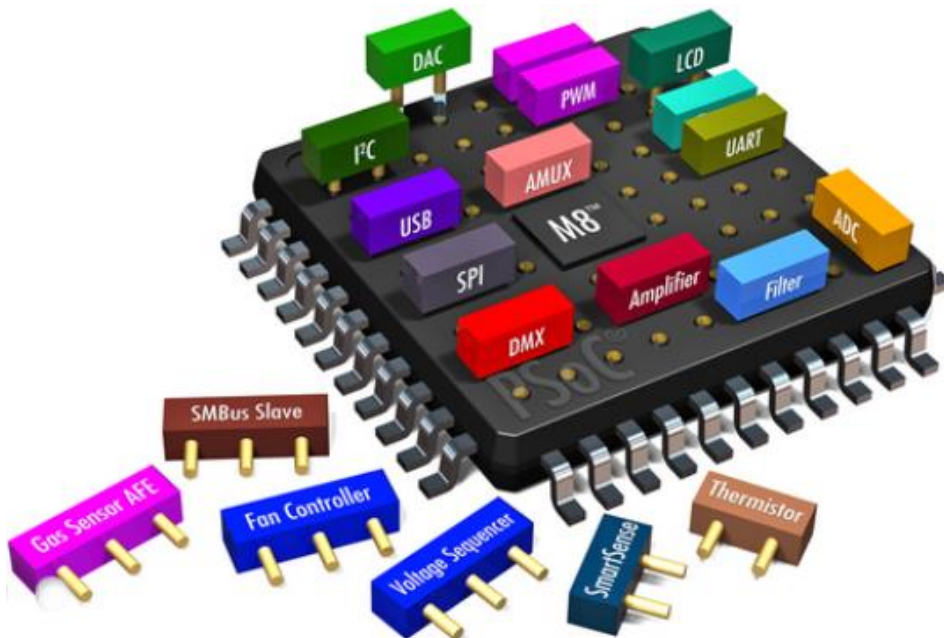
ASIC - Application Specific Integrated Circuits

- Circuitos projetados/fabricados para uma aplicação específica – Linguagem HDL
- Alta velocidade
- Baixo consumo
- Alto custo (fabricação em baixo/médio volume)
- ASSPs (Application-specific standard parts)
 - Equivalentes aos ASIC, porém podem ser utilizados em diferentes projetos
 - Um CI desenvolvido para um novo protocolo de comunicação



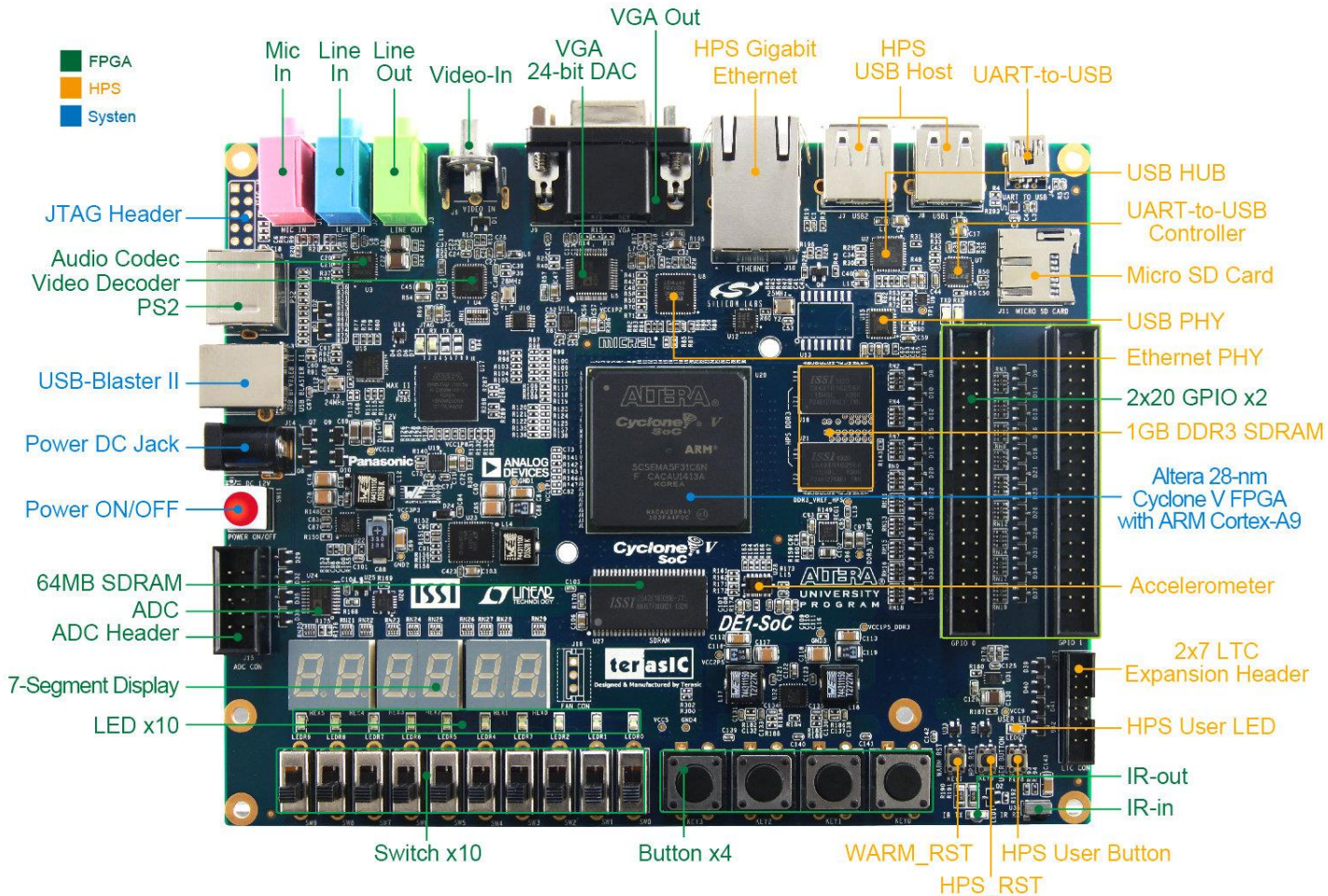
SoC – System-on-a-chip

- Pode conter um ou mais núcleos de processamento, incluindo DSPs e FPGA
- Pode ter grande capacidade de memória
- Existem modelos com diversos módulos analógicos integrados, o que reduz o custo de projeto/implementação dos periféricos.

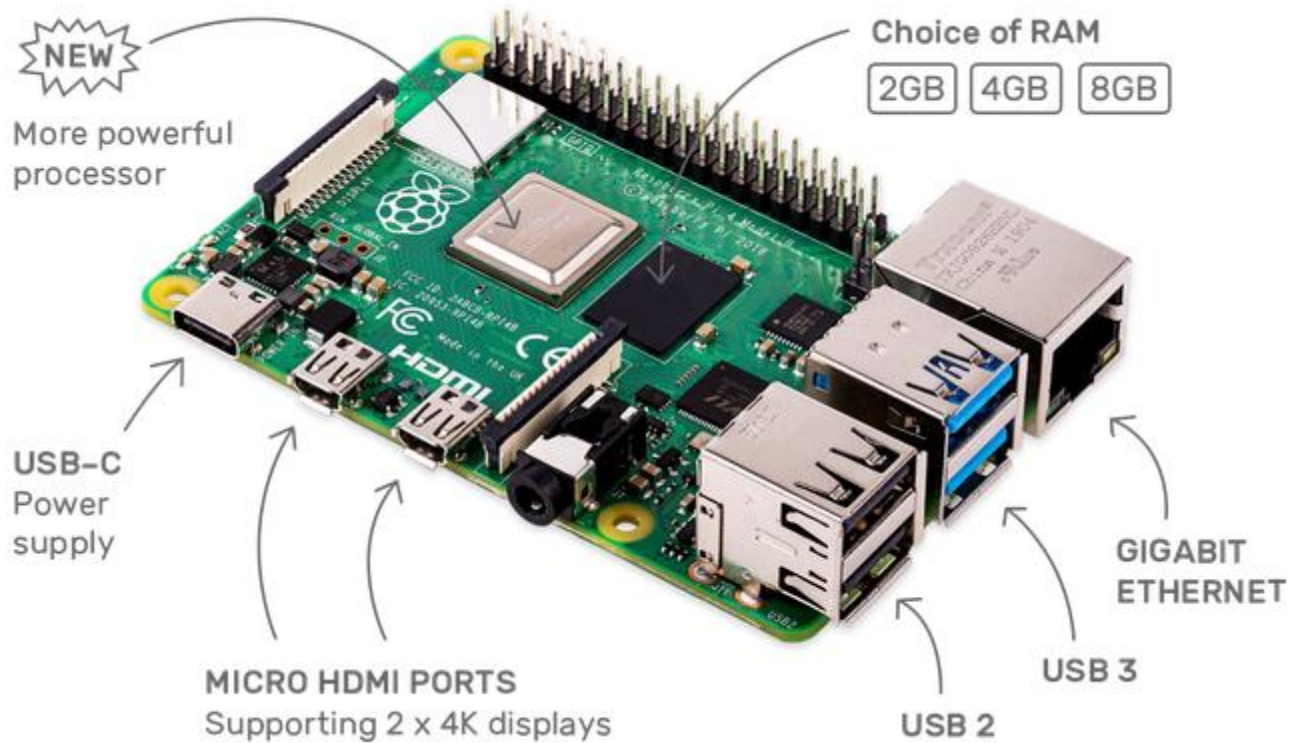


SoC – System-on-a-chip

- DE1-SoC - ALTERA
- FPGA Cyclone V SoC 5CSEMA5F31C6 – 85k Elementos lógicos.
- Dual-core ARM Cortex-A9



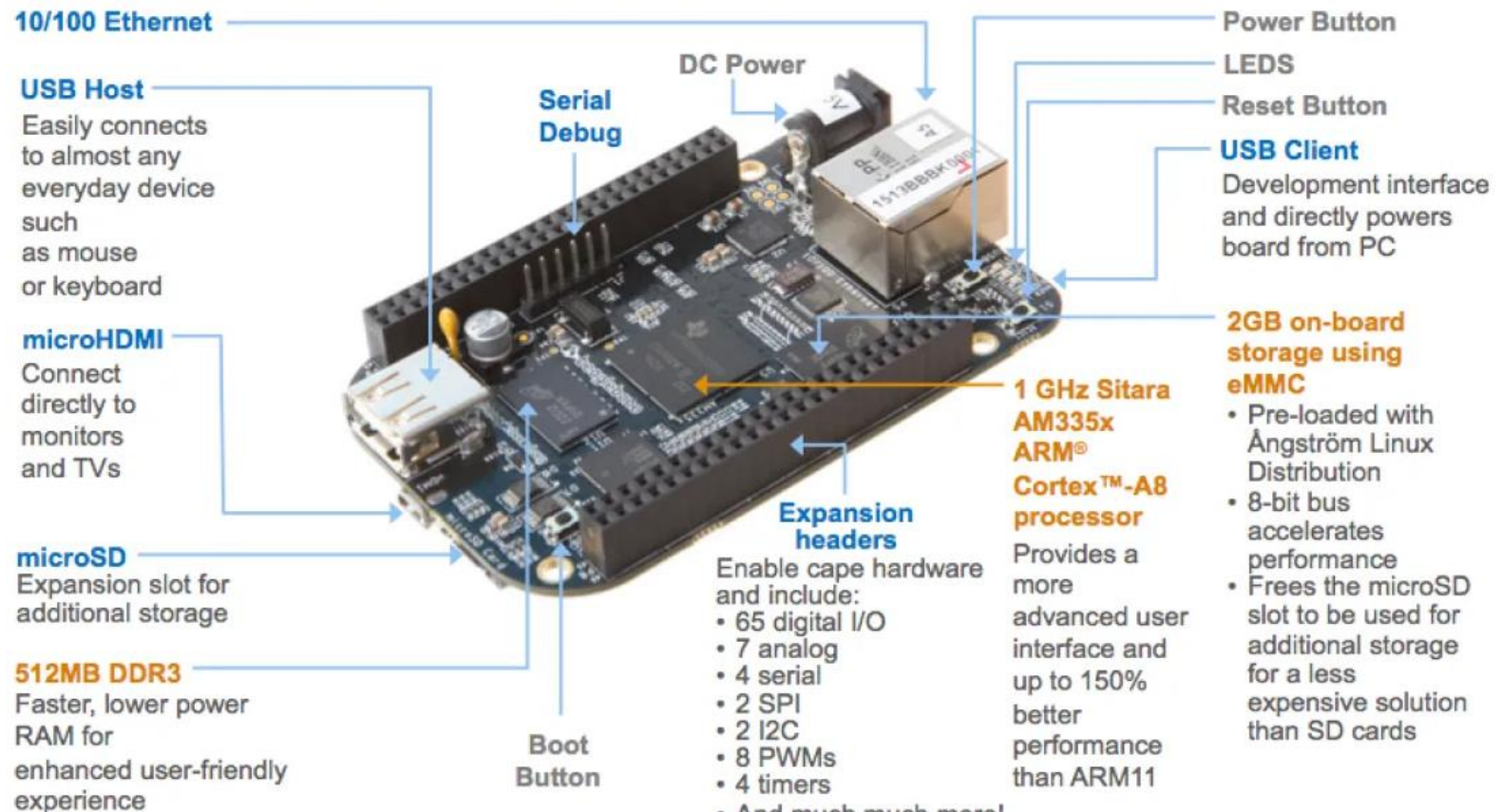
- Raspberry Pi 4



<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>

- Beaglebone

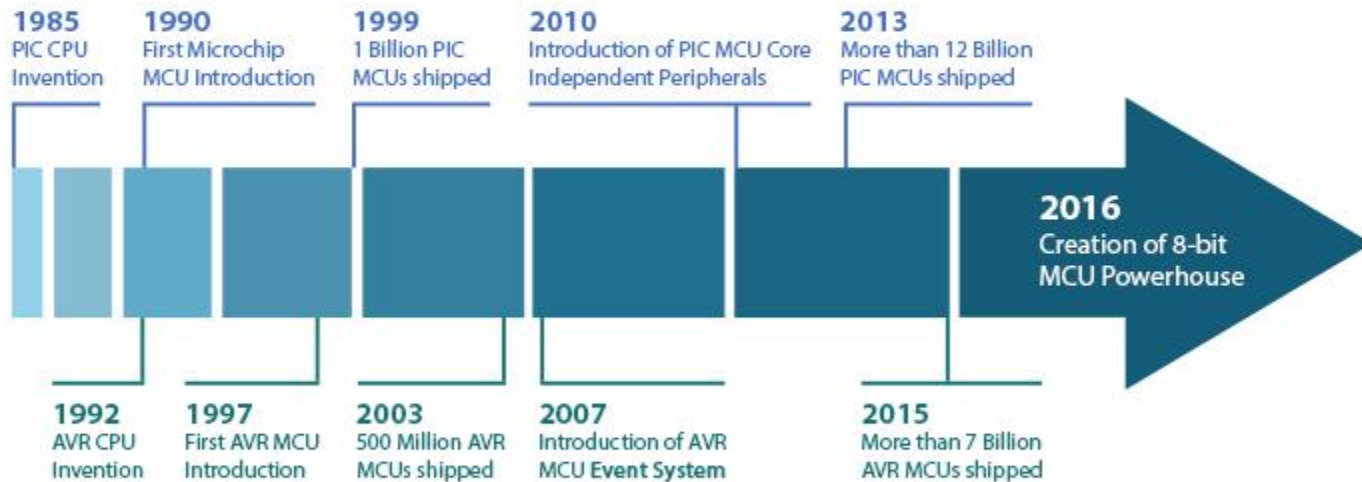
1 GHz performance ready to use for \$45



- ESP32
 - Desenvolvido para aplicações IoT e AIoT (Artificial intelligence of things) de baixo consumo.
 - Diversos modelos com diferentes configurações
 - CPU: Tensilica Xtensa LX6 Dual-Core 32 bit 240MHz
 - ROM: 448 Kbytes , RAM: 520 Kbytes, Flash: 4 MB
 - Wi-fi (802.11 b/g/n), RF (2.4 GHz) e Bluetooth (4.2) com antena
 - Faixa de temperatura: -40°C a +125°C
 - Ultra-Low Power
 - ADC e DAC



PIC[®] MCU



AVR[®] MCU

- Microcontrolador com arquitetura harvard e conjunto de instruções CISC
- Versão original lançada em 1980 pela Intel.
- Atualmente existem várias versões do original que são amplamente utilizadas em projeto de baixa complexidade devido ao seu baixo custo e alta disponibilidade.
- Algumas empresas oferecem o 8051 como IP Core para uso em projetos FPGA e ASICs.
- Custam na ordem de poucas dezenas de centavos de dólar



- Utilizam arquitetura Harvard de 8bits, desenvolvido pela Atmel e atualmente pertencente a Microchip. São processadores com relativo baixo poder de processamento. Possuem núcleo RISC.
- Bastante utilizado por hobbystas devido à facilidade de utilização e disponibilidade de kit de desenvolvimento de baixo custo.
- Possui as famílias
 - ATtiny
 - ATmega
 - ATxmega
- Utilizam armazenamento com memórias Flash, SRAM e EEPROM.
- São os microcontroladores utilizados no Arduino.



- Utilizam arquitetura Harvard de 8bits, 16 e 32 bits, fabricados pela Microchip.
- Bastante utilizado por hobbystas devido à facilidade de utilização e disponibilidade de kit de desenvolvimento de baixo custo.
- Ampla variedade de modelos com periféricos específicos como ethernet, CAN, LIN, RS-232, SPI, I²C com baixo custo.
- Utilizam armazenamento com memórias Flash, SRAM e EEPROM.
- Conjunto de instruções pequeno.
- Família dsPIC: Possui arquitetura híbrida, com poder computacional de um DSP mantendo a simplicidade de programação de um microcontrolador.
 - Otimizado para projetos de tempo real.
 - Projetado para operar em ambientes agressivos, com temperatura estendida (125°C) e alta temperatura (150°C). Úteis para o uso em conversores DC/DC, por exemplo. Também são utilizados em sensores instalados em escapamentos automotivos.



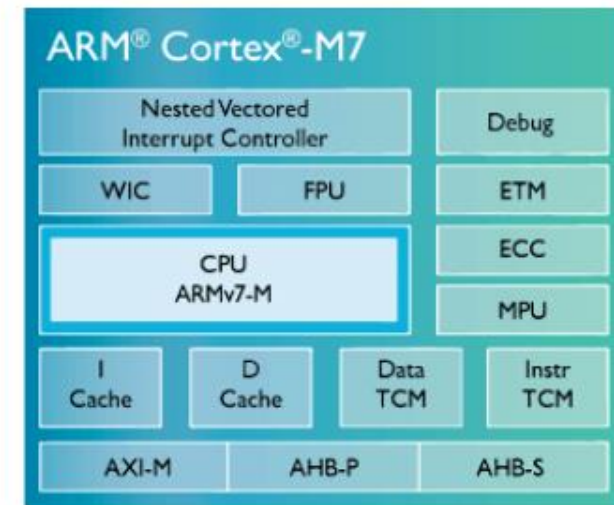
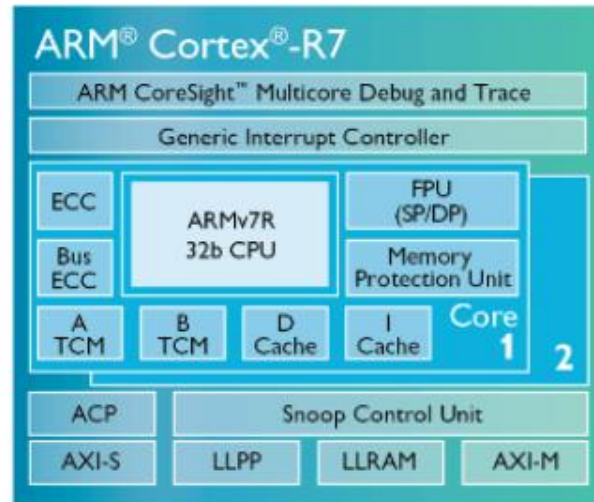
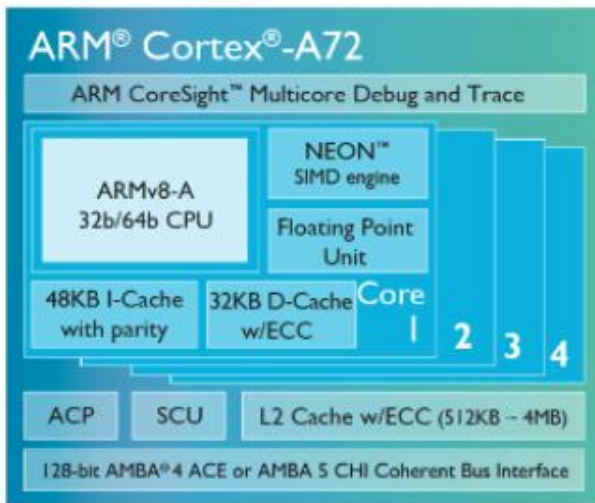
- Microcontrolador RP2040 ARM Cortex-M0+ Dual Core
- Clock 133 MHz
- Memória RAM: 256 KB
- Memória FLASH: 2 MB
- Comunicação: 2×SPI, 2×I2C, 2×UART, 3×12-bit ADC, 16×canais PWM controláveis
- GPIO de 40 pinos (26 pinos multifuncionais)
- RTC (Real Time Counter) de alta precisão
- Sensor de temperatura on-board
- Preço: R\$50,00



- Categorias:
 - Sistemas embarcados de tempo real (R): sistemas para aplicações de armazenamento, automotivas, industriais e de rede.
 - Utilizados em aplicações Hard Real-Time.
 - Plataformas de aplicação (A): dispositivos executando sistemas operacionais abertos.
 - Microcontroladores (M): Sistemas embarcados e aplicações que necessitem soft ou firm Real-Time. Máquinas de autoatendimento, máquinas de lavar roupas...

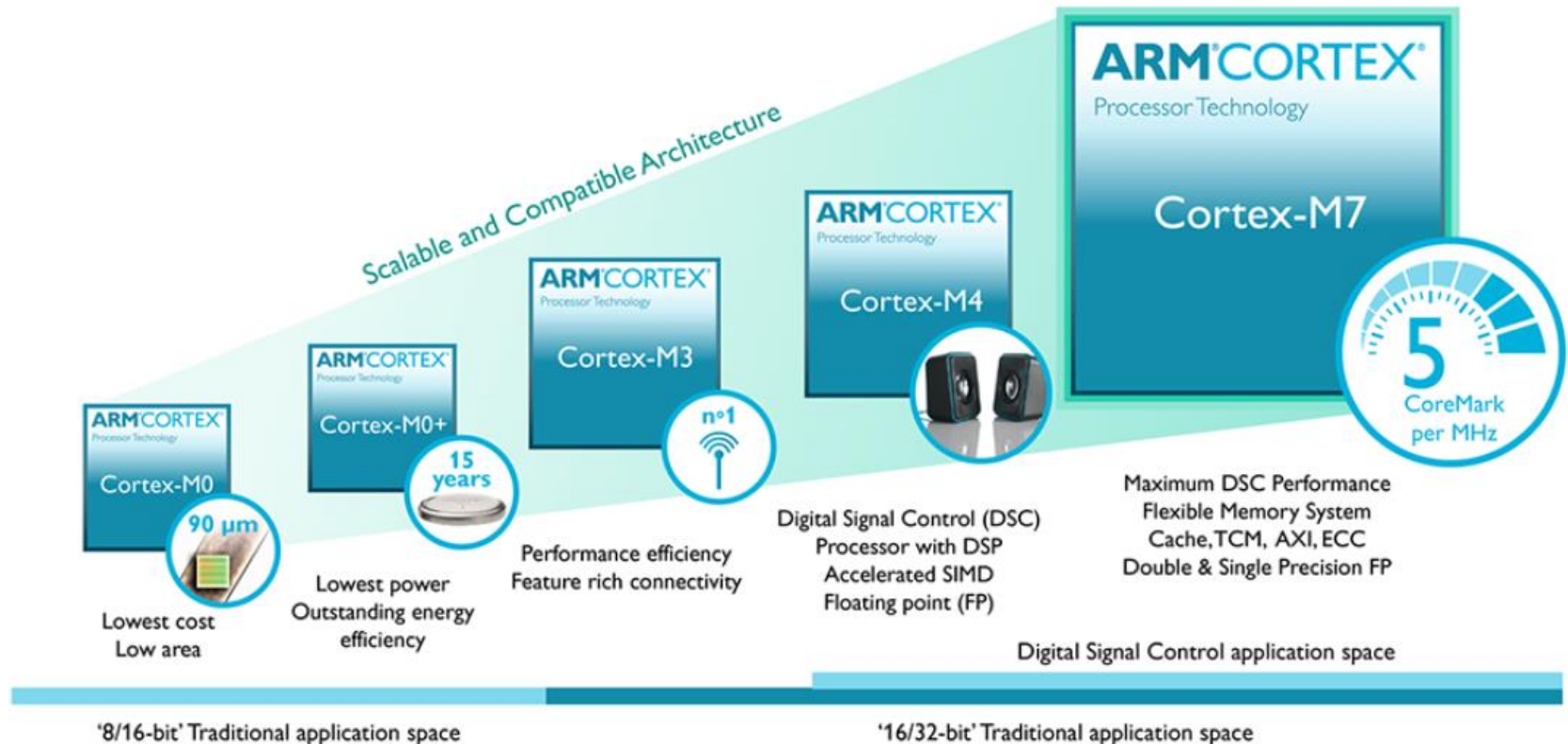
Microcontroladores ARM

- Qualcomm Snapdragon
- Nvidia Tegra



TCM (*Tightly-Coupled Memory*) – Pequenas memórias localizadas bem próximas ao processador, de forma que podem ser acessadas em cada ciclo de clock. Utilizadas para guardar informações e dados críticos.

Microcontroladores ARM



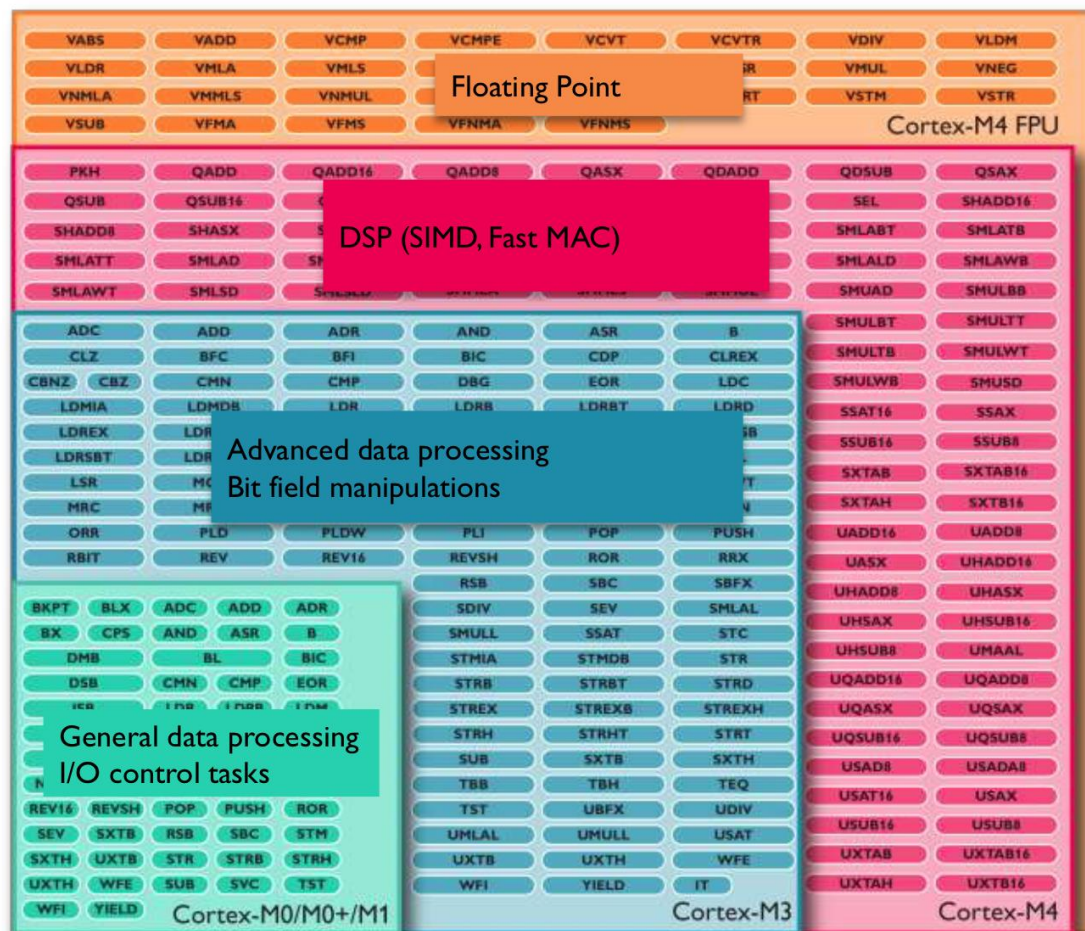
Fonte: https://hexus.net/static/arm_cortex_m/

ARM Cortex-M Instruction Set Architecture

Cortex-M4

Cortex-M3

Cortex-M0/M0+



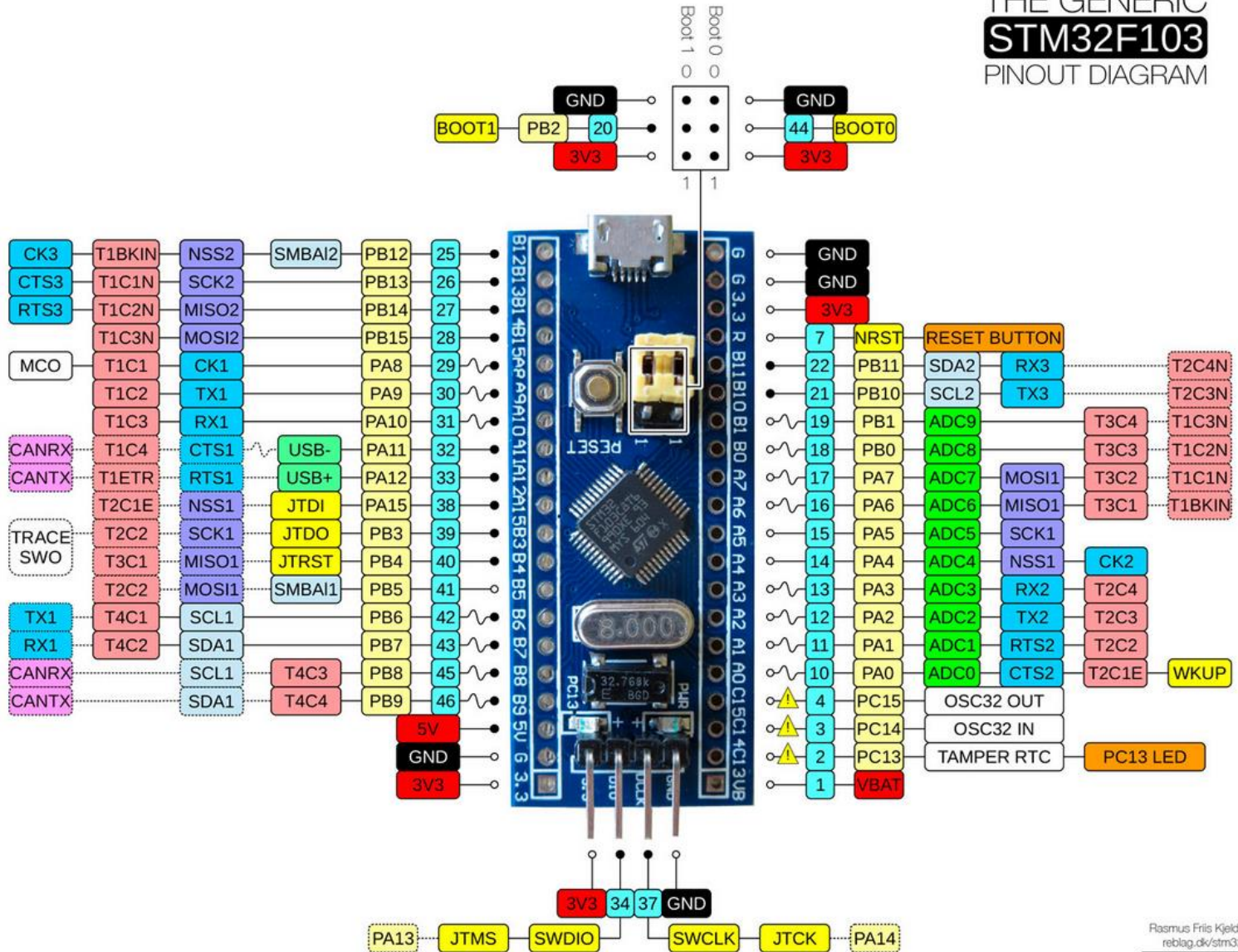
<https://www.anandtech.com/show/8400/arms-cortex-m-even-smaller-and-lower-power-cpu-cores>

Microcontroladores ARM

LEGEND

POWER
GROUND
PHYSICAL PIN
PIN NAME
CONTROL
ANALOG
TIMER & CHANNEL
USART
SPI
I2C
CAN BUS
USB
MISC
BOARD HARDWARE
● 5V tolerant
○ Not 5V tolerant
~ PWM pin
⋯ Alternate function
⚠ PC13, PC14, PC15: Sink max 3mA, source 0mA, max 2mhz, max 30pF
Absolute MAX 150mA total source/sink for entire CPU
Max ±20mA per pin, ±8mA recommended

THE GENERIC STM32F103 PINOUT DIAGRAM



https://erc-bpgc.github.io/handbook/electronics/Development_Boards/STM32/

Naming Convention of STM microcontrollers

Parameter	Meaning
STM	name of the manufacturer (STMicroelectronics)
32	32 bit ARM architecture
F	Foundation
1	Core (ARM Cortex M3)
03	Line (describes peripherals and speed)
C	48 pins
8	64 KB flash memory
T	LQFP package (Low Profile Quad Flat Pack)
6	Operating Temperature Range (-40 °C to 85 °C)

Parameter	Meaning
Architecture	32 bit ARM Cortex M3
Operating Voltage	2.7V to 3.6V
CPU Frequency	72 MHz
Number of GPIO pins	37
Number of PWM pins	12
Analog Input Pins	10 (12 bit resolution)
I2C Peripherals	2
SPI Peripherals	2
CAN 2.0 Peripheral	1
Timers	3(16-bit), 1
Flash Memory	64KB
RAM	20kB

1 MSA/s

https://erc-bpgc.github.io/handbook/electronics/Development_Boards/STM32/

- Família M4
- Periféricos:
 - GPIO (General Purpose IN/OUT): São as conexões de entrada e saída do microcontrolador. Através deles é possível ler o estado de sensores externos (entrada), e acionar relés, LEDs, telas (saída).
 - Tensões de saída: 3,3V. (Série ULN)
 - Tensões de entrada: 3,3V.
 - Para tratamento de prioridade, todos os pinos podem funcionar com interrupção.
 - Recursos como: Schmitt Trigger, Pull Up, Pull Down, modo coletor aberto e controle de corrente de saída a 2, 4, 6, 8, 10 ou 12mA.

Performance



**Baixo
Consumo**



Baixo Custo



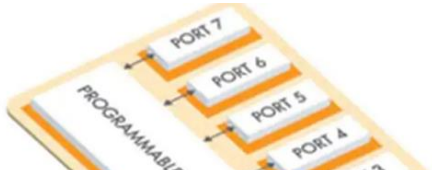
**Determinismo:
garantia de
execução
críticas (Real
Time)**

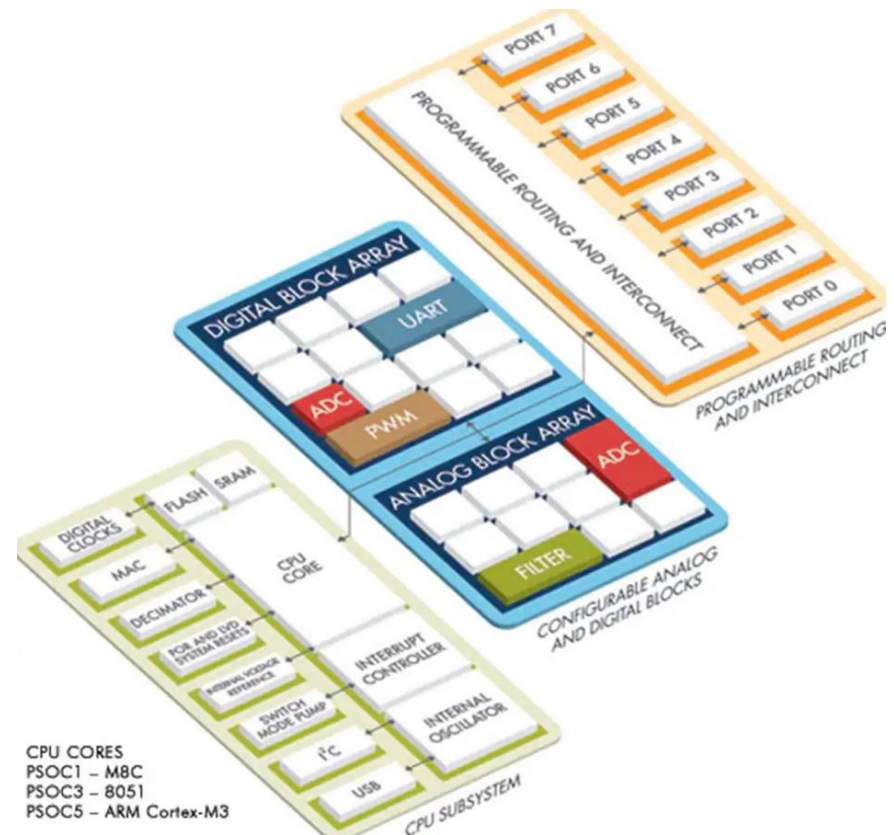
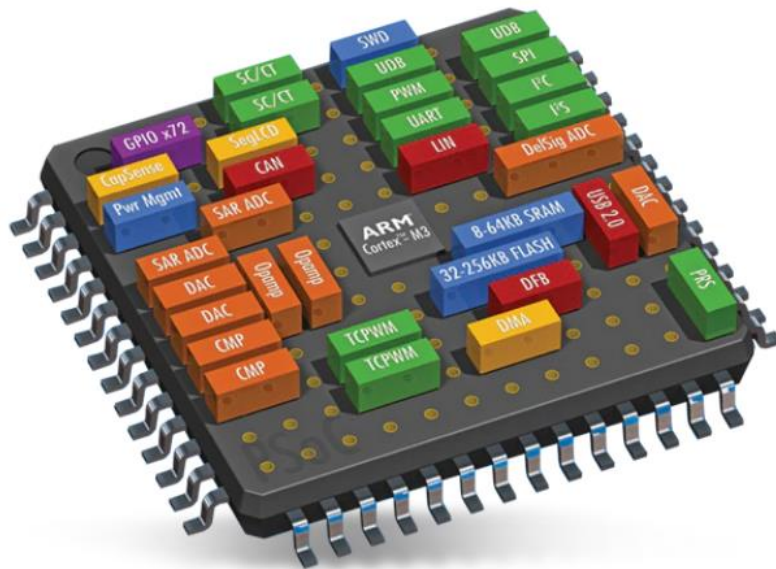


**Fácil de Usar
Prog. C
Ferramentas
Debug**

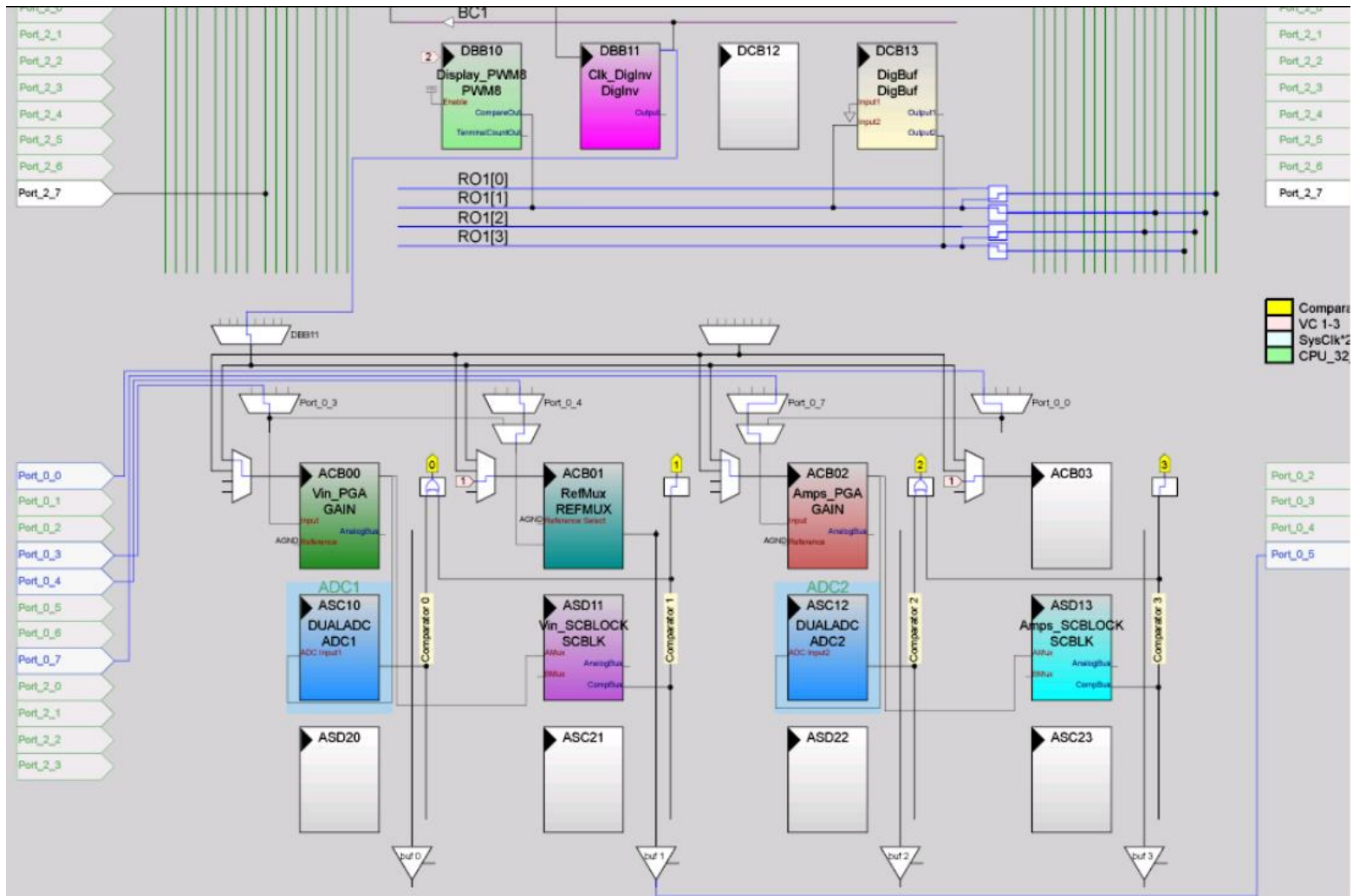


PSoC – Programmable System-on-a-chip

- Fabricado pela Cypress
 - É a “mistura” entre um microcontrolador padrão + módulos analógicos e digitais reconfiguráveis
 - Os periféricos não possuem número fixo, você pode alocá-los de acordo com os blocos analógicos e digitais disponíveis
 - Núcleos: 8051, ARM M0, M0+, M3, M4...
- 
- Diagrama de um módulo reconfigurável, mostrando uma série de portas programáveis (PORT 4, PORT 5, PORT 6, PORT 7) e um bloco rotulado 'PROGRAMMABLE'.



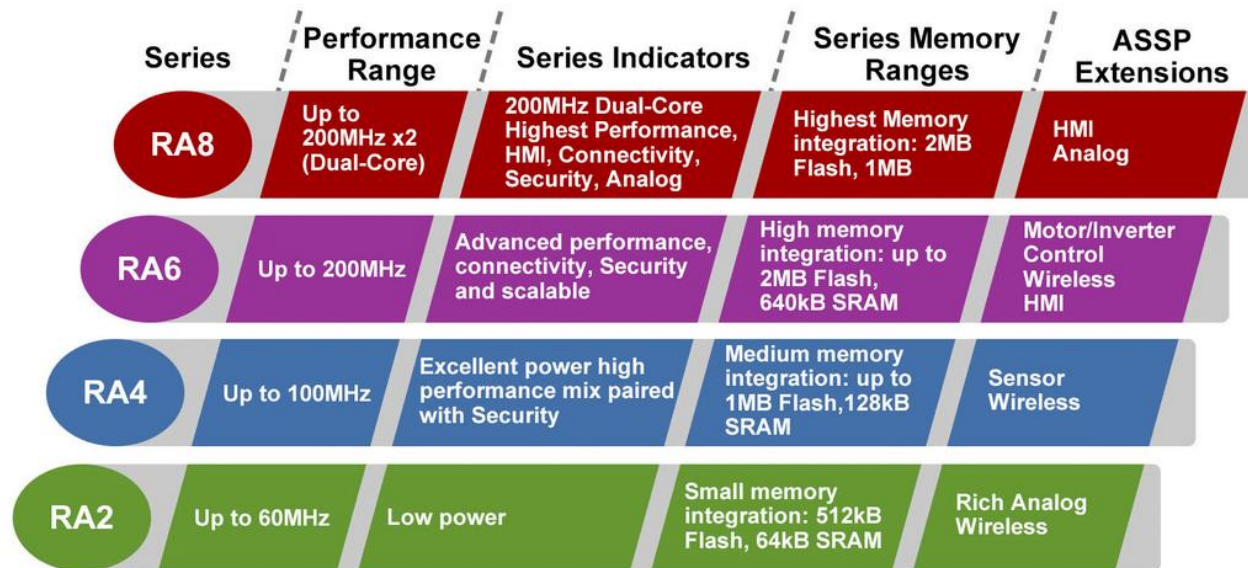
- Ambiente de desenvolvimento



- Exemplos de módulos PSoC:
 - Contadores (8,16,24 e 32 bits)
 - Verificador de redundância cíclica (CRC16)
 - Módulo I2C (mestre ou escravo)
 - Módulos de recepção e transmissão infravermelho
 - Módulos UART
 - Módulos SPI
 - Buffers
 - Módulo para conexão de display LCD
 - Módulo para conexão de display sete segmentos
 - Módulo para conexão de LEDs
 - Módulos de baixo consumo de energia

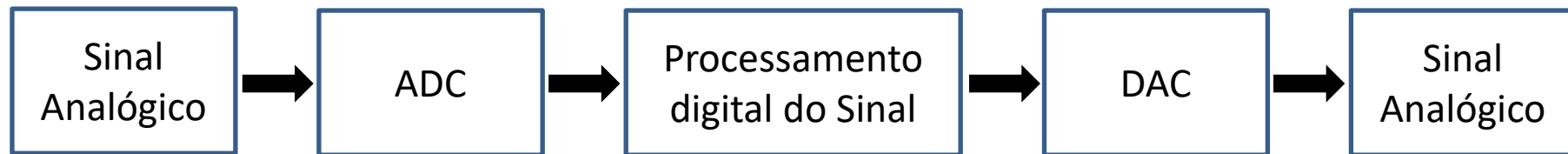
- Exemplos de módulos PSoC:
 - Módulo PWM (8 ou 16 bits, com ou sem tempo morto)
 - Pseudo Sequenciador Randômico de 8, 16, 24 ou 32 bits
 - Temporizadores de 8, 16, 24 ou 32 bits
 - Amplificadores com ganho programável
 - Filtros de 2 pólos
 - Multiplexadores
 - Conversores D/A (múltiplas configurações)
 - Conversores A/D (múltiplas configurações)
 - Conversor Boost (para alimentações de baixa tensão)

- Amplamente utilizado na indústria
- Lançada uma nova família, chamada de Renesas Advanced (RA), utiliza núcleo ARM Cortex-M de 32 bits.
 - Focado no mercado de IoT, com tecnologias integradas de segurança, proteção, conectividade e IHM.
 - Possui certificação PSA (certificação de segurança para hardware, software e dispositivos da Internet das Coisas)



<https://www.renesas.com/us/en/products/microcontrollers-microprocessors/ra-cortex-m-mcus>

Processadores Digitais de Sinais (DSP)



- É a ciência que estuda as regras que governam o comportamento de sinais discretos, em conjunto com os dispositivos que os processam.
- Surge como uma alternativa ao processamento analógico de sinais, apresentando diversas vantagens.
- Para que o sinal seja processado digitalmente, necessita ser, primeiramente, discretizado.

- **Offline**

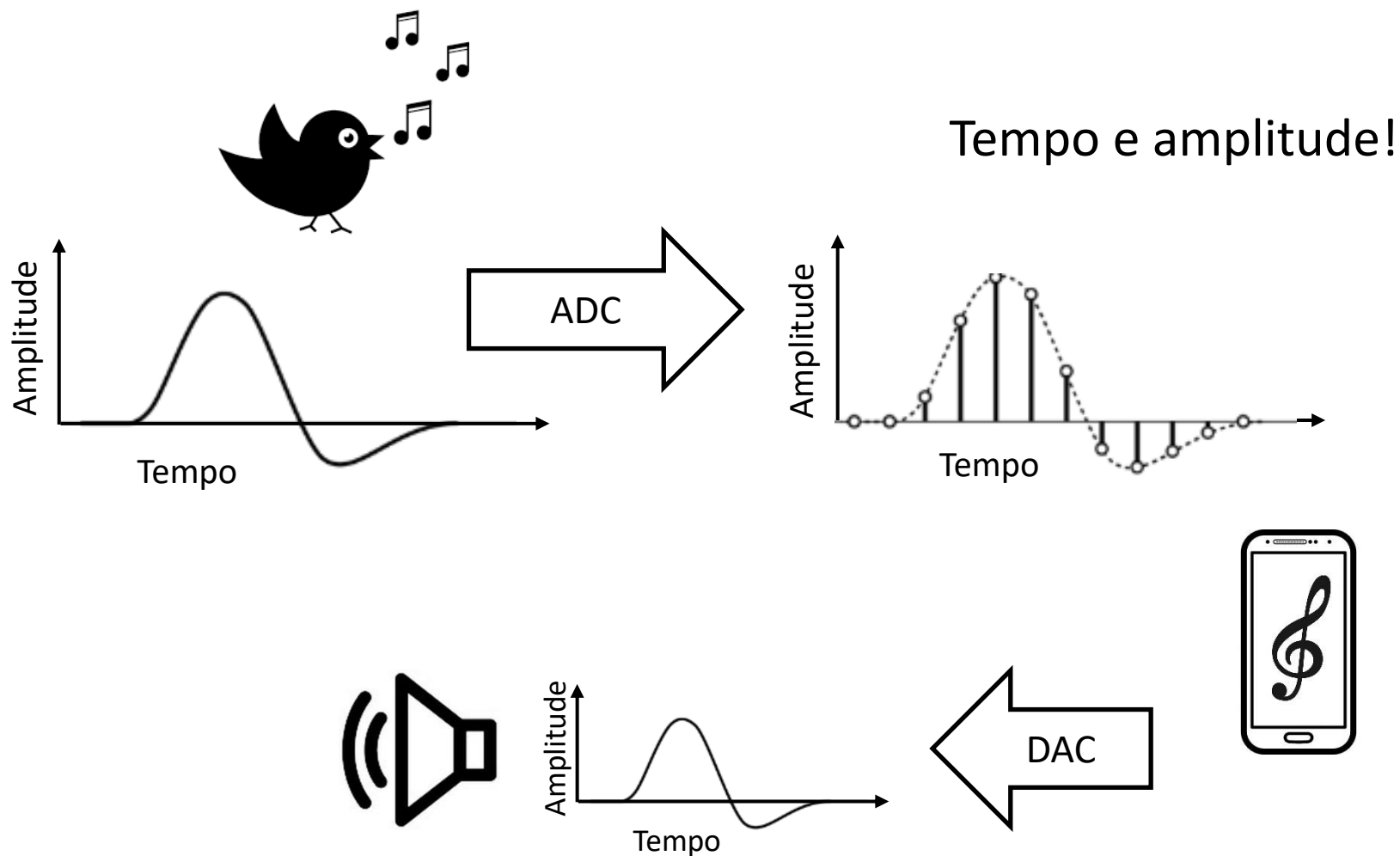
- Pós-processamento – Os dados já estão armazenados.
- Não existe restrição de tempo.

- **Online**

- Os dados são repassados ao processador, mas o mesmo não precisa terminar o processamento antes que um novo dado chegue.
- Necessidade de memória (buffer).

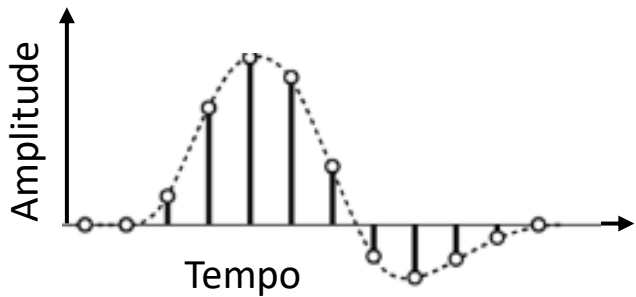
- **Real Time**

- Tempo de processamento crítico.
- O processamento de um dado DEVE terminar antes que um novo chegue.



- Qual a função do DSP nessa situação?

Processamento Digital de Sinal: Exemplo



Atuação do DSP codificando o sinal em MP3 e salvando-o na memória

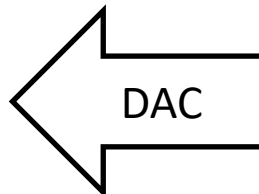
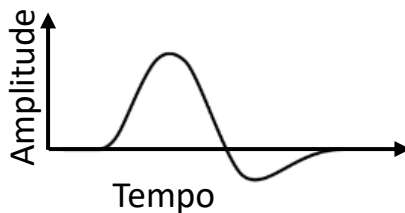
Mais alguma outra utilidade para o DSP?

Outras funções:

- Controle de volume;
- Equalização;
- Efeitos sonoros;
- Etc.



Mais alguma utilidade para o DSP?



O sinal codificado em MP3 é decodificado pelo DSP para então ser convertido pelo DAC

- Filtros Digitais!
 - Totalmente programáveis: Um filtro passa-baixa pode se “transformar” em um filtro passa alta sem a troca de hardware.
- Redução de custos: Reduzem os requisitos de hardware (devido a programabilidade).
- Reduzem o tempo de desenvolvimento (ferramentas de desenvolvimento, suporte de projeto).
- Implementação de algoritmos adaptativos:
 - Controle por filtro adaptativo (cancelamento de ruído).
- Funções especiais:
 - Compressão de dados;
 - Codificação de dados.

- Hardwares que podem trabalhar como DSP:
 - Microprocessadores;
 - Microcontroladores;
 - FPGAs
 - DSPs

- Microprocessadores
 - De uso geral.
 - Significativo gerenciamento de memória.
 - Uma operação por vez.
 - Otimizados para grandes aplicações.

- **FPGA**
 - São extremamente rápidas, uma vez que o circuito necessário é “construído” especificamente para a sua função.
 - Programação relativamente simples para aplicações simples....No entanto, torna-se bastante complexa para aplicações maiores.
 - **Pode realizar várias operações em paralelo.**

- Microcontroladores
 - Dedicados a uma única aplicação.
 - Possui uma unidade lógica e aritmética simples, que realiza todo o processamento.
 - Geralmente não possui dispositivos internos de otimização.
 - **dsPIC.**

- DSPs
 - Realizam funções matemáticas de alto nível.
 - Realizam múltiplas operações por ciclo.
 - Adequados para aplicações mais complexas, com requisitos matemáticos em tempo real.
 - Possuem diversos dispositivos internos de otimização do processamento.

- Conclusões
 - DSPs permitem o desenvolvimento de sistemas em tempo real de maneira fácil e segura.
 - Alguns DSPs já apresentam tamanha performance que já podem ser utilizados em projeto que eram dominados por FPGAs.
 - O crescimento do uso tem favorecido a queda nos preços, bem como o número de opções disponíveis.

- Levantamento de requisitos de software e de hardware

- Material de aula Prof. Hugo Vieira Neto
- Material de aula prof. Cesar Ofuchi
http://paginapessoal.utfpr.edu.br/ofuchi/sistemas-embarcados-el68e/aula-2-arquitetura-arm-cortex-m3/1_arquitetura_cortex_M3.pdf/view
- TOCCI, Ronald J.; WIDMER, Neal S.; MOSS, Gregory L. Sistemas digitais : princípios e aplicações. 11. ed. São Paulo: Pearson Prentice Hall, 2011. ISBN 978-85-7605-922-6
- “Introduction to ARM Cortex M Microcontrollers”. Jonathan Valvano. 5ª Edição, 2014.
- DSPs [http://www.cbpf.br/cat/pdsi/downloads/DSPs E Suas Aplicacoes Em DSP.pdf](http://www.cbpf.br/cat/pdsi/downloads/DSPs_E_Suas_Aplicacoes_Em_DSP.pdf)