

ARQUITETURA

⇒ Microprocessadores

↳ é o agente capaz de executar algumas instruções.

entrada → microprocessador → resposta

⊕ Microprocessador é mais caro que um microprocessador.

• O MP só entende 0 e 1.

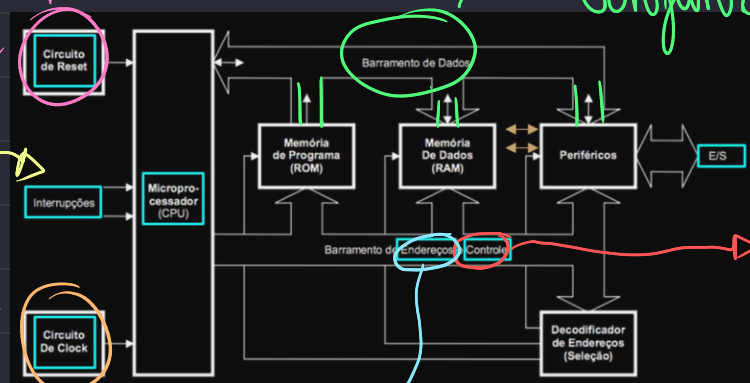
↳ O MP é capaz de:

- ler instruções da ROM.
- armazenar informações temporárias na RAM.
- lidar com o barramento de dados em instruções.
- se comunicar com periféricos.

read-only

Resetar o equipamento

Instruções de urgência
Quando executar as instruções



Conjunto de fios

Qual memória será acessada.

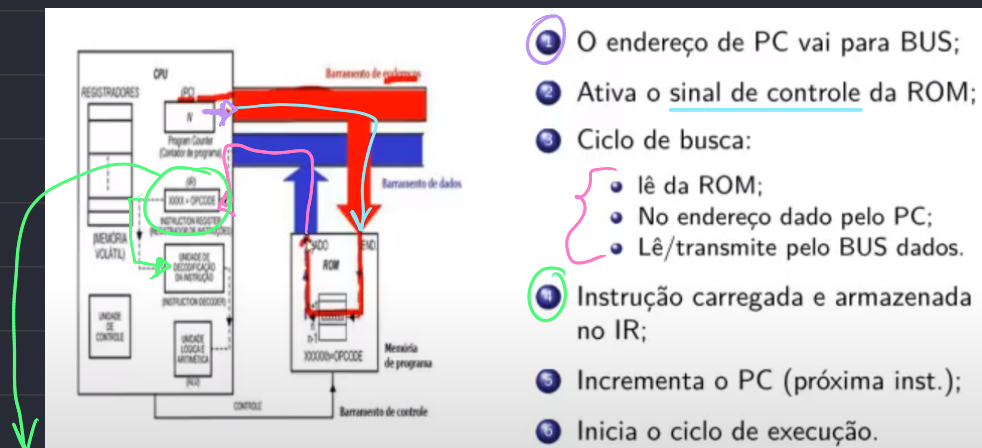
Informa o endereço de certa instrução.

⇒ Funções da CPU

- buscar informações continuamente na ROM.
- executar instruções, funções lógicas e aritméticas.
- transferência de dados.
- tomada de decisões.

⊕ Registradores são memórias internas de um MP, super rápidas.

⊕ Program Counter (PC) mostra qual a próxima instrução.



Quais bits delem
ou armazenam

⇒ Microcontroladores

- ↳ apresenta componentes que o MP não tem.
- ↳ MP tem tudo que o MC tem, porém é mais rápido.

⇒ Arquitetura ARM Cortex M4

* Dados ≠ programa. → Arquitetura Harvard

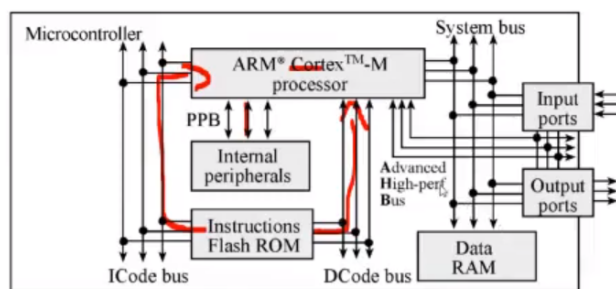
⇒ É uma linguagem de máquina.

A → + rápidos.

R → regl-timo, intermediários.

M → + lentos.

* Bus é um conjunto de fios.



• Barramentos, permitindo execuções em paralelo:

- ICode: busca opcodes da ROM;
- DCode: lê dados constantes da ROM;
- System: lê/escreve dados da RAM ou I/O, busca opcode da RAM;
- PPB: lê/escreve dados de periféricos internos (como interrupções);
- AHB: lê/escreve dados de portas I/O de alta velocidade.

↳ performance

* Cada MC tem um funcionamento diferente.

↳ Registradores R0 a R15:

- 13 de uso geral

↗ 32 bits

R13: Main Stack Pointer (MSP)

R14: Link Register (LR)

R15: Program Counter (PC)

↳ próxima instrução.

PSR: Program Status Register

⚡ Informações sobre o que ocorreu na operação.

⊕ Para o nosso MC:

1024k Flash ROM	0x0000.0000
	0x000F.FFFF
	0x2000.0000
256k RAM	0x2003.FFFF
	0x4000.0000
I/O	0x400F.FFFF
	0xE000.0000
I/O PPB Interno	0xE004.1FFF

⇒ CISC

- conjunto de instruções cada vez mais complicado, para se aproximar de Assembly.

⇒ RISC

- instruções executam mais rapidamente.