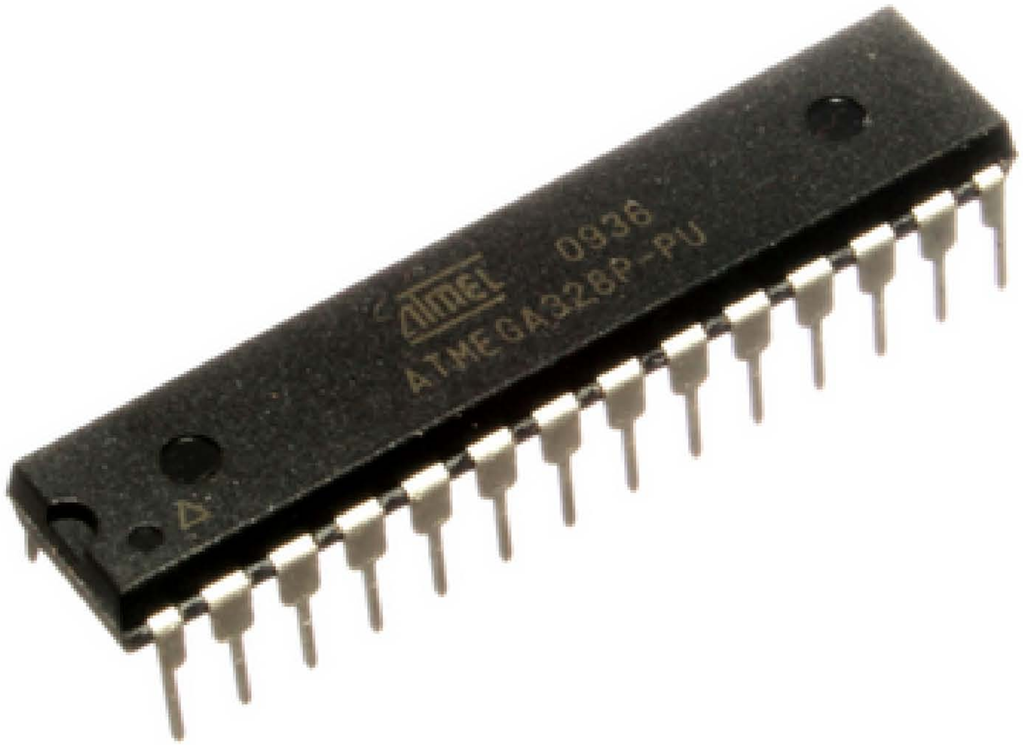
AVR ATMEGA328P



© Prof. Engº esp Luiz Antonio Vargas Pinto
www.vargasp.com
© setembro 2023

Histórico

- A arquitetura básica foi desenvolvida na Noruega pelos doutorandos Alf-Egil-Boden e Vegard Wollan da *University of Science and Technology* em **1992**
- Sugere-se, porém nunca foi comprovado, que AVR é o acrônimo de **A**lf, **V**egard e **R**ISC
- Após algum tempo de aperfeiçoamento, eles venderam o projeto para a ATMEL
- O primeiro AVR, o AT90S1200, foi lançado em **1997**.
- A ATMEL e a tecnologia AVR foi comprada pela Microchip em **2016**

© Prof. Vargas

Algumas considerações

- Aplicações
  - Câmeras, Aparelhos de DVD, Telefones celulares, Microondas, Máquinas de lavar roupa, etc . . .

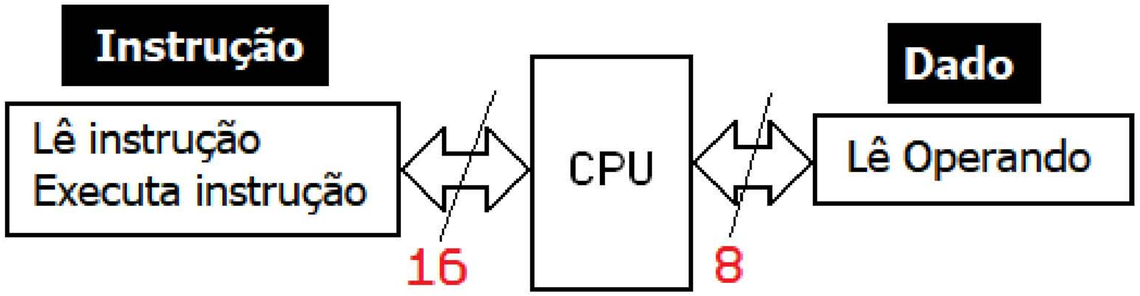
- MicroProcessador x MicroControlador
  - Microprocessadores tem periféricos externos ao chip
  - Microcontroladores tem os periféricos internos ao chip
    - muitas vezes ele é citado como:

      Computador em um chip

© Prof. Vargas

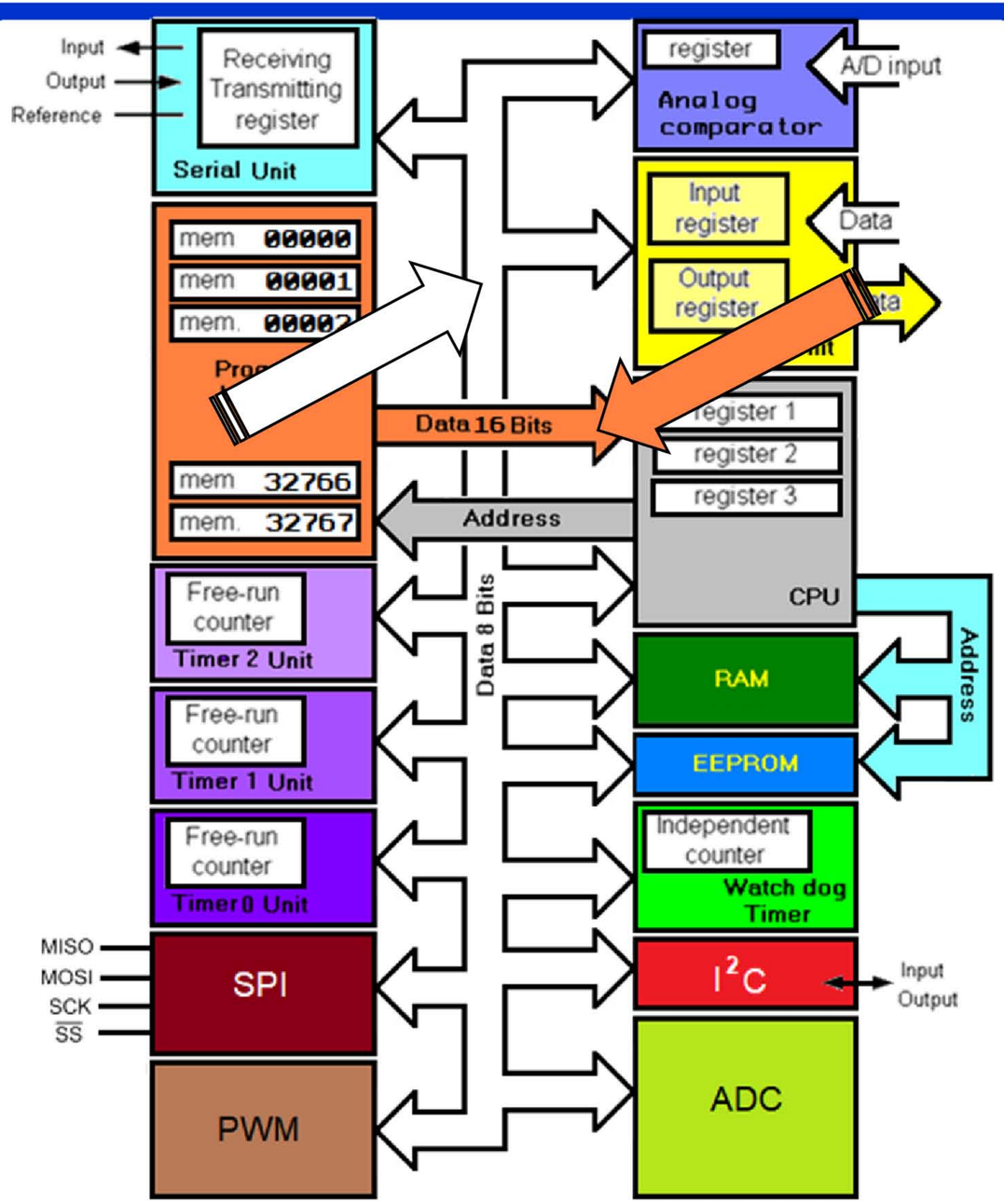
Arquitetura Harvard

- Data bus independe do instruction bus
  - Permite que a CPU acesse aos dois bus simultaneamente
    - Dados e instruções simultaneamente
  - Melhora o desempenho
- Memória e dados tem tamanhos diferentes
  - AVR Atmega328p: instruções 16 bits / dados 8 bits



© Prof. Vargas

Detalhe



© Prof. Vargas

RISC

- As instruções são maiores e portanto mais poderosas (16 bits)
- instruções são mais eficientes, mesmo a 1 μs (embora necessite de um cristal de 16 MHz)

© Prof. Vargas

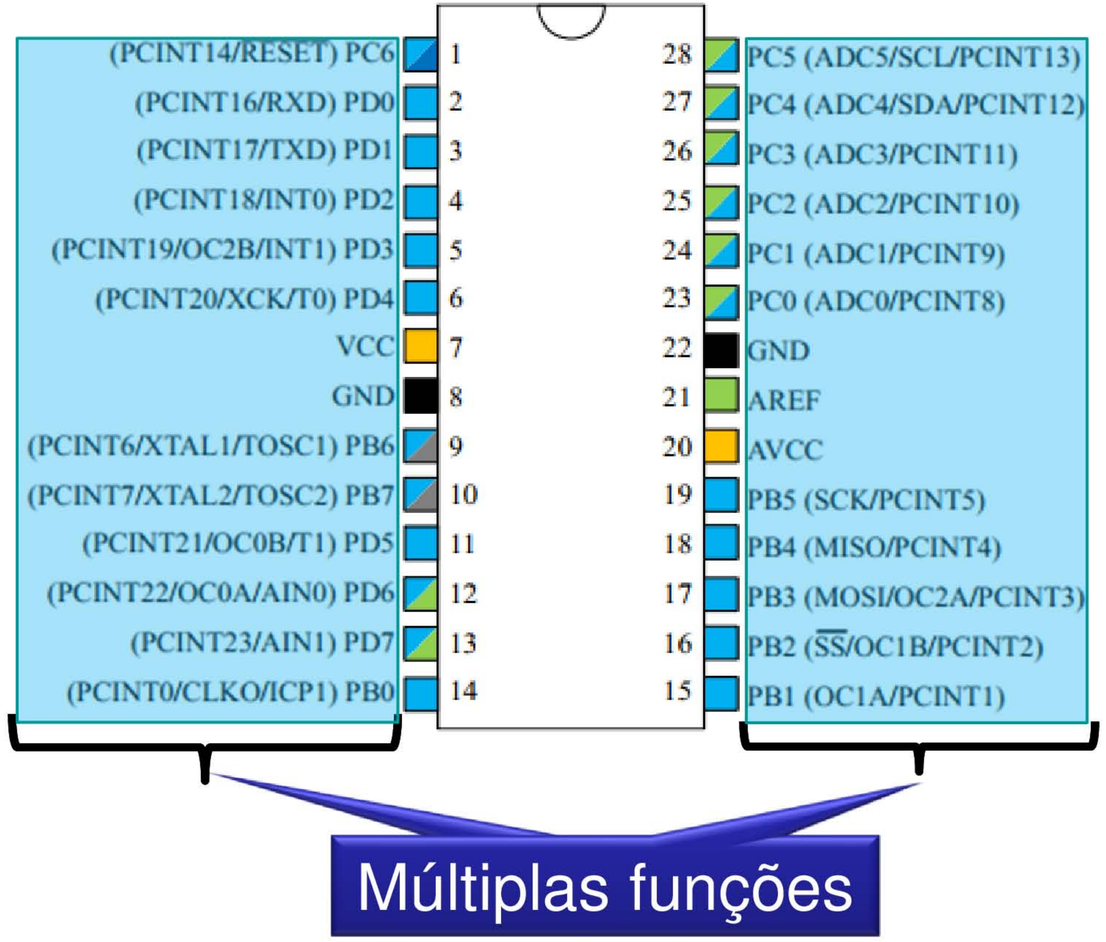
O ATMEGA328p

- 32768 endereços de programa em FlashROM
- 2048 bytes de memória SRAM
- 1024 bytes de memória EEPROM
- 1 Canal serial USART
- 2 Canais SPI
- 1 Canal I^2C
- 3 Timers programáveis por software
- 1 Timer Watchdog
- 23 ports de I/O (PB, PC e PD)
- 1 Comparador Analógico
- 6 Canais de PWM
- 8 Conversores A/D 10 bits
- 131 instruções

© Prof. Vargas

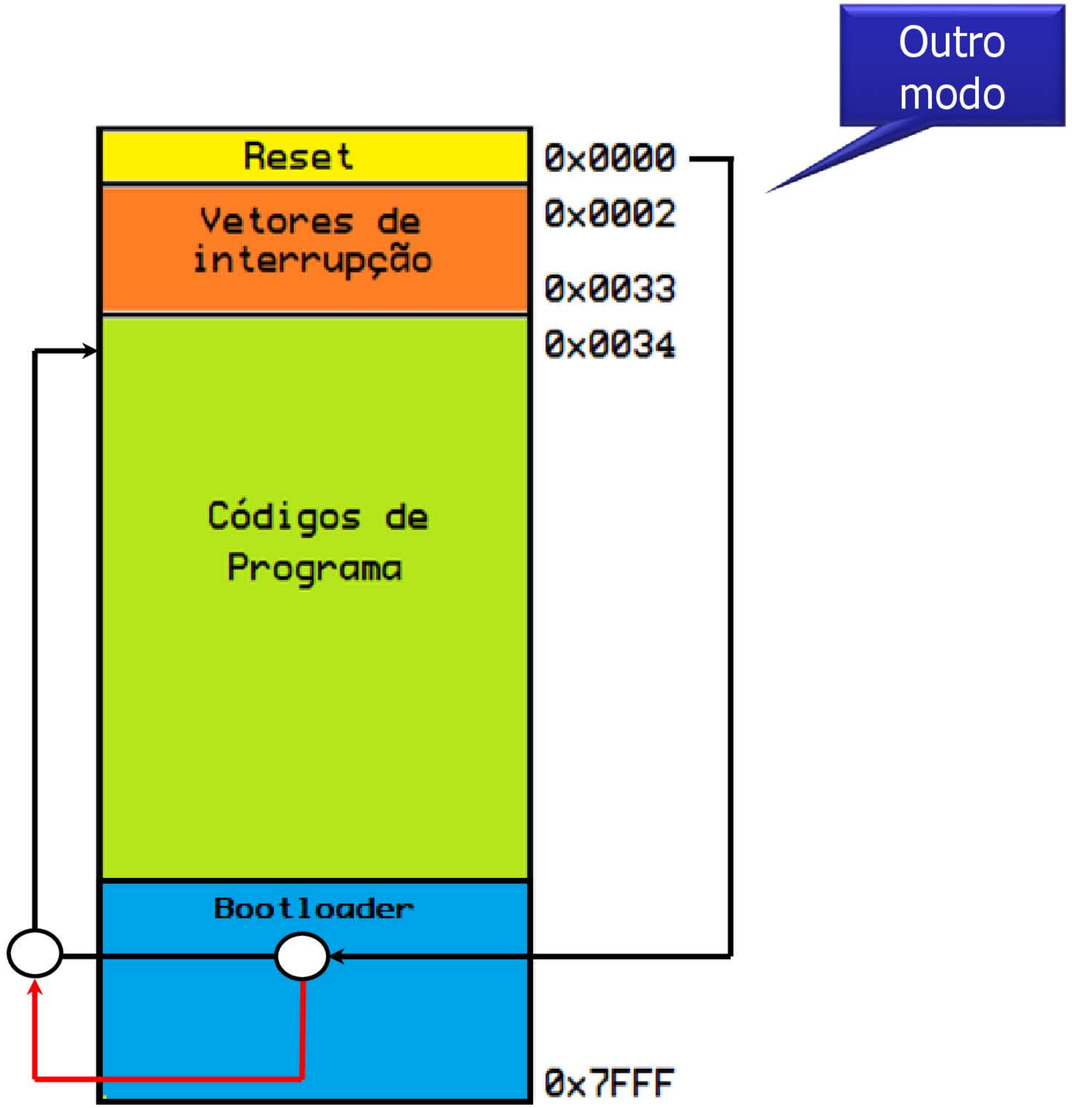
Overview

- Funciona de 1.8 a 5.5V
- Flashrom pode ser gravada pelo menos 10.000 vezes
- EEPROM permite pelo menos 100.000 gravações
- Dados retidos por 20 anos a 85ºC ou 100 anos a 25ºC



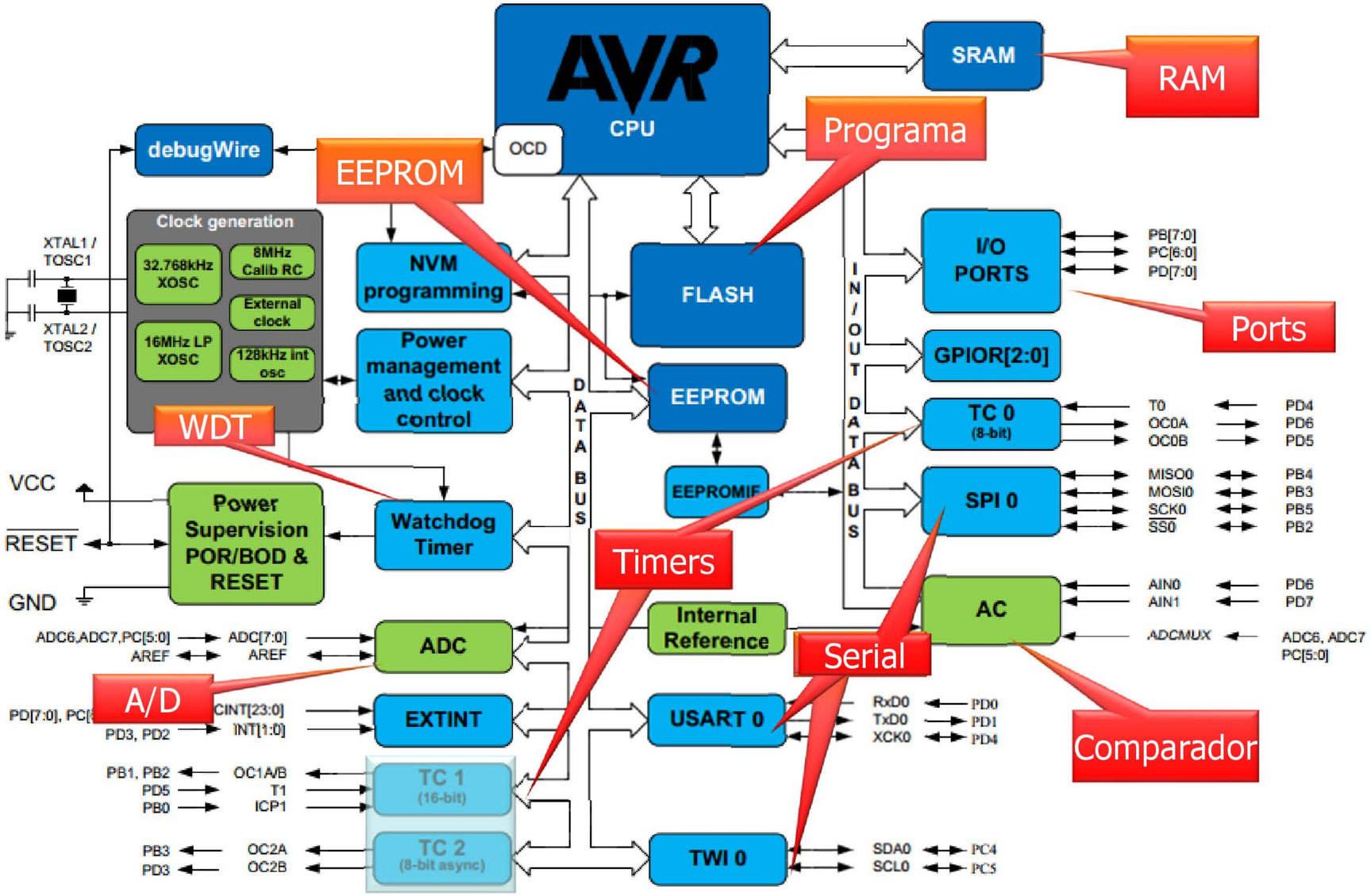
© Prof. Vargas

Estrutura da FlashRom



© Prof. Vargas

Arquitetura

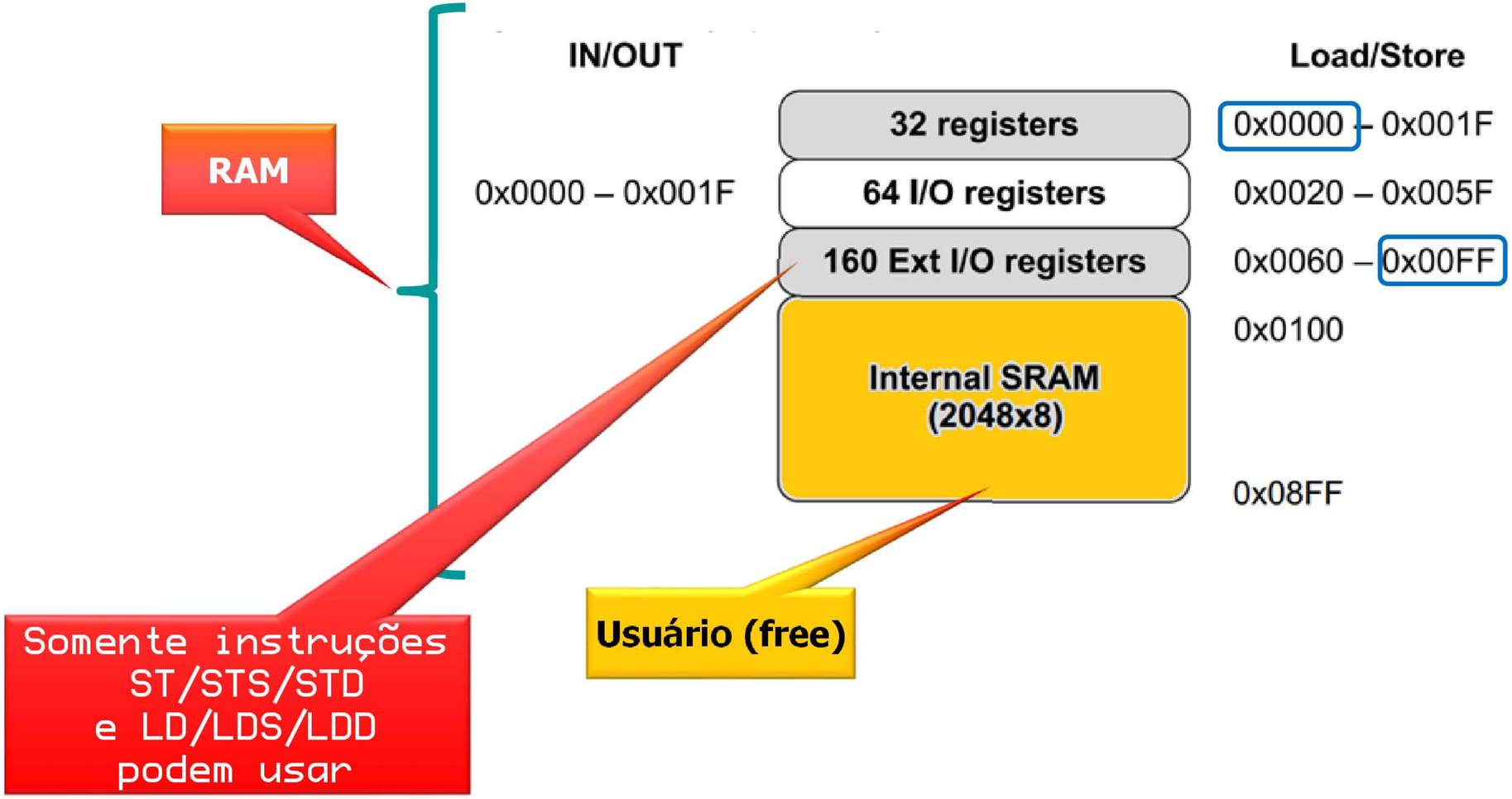


© Prof. Vargas

Arquitetura de memória

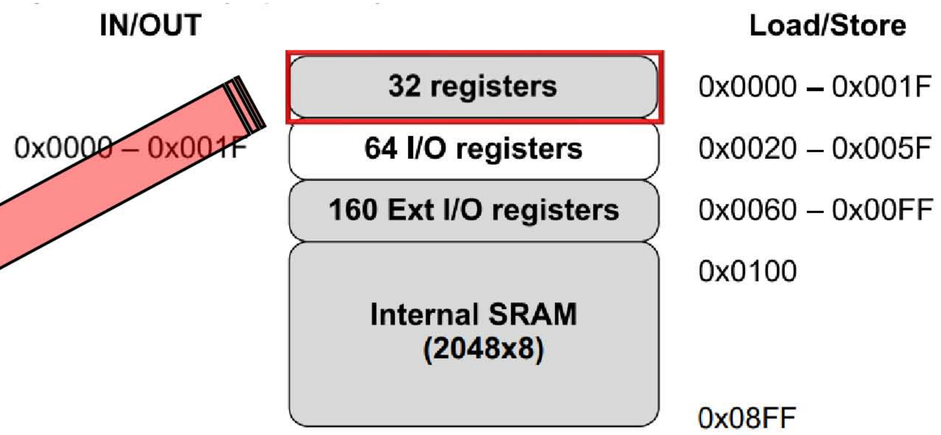
© Prof. Vargas

Memória de dados e registros de controle



© Prof. Vargas

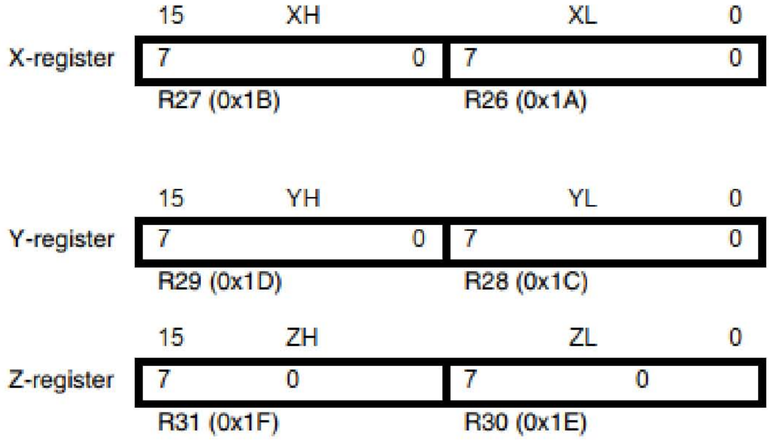
Os registradores



AVR CPU General Purpose Working Registers
Figure 7-2.

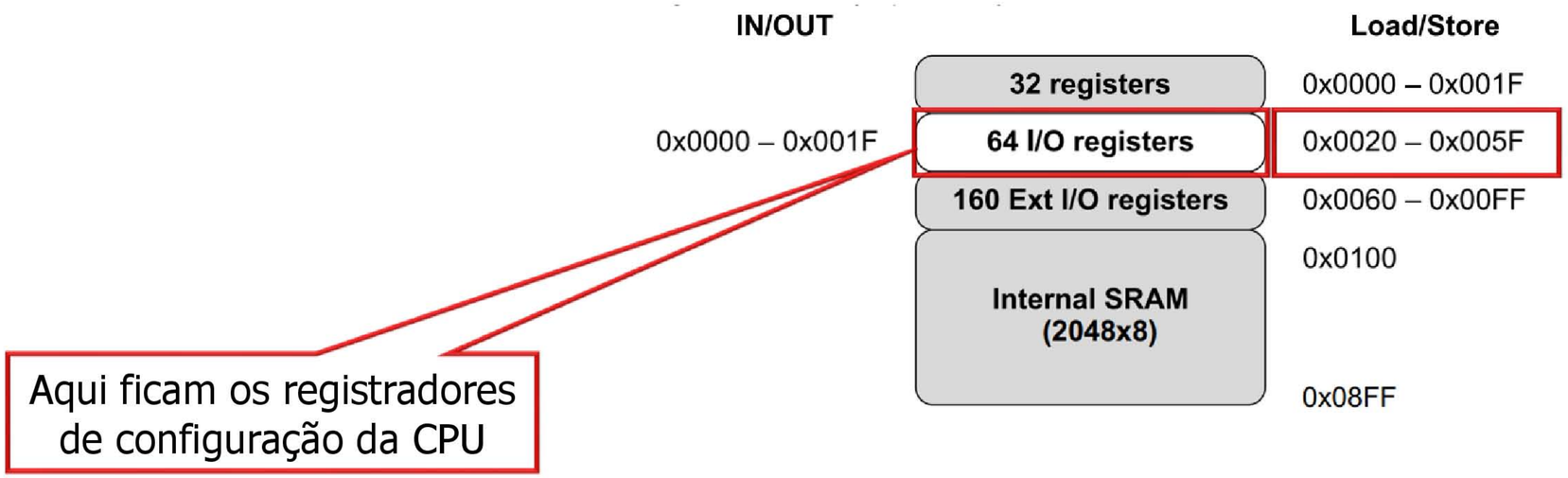
7 … 0	Addr.	
R0	0x00	
R1	0x01	
R2	0x02	
…		
R13	0x0D	
R14	0x0E	
R15	0x0F	
R16	0x10	
R17	0x11	
R26	0x1A	X-register Low Byte
R27	0x1B	X-register High Byte
R28	0x1C	Y-register Low Byte
R29	0x1D	Y-register High Byte
R30	0x1E	Z-register Low Byte
R31	0x1F	Z-register High Byte

The X-, Y-, and Z-registers
Figure 7-3



© Prof. Vargas

64 I/O Registers



© Prof. Vargas

Status REGister – SREG

© Prof. Vargas

Bit	7	6	5	4	3	2	1	0
0x3F (0x5F)	I	T	H	S	V	N	Z	C
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- Bit 7: I Global Interrupt Enable
  - 1 = Habilita 0 = Desabilita
  - Se: 0 ocorreu interrupção
    1 Sem interrupção
  - RETI põe I = 1
  - As instruções SEI e CLI afetam o bit I.
- Bit 6: T Copy Storage
  - As instruções BLD (Bit LoaD) e BST (Bit STore) usam o bit T em operações de bit.
- Bit 5: H Half Carry Flag
  - Usado em operações matemáticas - é o bit excedente

© Prof. Vargas

Bit	7	6	5	4	3	2	1	0
0x3F (0x5F)	I	T	H	S	V	N	Z	C
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

- **Bit 4: S Sign**
  - O bit S é sempre um exclusive OR entre o flag N e o complemento de 2 do bit V
- **Bit 3: V Overflow do complemento de 2**
  - Ele suporta aritmética em complemento de 2
- **Bit 2: N Negative Flag**
  - Indica resultado negativo em uma operação lógica ou aritmética
- **Bit 1: Z Zero Flag**
  - Indica ocorrência de Zero em alguma operação
- **Bit 0: C Carry Flag**
  - Usado em operações matemáticas – é o bit excedente

© Prof. Vargas

MCU Status REGister

© Prof. Vargas

MCU Status Register

Bit	7	6	5	4	3	2	1	0
					WDRF	BORF	EXTRF	PORF
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

- Bit 3: WDRF Watchdog System Reset Flag
  - 1 = Ocorreu Watchdog por falha de CPU
  - 0 = Ocorreu Power-On Reset ou escrita de 0
- Bit 2: BORF Brown-Out Reset Flag
  - 1 = Ocorreu Brown-Out
  - 0 = Ocorreu Power-On Reset ou escrita de 0
- Bit 1: EXTRF EXternal Reset Flag
  - 1 = Ocorreu
  - 0 = Não ocorreu ou escrita de 0
- Bit 0: PORF Power-On Reset Flag
  - 1 = Ocorreu Power-On Reset
  - 0 = Somente por escrita de 0

© Prof. Vargas

Estudo das instruções

AVR ATMEGA328P

© Prof. Vargas

Exemplo de Salto condicional (Branch)

3. Conditional Branch Summary

Test	Boolean	Mnemonic	Complementary	Boolean	Mnemonic	Comment
Rd > Rr	Z•(N ⊕ V) = 0	BRLT(1)	Rd ≤ Rr	Z+(N ⊕ V) = 1	BRGE*	Signed
Rd ≥ Rr	(N ⊕ V) = 0	BRGE	Rd < Rr	(N ⊕ V) = 1	BRLT	Signed
Rd = Rr	Z = 1	BREQ	Rd ≠ Rr	Z = 0	BRNE	Signed
Rd ≤ Rr	Z+(N ⊕ V) = 1	BRGE(1)	Rd > Rr	Z•(N ⊕ V) = 0	BRLT*	Signed
Rd < Rr	(N ⊕ V) = 1	BRLT	Rd ≥ Rr	(N ⊕ V) = 0	BRGE	Signed
Rd > Rr	C + Z = 0	BRLO(1)	Rd ≤ Rr	C + Z = 1	BRSH*	Unsigned
Rd ≥ Rr	C = 0	BRSH/ BRCC	Rd < Rr	C = 1	BRLO/BRCS	Unsigned
Rd = Rr	Z = 1	BREQ	Rd ≠ Rr	Z = 0	BRNE	Unsigned
Rd ≤ Rr	C + Z = 1	BRSH(1)	Rd > Rr	C + Z = 0	BRLO*	Unsigned
Rd < Rr	C = 1	BRLO/BRCS	Rd ≥ Rr	C = 0	BRSH/BRCC	Unsigned
Carry	C = 1	BRCS	No carry	C = 0	BRCC	Simple
Negative	N = 1	BRMI	Positive	N = 0	BRPL	Simple
Overflow	V = 1	BRVS	No overflow	V = 0	BRVC	Simple
Zero	Z = 1	BREQ	Not zero	Z = 0	BRNE	Simple

© Prof. Vargas

Composição de cada comando

- Exemplo:

Mnemônico/ instrução

Complemento

```
subi  Rd,K
```

Registro / Operando

d é um valor de 0 a 31 correspondente aos registros de uso geral

```
subi  R3,0x2E
```

© Prof. Vargas

Obrigado !
A gente se vê na próxima..

© Prof. Vargas