

Arquitetura AVR

ATmega328p

Prof. Roberto de Matos

roberto.matos@ifsc.edu.br



**INSTITUTO
FEDERAL**
Santa Catarina

Câmpus
São José

Objetivo

- Visão geral dos microcontroladores ATmega48A/PA/88A/PA/168A/PA/328/P
- Entender o núcleo do AVR (Arquitetura)
- Introduzir a hierarquia de memória do ATmega328
- Visão geral do circuito integrado (encapsulamentos e pinos)
- Circuito mínimo para o funcionamento do MCU (*clock* e *reset*)

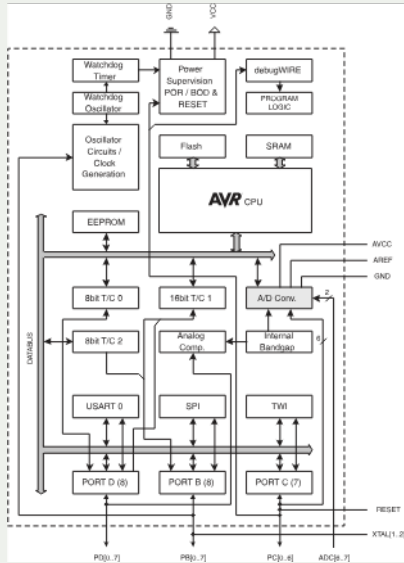


Visão Geral

- Microcontrolador de 8 bits
- Baixo consumo
- Arquitetura Harvard Modificado/RISC
- Maioria das instruções são executadas em 1 ciclo de clock



Diagrama de Blocos do Microcontrolador AVR



- 4/8/16/32KBytes de FLASH reprogramável pelo sistema
- 512/1K/1K/2Kbytes SRAM
- 256/512/512/1Kbytes EEPROM
- 23 linhas de entrada e saída de uso geral (GPIO)
- 32 registradores de uso geral
- 3 Temporizadores/Contadores com comparadores
- Interrupções internas e externas



- Interface serial programável (USART)
- Interface serial de dois fios (2-wire)
- ADC de 10-bit
 - 6 canais no DIP
 - 8 canais nos TQFP e QFN/MLF)
- Watchdog programável com oscilador interno
- 5 modos de economia de energia



- Tamanho das memórias
- Tamanho dos Vetores de Interrupção
- Suporte do bootloader:
 - ATmega88A/PA/168A/PA/328/P suporta mecanismo de auto-programação “real” (*Read-While-Write*) com seção de *Boot Loader* separada e as instruções SPM somente podem rodar dessa seção.
 - ATmega48A/48PA não possui mecanismo *Read-While-Write* e não possui seção de *Boot Loader*. Instruções SPM podem rodar de qualquer parte do código



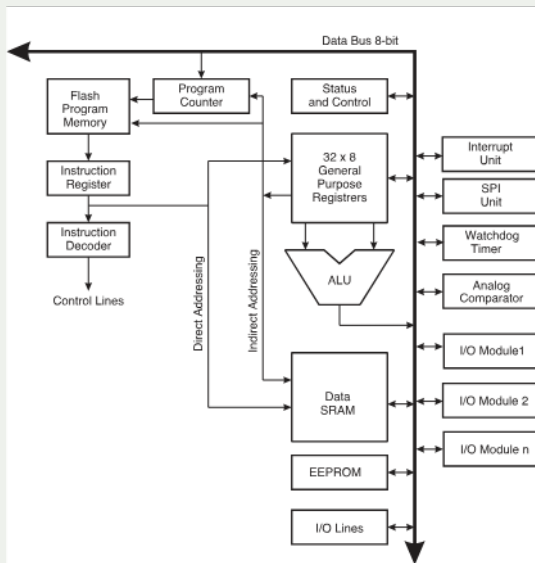
Tamanho das Memórias por Modelo:

Device	Flash	EEPROM	RAM	Interrupt Vector Size
ATmega48A	4KBytes	256Bytes	512Bytes	1 instruction word/vector
ATmega48PA	4KBytes	256Bytes	512Bytes	1 instruction word/vector
ATmega88A	8KBytes	512Bytes	1KBytes	1 instruction word/vector
ATmega88PA	8KBytes	512Bytes	1KBytes	1 instruction word/vector
ATmega168A	16KBytes	512Bytes	1KBytes	2 instruction words/vector
ATmega168PA	16KBytes	512Bytes	1KBytes	2 instruction words/vector
ATmega328	32KBytes	1KBytes	2KBytes	2 instruction words/vector
ATmega328P	32KBytes	1KBytes	2KBytes	2 instruction words/vector



Núcleo do AVR

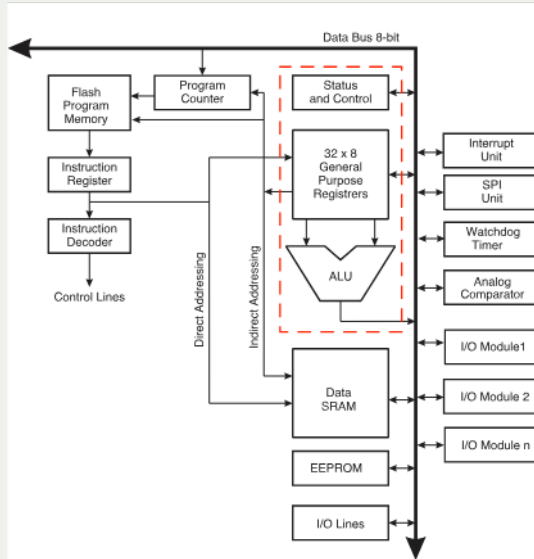
Diagrama de Blocos do Núcleo do AVR

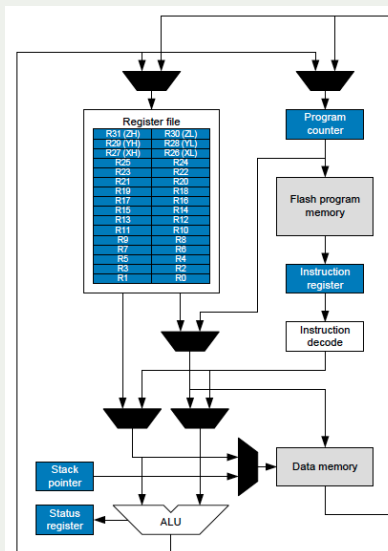


- **Arquitetura Harvard modificada:** Além de possuir barramentos separados para memória de dados (SRAM) e programa (FLASH), permite ler e escrever a memória de programa usando instruções especiais.
- **Pipeline de um nível:** Enquanto uma instrução está sendo executada, a próxima instrução é buscada na memória do programa. Este conceito permite que as instruções sejam executadas em cada ciclo do relógio, ou seja, a performance de um AVR de 16 MHz é aproximadamente 16 MIPS (Milhões de Instruções por Segundo).
- A **simplicidade** da arquitetura se traduz em uma curva de aprendizado mais rápida.



Unidade Lógica Aritmética





- **Operandos:** Registradores de uso geral ou entre um registrador e um imediato (constante armazenada na memória de programa).
- **Operações:** Aritmética, lógica e funções de bit. Alguns microcontroladores da família também provêem suporte para multiplicação (com sinal, sem sinal e fracionário).
- **Status:** Após uma operação na ULA, o Registrador de Status é atualizado para refletir as informações sobre o resultado da operação.



Bit	7	6	5	4	3	2	1	0	
0x3F (0x5F)	I	T	H	S	V	N	Z	C	SREG
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	

I – Habilitação de interrupção global (*Global Interrupt Enable*)

T – Bit de armazenamento (*Bit Copy Storage*)

H – Sinalizador de carry em nibble (*Half Carry Flag*)

S – Bit de sinal, $S = N \oplus V$ (*Sign Bit*)

V – Sinalizador de estouro do complemento de dois (*Two's Complement Overflow Flag*)

N – Sinalizado de negativo (*Negative Flag*)

Z – Sinalizador de zero (*Zero Flag*)

C – Sinalizador de carry (*Carry Flag*)



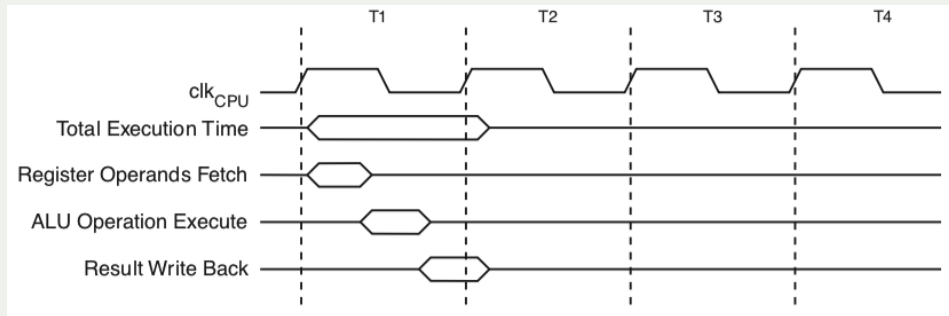
ULA - Exemplo Instruções

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
ADIW	Rdl,K	Add Immediate to Word	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
SBIW	Rdl,K	Subtract Immediate from Word	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \bullet Rr$	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \bullet K$	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z,C,N,V	1
NEG	Rd	Two's Complement	$Rd \leftarrow 0x00 - Rd$	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \bullet (0xFF - K)$	Z,N,V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \bullet Rd$	Z,N,V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z,N,V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2



Ciclo de Operação da ULA

Em um único ciclo de relógio a ULA executa uma operação usando dois operandos de registro e o resultado é armazenado de volta no registro de destino.



Forma de Acesso

Os registradores de uso geral são otimizados para o conjunto de instruções do AVR. Para obter o desempenho e a flexibilidade necessários, os seguintes esquemas de entrada/saída são suportados pelo banco de registradores.

Operando(s) Saída	Operando(s) Entrada
1x de 8 bits	1x de 8 Bits
2x de 8 bits	1x de 8 Bits
2x de 8 bits	1x de 16 Bits
1x de 16 bits	1x de 16 Bits



Registradores de Uso Geral

7

0

Addr.

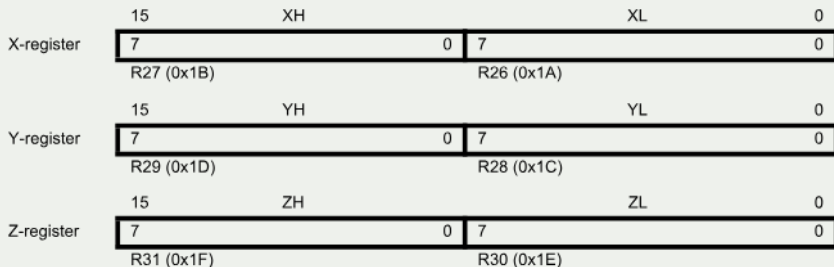
R0	0x00	
R1	0x01	
R2	0x02	
...		
R13	0x0D	
R14	0x0E	
R15	0x0F	
R16	0x10	
R17	0x11	
...		
R26	0x1A	X-register Low Byte
R27	0x1B	X-register High Byte
R28	0x1C	Y-register Low Byte
R29	0x1D	Y-register High Byte
R30	0x1E	Z-register Low Byte



- Os registradores R26 ... R31 também possuem funções especiais. Eles são ponteiros de endereço de 16 bits para endereçamento indireto, definidos como X, Y e Z.
- Nos diferentes modos de endereçamento, esses registradores de endereços têm funções como deslocamento fixo, incremento automático e decréscimo automático.

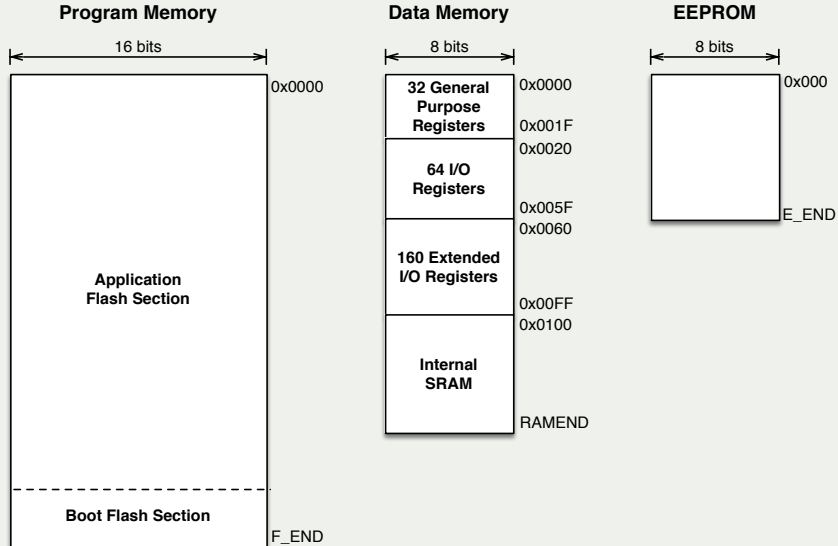


- Os registradores R26 ... R31 também possuem funções especiais. Eles são ponteiros de endereço de 16 bits para endereçamento indireto, definidos como X, Y e Z.
- Nos diferentes modos de endereçamento, esses registradores de endereços têm funções como deslocamento fixo, incremento automático e decréscimo automático.



Memórias do AVR

Memórias – Organização



■ Endereços Finais vs. Modelos:

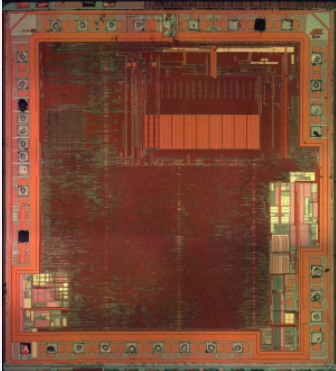
Modelo	F_END	RAMEND	E_END
ATmega88*	0x0FFF	0x04FF	0x1FF
ATmega168*	0x1FFF	0x04FF	0x1FF
ATmega328*	0x3FFF	0x08FF	0x3FF

■ Tamanhos vs. Modelos:

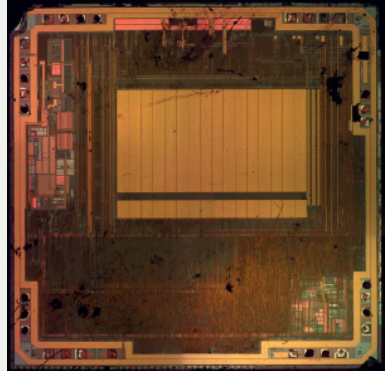
Modelo	FLASH	RAM	EEPROM
ATmega88*	4K x 16 bits	1K x 8 bits	512 x 8 bits
ATmega168*	8K x 16 bits	1K x 8 bits	512 x 8 bits
ATmega328*	16K x 16 bits	2K x 8 bits	1K x 8 bits



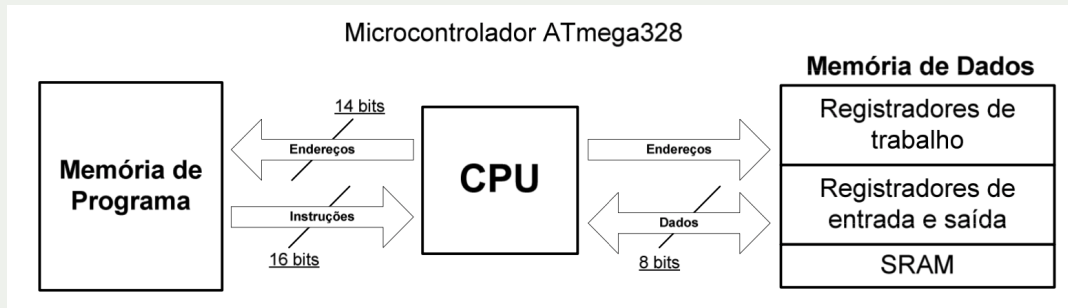
■ ATmega48: 7.01mm^2



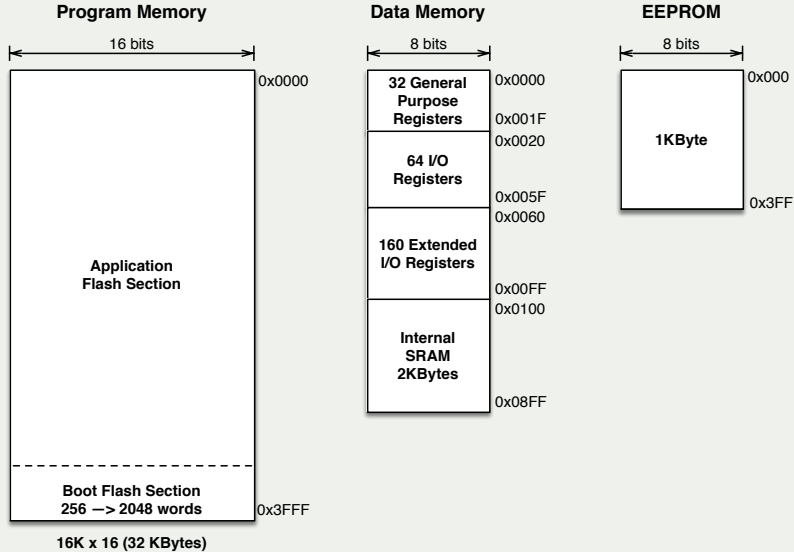
■ ATmega328p: 8.82mm^2



¹Fonte: <https://siliconpr0n.org>

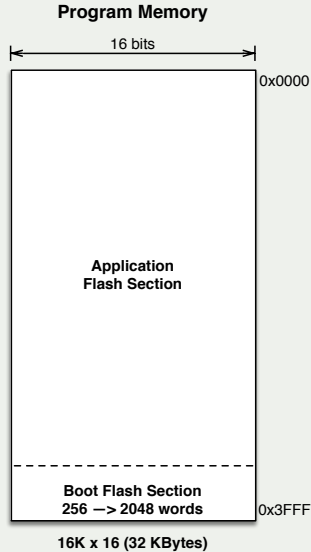


Memórias ATmega328p - Detalhe



FLASH

Memória de Programa - FLASH

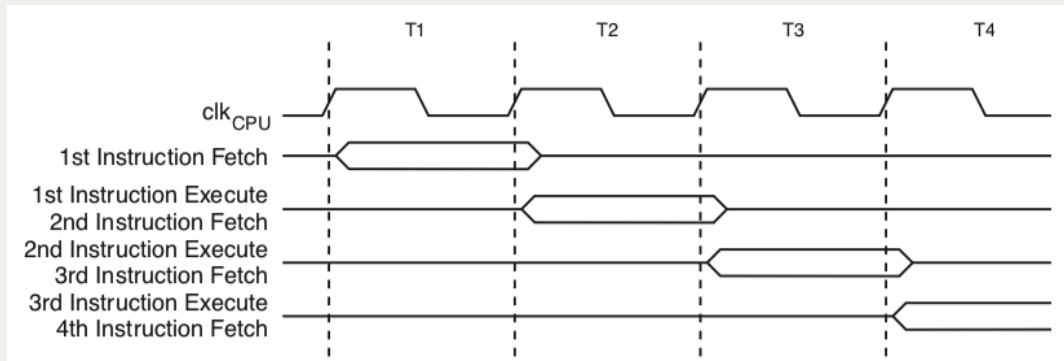


- Como todas as instruções do AVR são de 16 ou 32 bits, a palavra de memória da FLASH foi projetada para ter 16 bits, facilitando a *busca* das instruções.
- A memória tem uma durabilidade de 10.000 ciclos de re-escrita.
- O tamanho do PC para o ATmega48A/PA/88A/PA/168A/PA/328/P é de 11/12/13/14 bits respectivamente.
- Exceto pelo ATmega48*, todo os outros modelos possuem área de bootloader e permitem escrita na FLASH durante a leitura.
- Tabelas de constantes podem ser alocadas em qualquer ponto da FLASH. Para leitura é usado a instrução LPM (*Load Program Memory*).



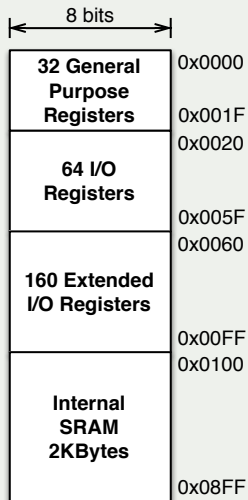
Diagrama de tempo de Busca e Execução

O funcionamento do *pipeline* é possibilitado pela palavra de 16 bits da memória de programa, a arquitetura Harvard e o conceito de acesso rápido ao banco de registradores.

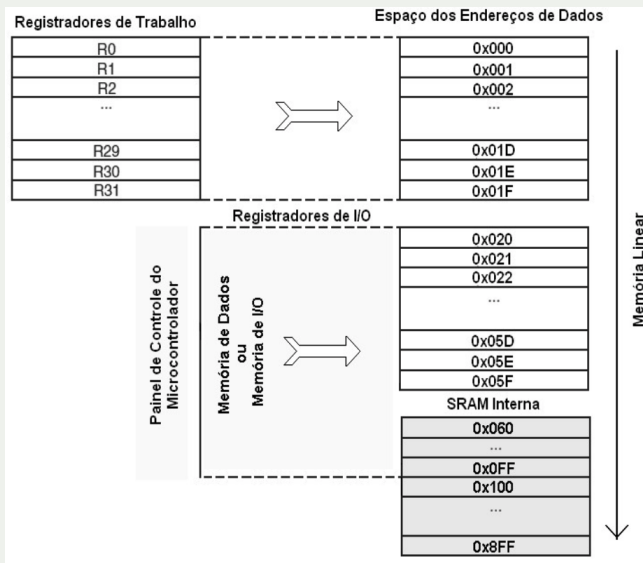


Memória de Dados

Data Memory



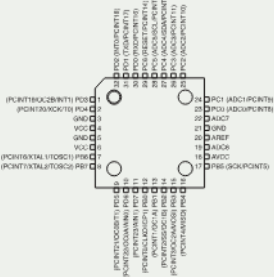
Memória de Dados – Detalhe



Circuito Integrado

Encapsulamentos e Pinos

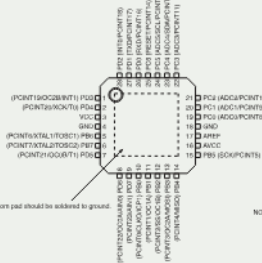
32 TOFP Top View



28 PDIP

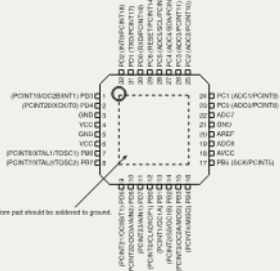


28 MLF Top View



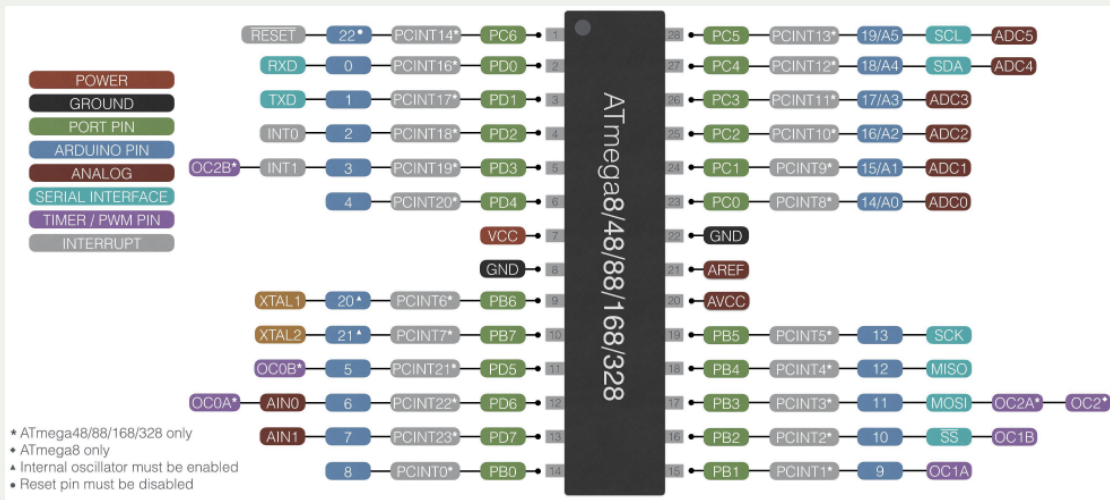
NOTE: Before pad should be soldered to ground.

32 MLF Top View



NOTE: Bottom pad should be soldered to ground.

Pinos do Encapsulamento DIP¹

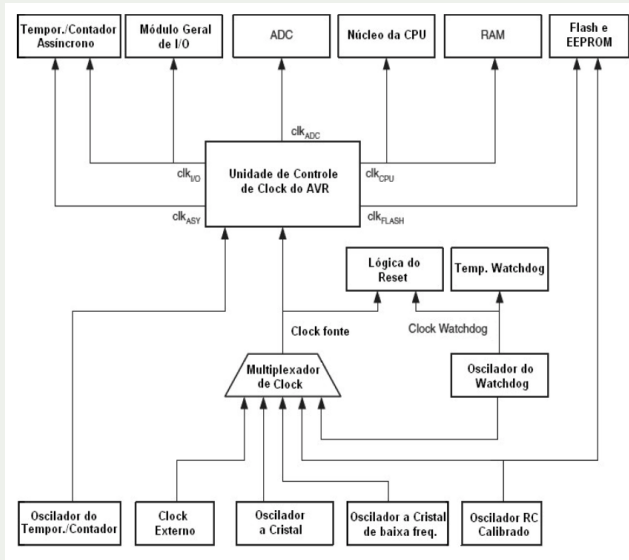


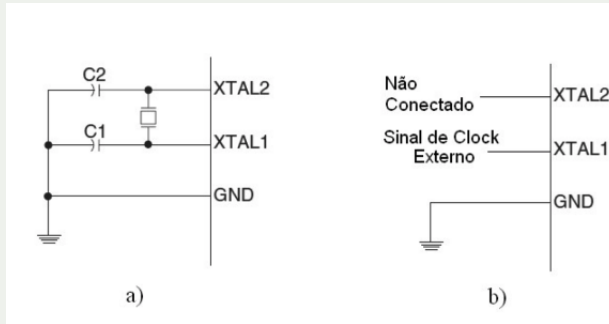
¹Adaptado de <https://github.com/MCUdude/MiniCore>



Clock

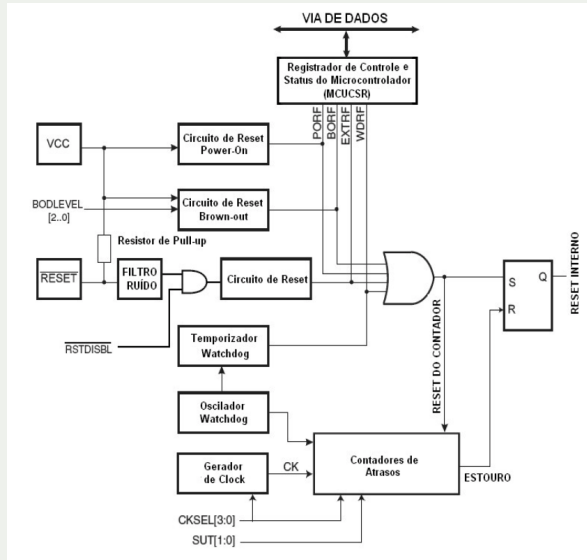
Clock - Esquema interno



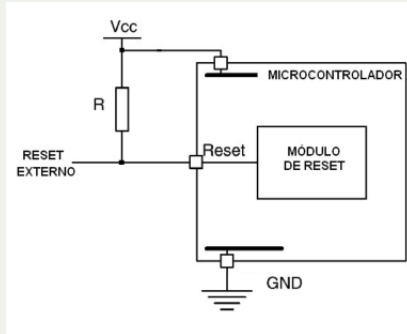


Reset

Reset - Esquema interno

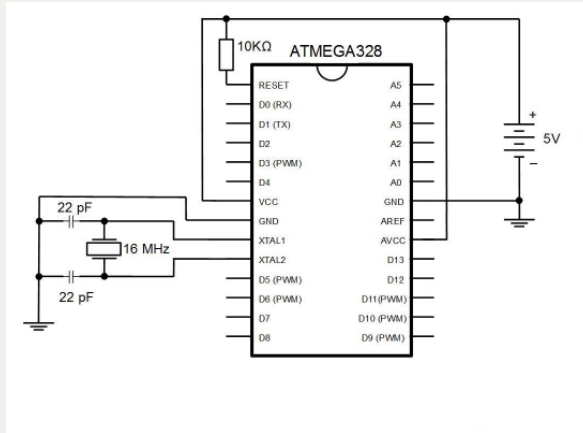


Reset - Exemplo de circuito

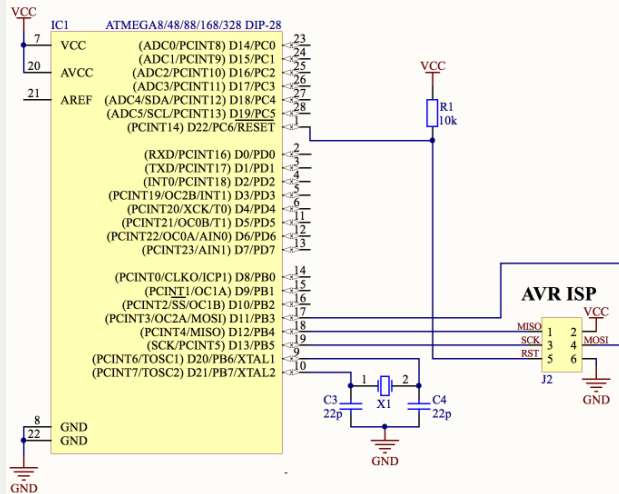


Circuito mínimo

Exemplo de circuito mínimo com clock externo



Exemplo de circuito mínimo com clock externo e gravação¹



¹Adaptado de <https://github.com/MCUdude/MiniCore>



Fim!