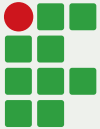
Projetando, codificando e simulando para MCU

Assembly do ATmega328

Prof. Roberto de Matos

roberto.matos@ifsc.edu.br



INSTITUTO FEDERAL
Santa Catarina
Câmpus São José

Objetivo

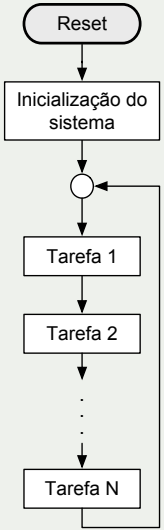
Objetivo

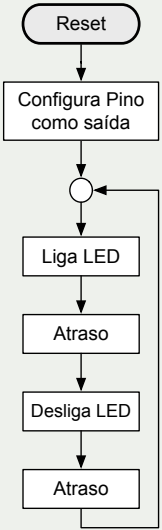
- Entender como projetar um *software* embarcado básico
- Entender como codificar um *software* em Assembly
- Introdução à simulação

Software embarcado básico

Software embarcado básico - Exemplo

- Conhecido como *Round-robin* ou “loopão”
- Simplicidade
- Funcionamento básico:
  - Configura o sistema logo depois do *reset*
  - Executa as tarefas de forma sequencial e repetida
- Exemplo: Pisca-pisca





Fluxograma

Fluxograma - Símbolos básicos

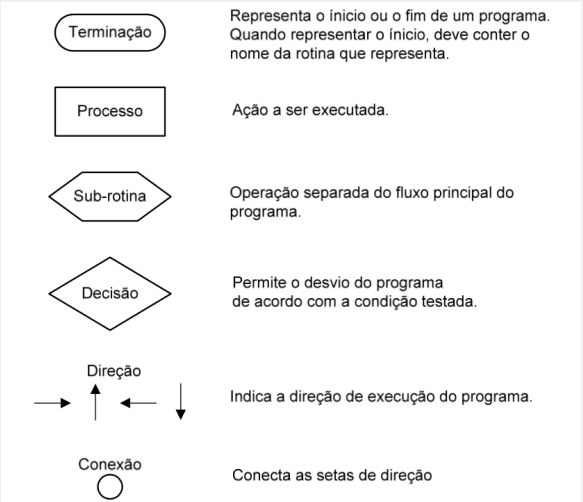
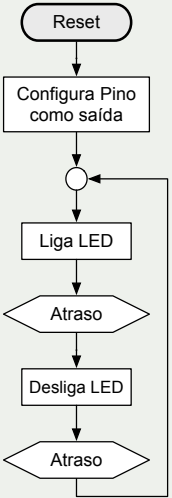
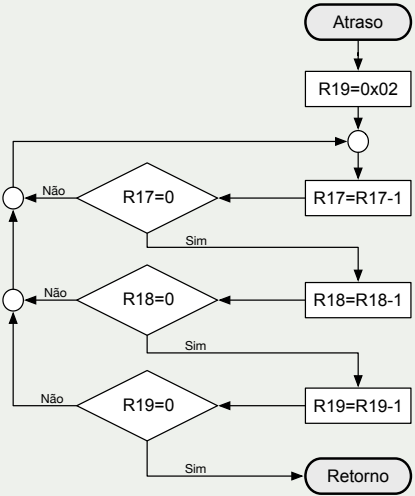


Fig. 4.1– Símbolos básicos para o desenho de fluxogramas.

Fluxograma - Exemplo





Programando em *Assembly*

Programando em *Assembly* - Pontos chaves

- Planejar e modelar o código
- Conhecer a arquitetura (hierarquia de memória, periféricos, etc.)
- Conhecer as instruções *assembly*[1]
- Conhecer as diretivas do montador[2] (*assembler*)

[1] Manual do conjunto de instruções do AVR
[2] Manual online do montador

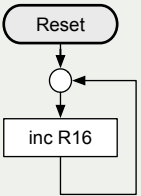
Programando em *Assembly* - Arquivo Fonte

- Sintaxe do arquivo:

```
1  [Rótulo:] .diretiva  [operandos] [comentários]
2  [Rótulo:] instrução [operandos] [comentários]
3  Comentários
4  Linhas em branco
```

- Entre [ ] é opcional
- Diretivas → Montador
- Instruções → CPU
- Todo texto após “;” ou “//” é comentário
- Exemplo:

```
1 label: .EQU var1=100 ; associa a constante 100 à var1 (Diretiva)
2        .EQU var2=200 ; associa a constante 200 à var2 (Diretiva)
3
4 start: INC r16       ; incrementa o r16 (Instrução)
5        RJMP start    ; salta para o rótulo ``start'' (Instrução)
6                       ; linha com comentário
7                       // outro comentário
```



Diretivas do Montador[1]

- Definição do seguimento de dados ou código: **.DSEG**, **.CSEG**, **.ESEG**
- Reserva de bytes na memória SRAM (variável): **.BYTE**
- Reserva de bytes/words (constantes): **.DB**, **.DW**
- Definir um nome simbólico para um registrador: **.DEF**
- Setar um símbolo a partir de uma expressão: **.EQU**, **.SET**
- Setar um endereço de origem: **.ORG**
- Incluir um arquivo fonte: **.INCLUDE**

[1] Verificar lista completa de diretivas no manual online do montador.

Formatos de inteiros constantes

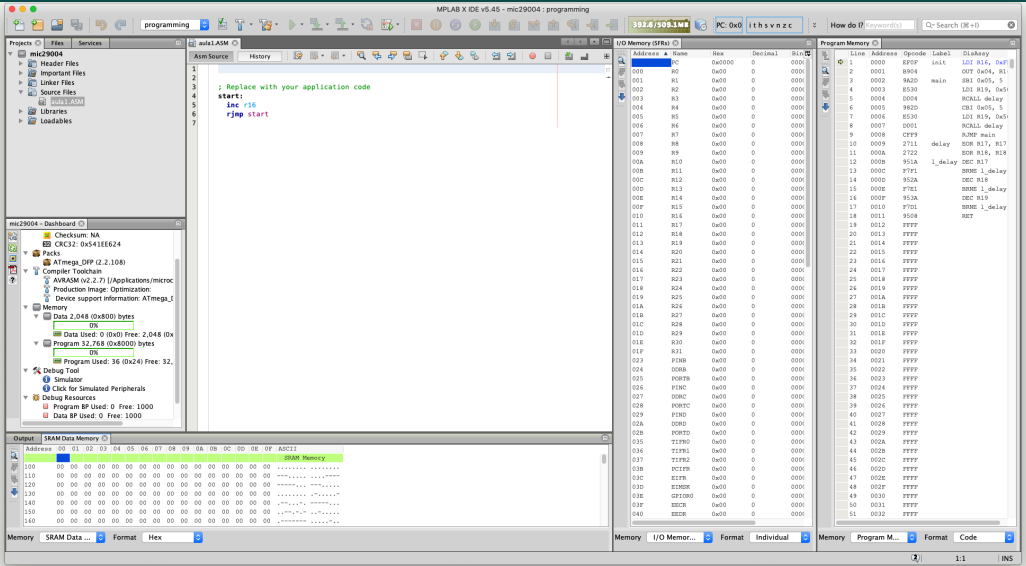
- Decimal: 10, 255
- Hexadecimal: 0x0a, $0a, 0xff, $ff
- Binário: 0b00110101, 0b1100_1010
- Octal (zero na frente): 010, 077

Ambiente de desenvolvimento e simulação

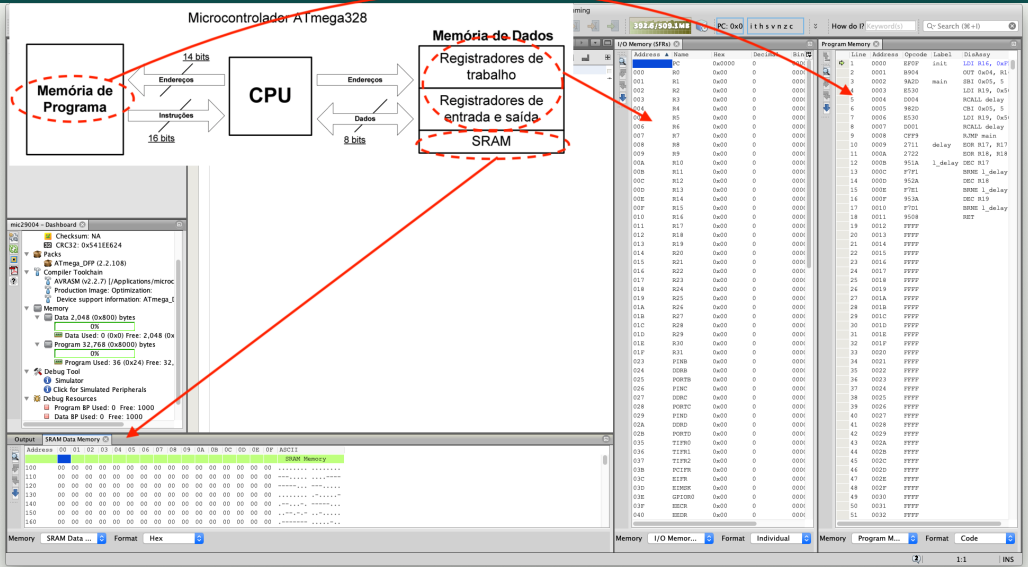
Abrindo o Projeto

- Descompactar o arquivo mic29004.zip em um local conhecido.
- Abra o MPLAB X IDE
- Clique em *File -> Open Project*
- Selecione o diretório descompactado
- Clique em *Open Project*
- No painel no canto superior esquerdo *Projects*, expanda o projeto e o diretório *Source Files*
- Abra o arquivo `aula1.asm`
- Organize as janelas para a simulação conforme sugestão

Sugestão de organização



Arquitetura vs. simulador



Simulando o código

- Monte o programa:
  - *Production -> Build Main Project* (atalhos: F11 ou ícone do “martelo").
- Inicie a simulação:
  - *Debug -> Discrete Debugger Operation -> Launch Debugger Main Project*
- Use os controles do debugger para a simular o código:

Fim!