ATMEGA328P-PU EM C

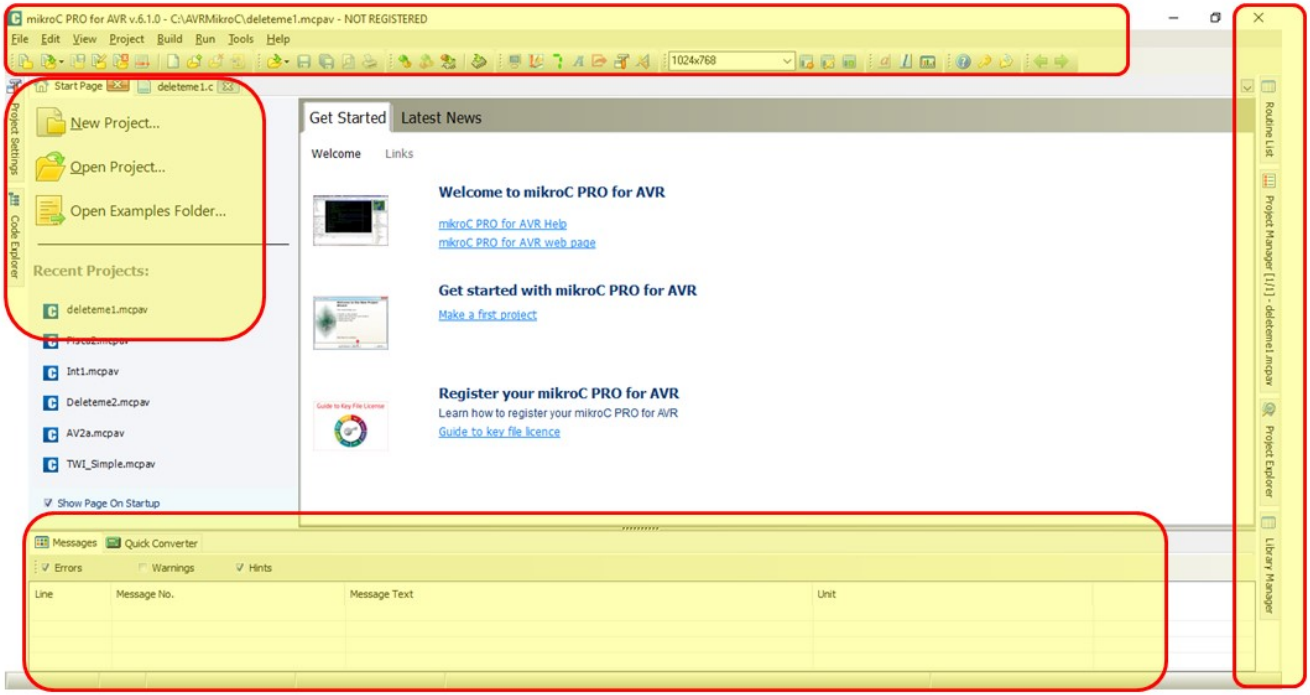
© Prof. Engº esp Luiz Antonio Vargas Pinto
www.vargasp.com
Atualizado em 2026

Criar Ambiente

- Crie uma pasta na raiz do HD ou, preferencialmente na pendrive, com o nome "MikroC"
- Abra o MikroC já instalado

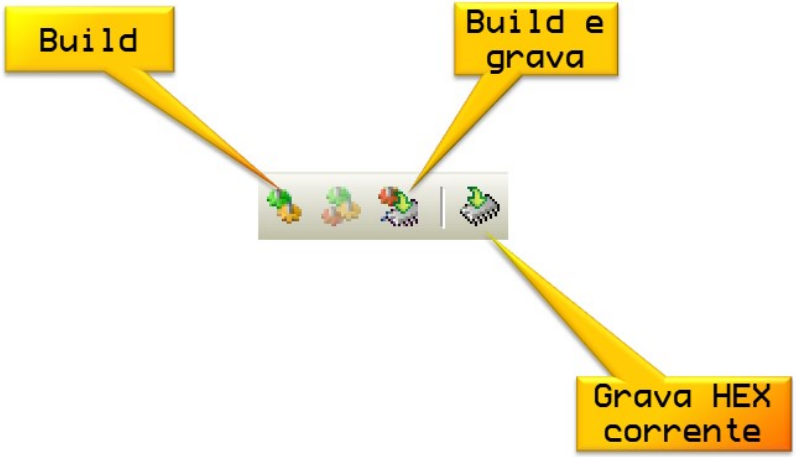
© Prof. Vargas

Tela principal



© Prof. Vargas

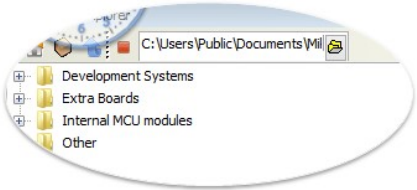
Sobre os comandos



© Prof. Vargas

Menu lateral

Árvore de projeto

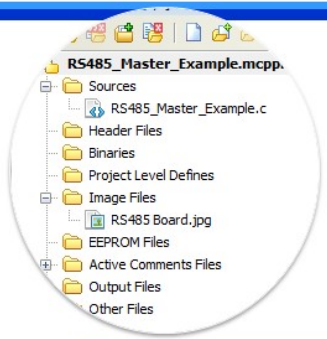


Routine List

Project Manager [1/1] - RS485_Master_Example.mcppi

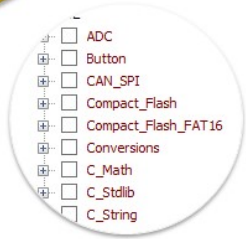
Project Explorer

Library Manager



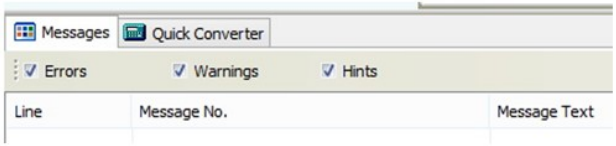
Informações de projeto

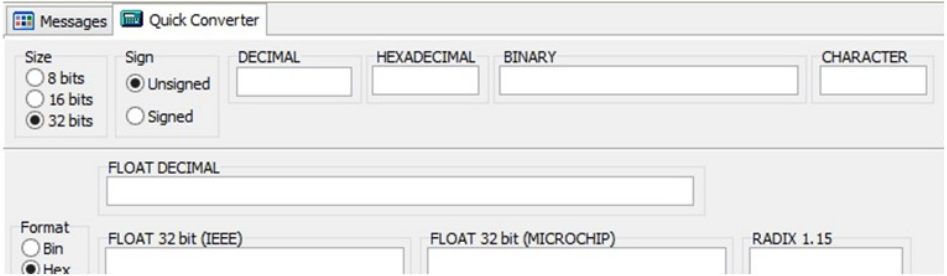
Biblioteca



© Prof. Vargas

Janelas auxiliares

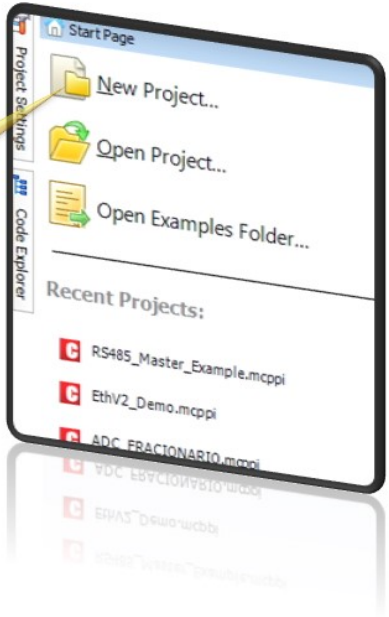


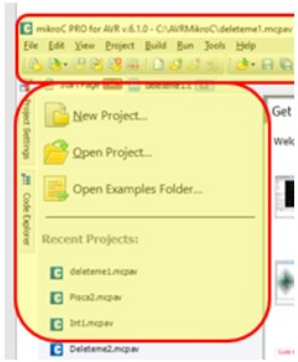


© Prof. Vargas

Abre Projeto

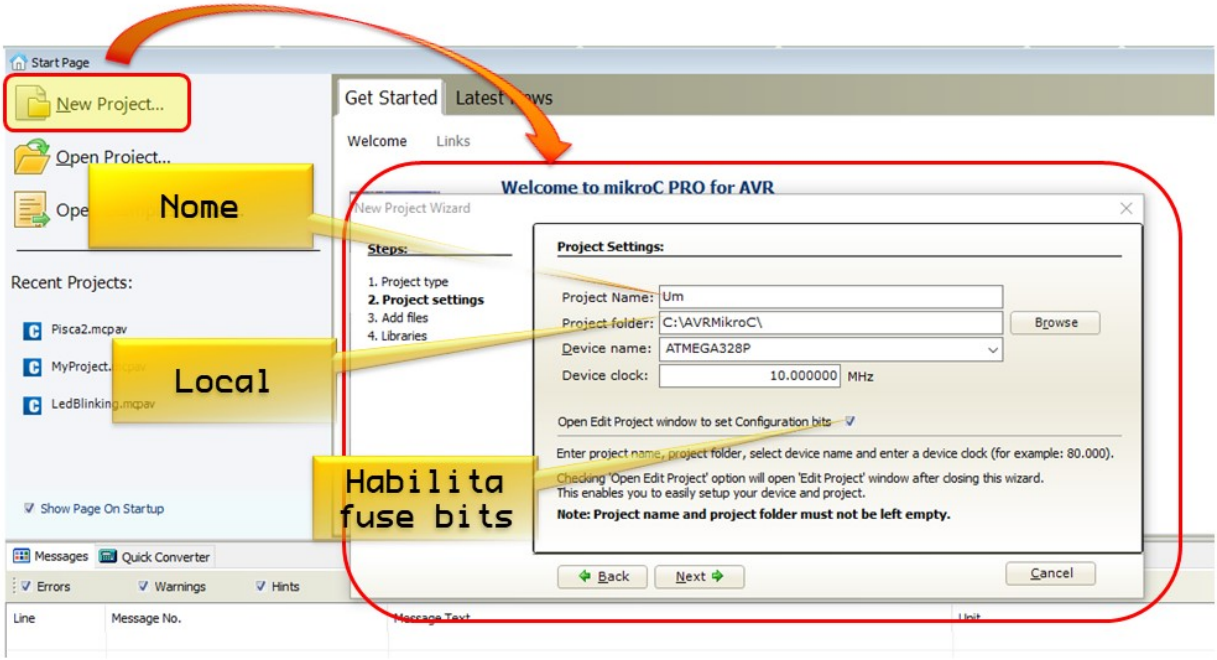
Abertura de novo projeto





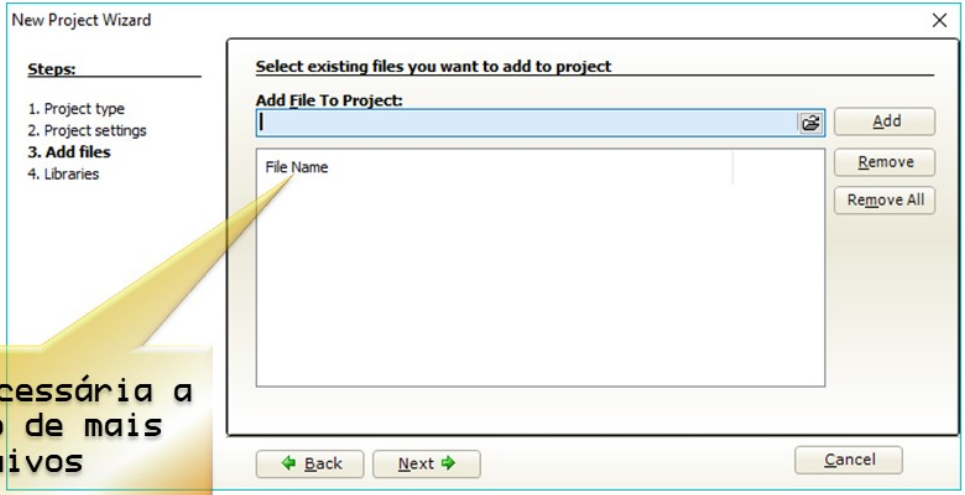
© Prof. Vargas

Um "New Project"



© Prof. Vargas

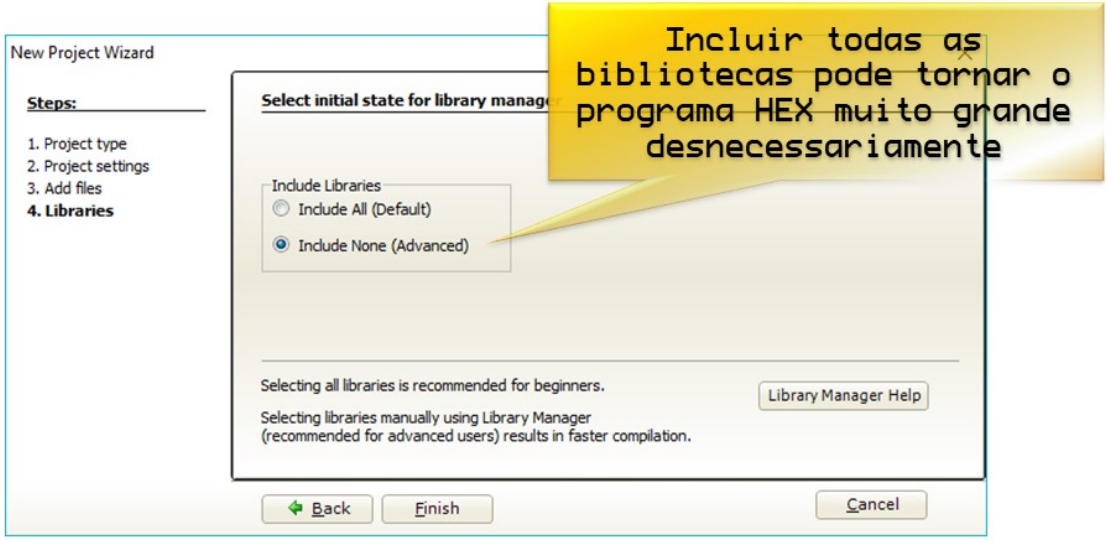
Adicionar arquivos



Se for necessária a inclusão de mais arquivos

© Prof. Vargas

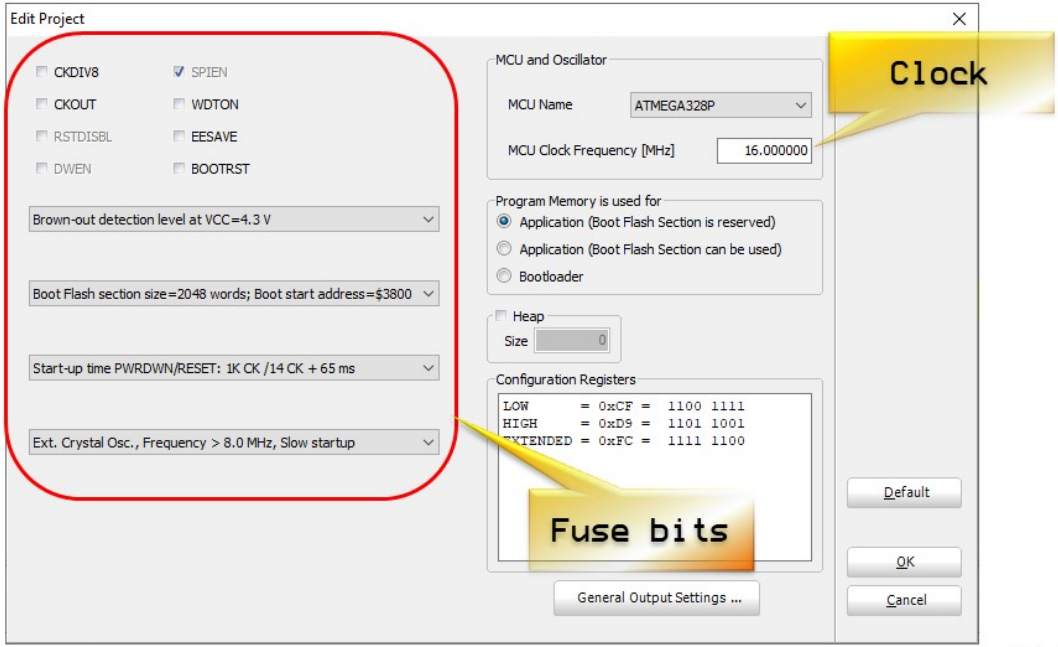
Bibliotecas



© Prof. Vargas

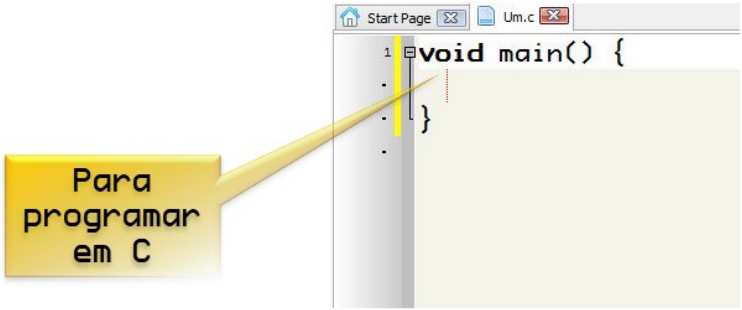
Os fuse bits

- Aqui definimos as características necessárias ao ATMEGA328p-pu.



© Prof. Vargas

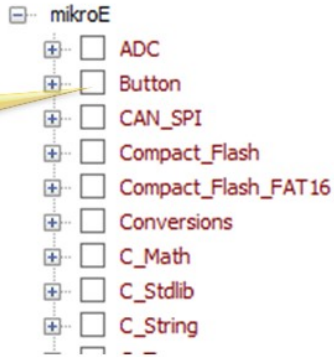
Janela final para criar código



© Prof. Vargas

Seleção de biblioteca

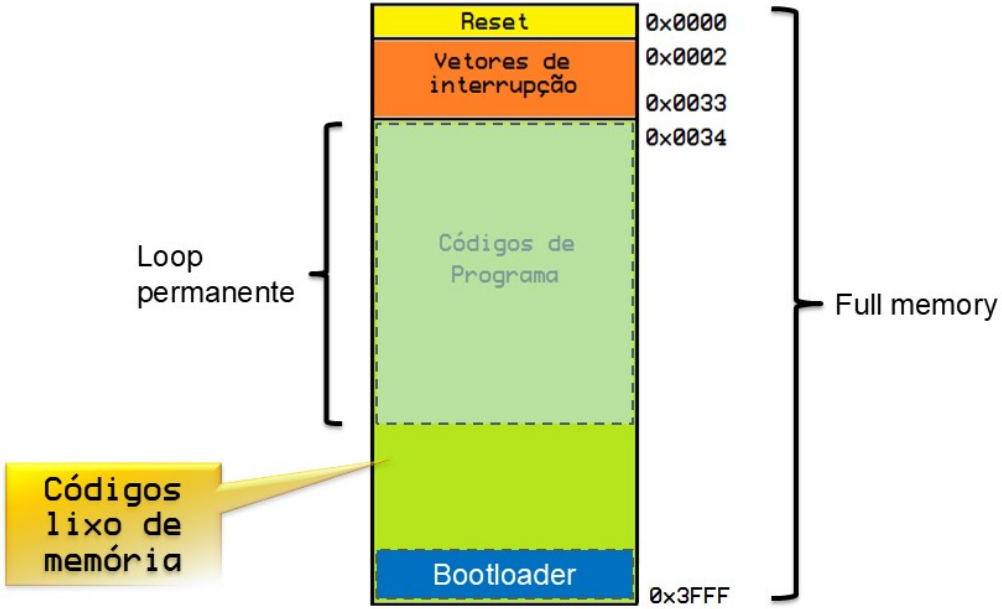
Clicando aqui habilitamos as funções de botão, por exemplo



© Prof. Vargas

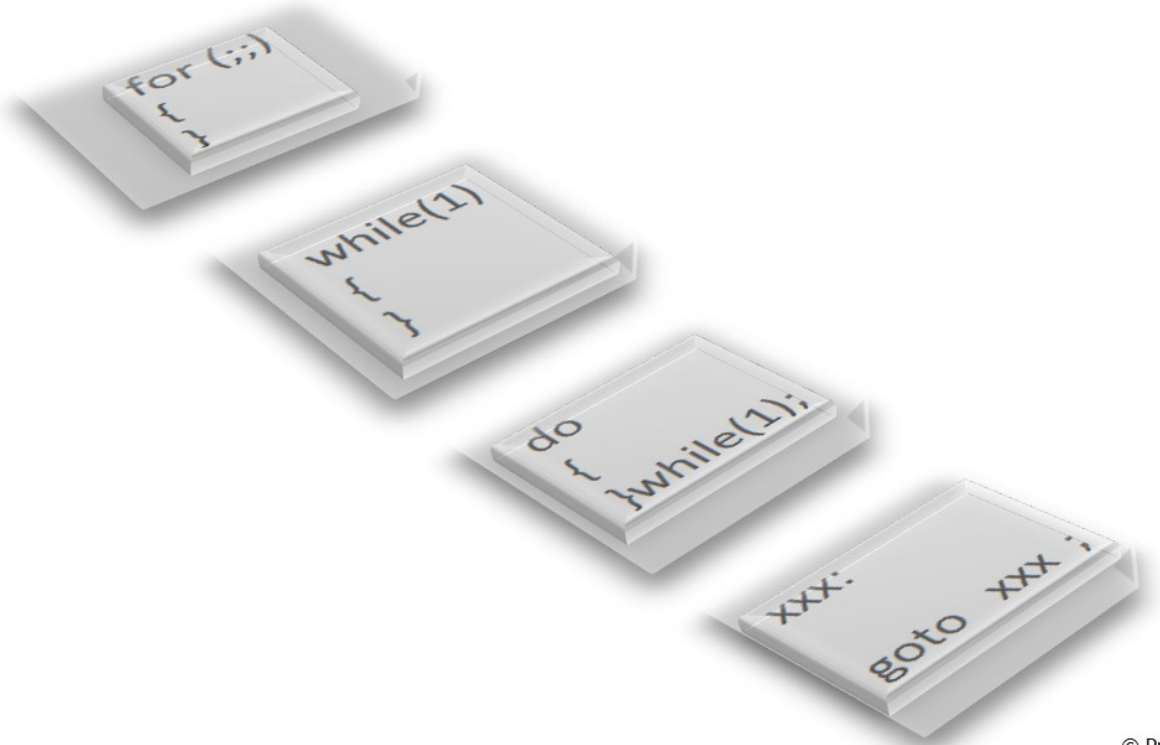
Loop permanente

- Isola a parte útil da flashRom, memória de 32768 endereços, onde estão os códigos de programa atual.



© Prof. Vargas

Opções para o loop permanente



© Prof. Vargas

O programa

```
 1: /*-------------------------------------------------------
 2:  Programa : Um.c
 3:  Data     : 25/05/2021
 4:  Autor    : Prof. Vargas
 5:  Descrição:
 6:     Como exercício inicial vamos criar um programa que faz os
 7:     LEDs D1 a D6 piscarem alternadamente com 0.5S de tempo.
 8:     Sendo que os mesmos estão conetados respectivamente em
 9:     PB0 a PB5.
10:  -------------------------------------------------------*/
11: void main()
12: {
13:    PORTB = 0b11010101; //dado que impede qualquer LED acender
14:    DDRB  = 0b00111111; //define quem é entrada e quem é saída
15:    for(;;)
16:    {
17:       PORTB.B0 = !PORTB.B0; //muda estado do LED D1
18:       PORTB.B1 = !PORTB.B1; //muda estado do LED D2
19:       PORTB.B2 = !PORTB.B2; //muda estado do LED D3
20:       PORTB.B3 = !PORTB.B3; //muda estado do LED D4
21:       PORTB.B4 = !PORTB.B4; //muda estado do LED D5
22:       PORTB.B5 = !PORTB.B5; //muda estado do LED D6
23:       Delay_ms(500);  //temporiza 0.5S
24:    }//final do loop permanente
25: }
26:
```

© Prof. Vargas

Já alguns cuidados com I/O

- PortC tem 7 bits

© Prof. Vargas

© Prof. Vargas