

PRIMEIRA AVALIAÇÃO DE E 209 – 12/09/2022

PROF. JOÃO PEDRO MAGALHÃES DE PAULA PAIVA

NOME: _____ MAT.: _____ PERÍODO _____ CURSO: _____

ORIENTAÇÕES GERAIS DA AVALIAÇÃO:

1. Avaliação **Individual** e com consulta.
2. A interpretação faz parte da avaliação.
3. Bons estudos e boa avaliação!

ORIENTAÇÕES ADICIONAIS:

1. Essa avaliação contém 3 questões. CONFIRA!!!!!!!!!!!!!!

1. (30 Pontos) Analise as 10 frases a seguir e justifique as 4 sentenças falsas.
Observação: A não justificativa de 4 sentenças acarretará na anulação dessa questão.

Afirmativas

- I. Um sistema computacional microcontrolado é caracterizado por um processador, em conjunto com seus periféricos que realizam a interface com dispositivos de entradas e saídas eletrônicos, assim como uma interface homem-máquina, além de memórias para armazenamento de dados utilizados pelo programa de aplicação dedicado para o sistema, o denominado hardware.
- II. Os microcontroladores podem ser aplicados em diversos cenários dentro da engenharia. Um cenário é o controle de diversos periféricos existentes em veículos automotores como o controle do acendimento de faróis em decorrência da baixa luminosidade da rodovia e, atualmente, também é possível utilizar um controle totalmente automático de detecção de colisões para aumentar a segurança dos passageiros.
- III. Um microcontrolador é um pequeno computador em um único circuito integrado. Um microcontrolador contém um ou mais núcleos de processamento junto com uma memória e periféricos de entrada e saída programáveis. Esses dispositivos são utilizados em produtos com controle automático de sua operação. Por meio da redução de tamanho e custo comparado à um projeto que utiliza microprocessador, memória e dispositivos separados, os microcontroladores tornam mais econômicas as digitalizações realizadas em sistemas legados.
- IV. Como partes constituintes do microcontrolador, podem ser citadas a CPU, as memórias volátil e não-volátil e as linhas de I/O responsáveis por capturar e entregar os dados de entradas e saídas para os periféricos. No ATmega328, por exemplo, é possível utilizar até 2 periféricos de comunicação com dispositivos externos: A USART e a SPI.
- V. Quando um sistema de processamento de dados (processadores e microcontroladores) possui uma única área de memória na qual ficam armazenados os dados (variáveis) e o programa a ser executado (software), dizemos que esse sistema segue a arquitetura de Von Neuman. No caso em que os dados (variáveis) ficam armazenados em uma

área de memória e o programa a ser executado (software) fica armazenado em outra área de memória, dizemos que esse sistema segue a arquitetura Harvard.

- VI. Os microcontroladores AVR RISC da fabricante ATMEL possuem arquitetura Von Neuman. Ele possui três memórias: uma memória de 32kB de memória flash para armazenamento de programas, uma memória de 2kB RAM estática para armazenamento de dados e uma memória de 1kB EEPROM para armazenamento não-volátil.
- VII. Quando desejamos programar um microcontrolador, é fundamental o conhecimento de sua arquitetura. Um dos pontos iniciais de grande importância é a manipulação de registradores. Um registrador é um tipo de memória de pequena capacidade porém muito rápida, contida na CPU, utilizado no armazenamento temporário de dados durante o processamento. Os registradores estão no topo da hierarquia de memória, sendo desta forma o meio mais rápido e de menor custo para armazenar um dado.
- VIII. Para declarar os pinos 3 e 6 do portal D do ATmega328 como entrada, além de declarar os pinos 4 e 7 como saída, precisamos declarar o registrador DDRD como 90 (hexadecimal). Ainda, para declarar o pino 6 do portal C como saída, é necessário declarar o registrador DDRC como 40 (hexadecimal).
- IX. Caso seja necessário utilizar uma interrupção externa com transição de subida no pino PD4, precisamos declarar os seguintes registradores:
 - a. EICRA = BIT0 + BIT1;
 - b. EIMSK = BIT0;
- X. Caso seja necessário utilizar uma interrupção no pino PC0, precisamos declarar os seguintes registradores:
 - a. PCICR = BIT1;
 - b. PCMSK1 = BIT0;
 - c. sei();

Sentença 1:

Sentença 2:

Sentença 3:

Sentença 4:

2. (30 Pontos) Analise os códigos a seguir, indicando seu funcionamento e, caso necessário, aponte falhas nas configurações e como corrigi-las.

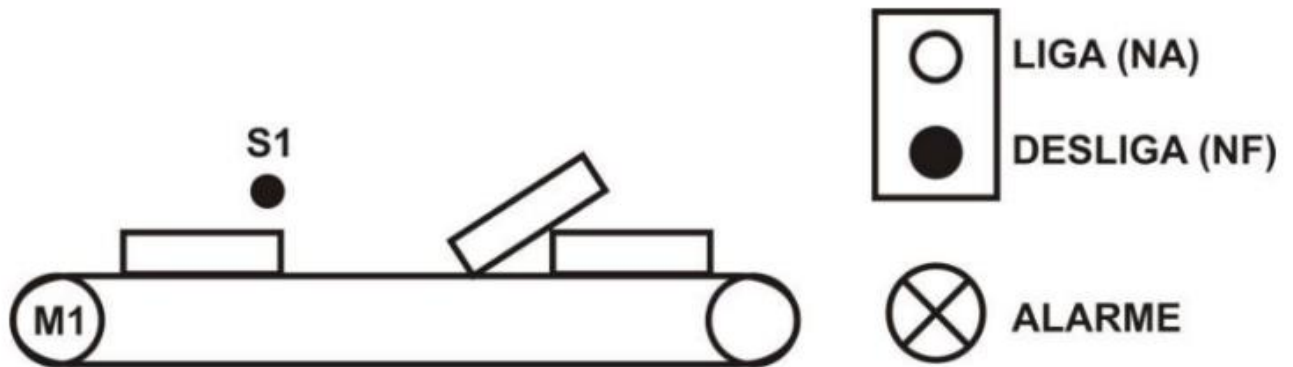
a)

```
1  #define BIT0 0b00000001
2  #define BIT1 0b00000010
3  #define BIT2 0b00000100
4  #define BIT3 0b00001000
5  #define BIT4 0b00010000
6  #define BIT5 0b00100000
7  #define BIT6 0b01000000
8  #define BIT7 0b10000000
9
10 int main (){
11
12     DDRB = BIT4 + BIT7;
13     PORTB = BIT1 + BIT5;
14     DDRD = BIT6;
15
16     for(;;){
17
18         if((PINB && BIT1 == 0) || (PINB && BIT5 == 0)){
19
20
21             PORTB |= BIT7;
22             PORTB &= ~BIT4;
23
24         }
25
26         else{
27
28             PORTB |= BIT4;
29             PORTB &= ~BIT7;
30
31         }
32
33     }
34
35 }
```

b)

```
1  #define BIT0 0b00000001
2  #define BIT1 0b00000010
3  #define BIT2 0b00000100
4  #define BIT3 0b00001000
5  #define BIT4 0b00010000
6  #define BIT5 0b00100000
7  #define BIT6 0b01000000
8  #define BIT7 0b10000000
9
10 int main (){
11
12     DDRD = BIT1;
13     EICRA = BIT0 + BIT1;
14     EIMSK = BIT0;
15
16     for(;;){
17
18     }
19
20 }
21
22 ISR(INT0_vect){
23
24     PORTB |= BIT1;
25     _delay_ms(500);
26     PORTB &= ~BIT1;
27     _delay_ms(500);
28
29 }
30
```

3. (40 pontos) Em uma esteira de transporte, foi instalado um sistema de verificação de peças posicionadas de forma errada. Elabore um firmware para controlar o sistema, seguindo os passos apresentados abaixo.
- Ao pressionar o botão LIGA (NA) a esteira entra em movimento (MOTOR = TRUE);
 - Ao pressionar o botão DESLIGA (NF) a esteira para seu movimento (MOTOR = LOW);
 - Caso aconteça o amontoamento de peças (S1 = HIGH), a esteira deverá parar imediatamente e alarme deverá ser ligado (ALARME = HIGH);
 - Enquanto as peças estiverem amontoadas, a esteira não poderá ser ligada;
 - Para desligar o ALARME, as peças deverão estar desamontoadas (S1 = LOW) e o botão LIGA deve ser pressionado.



Entradas / Saídas	GPIO Utilizado
Motor	PD3
Sensor 1	PB4
Botão Liga	PB1
Botão Desliga	PB2
Alarme	PD7