



CEFET-MG — Centro Federal de Educação Tecnológica de Minas Gerais
DEPARTAMENTO DE COMPUTAÇÃO DE DIVINÓPOLIS — DECOM-DV

Microprocessadores e Microcontroladores

Primeira Atividade Avaliativa

Aluno: KEMILY REZENDE SILVA

Valor: 30 pontos (cada questão vale 5 pontos)

Turma: 2024/1

Prof. M. Sc. Diego Ascânio Santos

Respostas:

1 2 3 4 5 6

Questão 1

Avalie as assertivas:

- I. Todos os sinais de circuitos são elétricos, porém, podem ser categorizados em dois tipos: analógicos e digitais.
- II. Não é possível para sinais analógicos assumirem qualquer valor arbitrário dentro de um intervalo especificado.
- III. Sinais digitais são representados por valores contínuos.
- IV. Sinais digitais são representados por valores discretos.
- V. A tensão digital em nível lógico ALTO (TTL) é de +5V.

Assinale a alternativa que contém todas as assertivas corretas:

- a) I, II, III e V.
- b) I, III e V.
- c) II, III, IV e V.
- d) I, IV e V.
- e) I e IV.

Questão 2

Interrupções no arduino podem ser habilitadas e associadas a funções *callback* — que são executadas quando interrupções ocorrem — através do **ISR** (Interrupt Service Routine). Em relações aos recursos e limitações que as funções de *callback* apresentam, é correto afirmar que:

- a) Funções de *callback* de interrupções podem receber argumentos e retornar valores.
- b) É possível executar múltiplos *callbacks* ao mesmo tempo.
- c) Funções de *callback* podem receber argumentos, mas, não podem retornar valores.
- d) Podem ser interrompidas por outras interrupções.
- e) O modificador de variável `volatile` precisa ser utilizado para permitir a modificação de variáveis globais nas funções de *callback* e na função principal.

Questão 3

Quanto a entradas digitais no Arduino, é correto afirmar que:

- a) Esperar um tempo após a leitura de um pino digital — técnica conhecida como **debounce** — é uma boa prática para garantir que o valor lido seja estável.
 - b) A comutação de chaves mecânicas é imune ao aparecimento de ruídos, efeito conhecido como **bouncing**.
 - c) Resistores de *pull-up* fazem com que o estado padrão de uma entrada digital seja nível lógico BAIXO.
 - d) Resistores de *pull-down* fazem com que o estado padrão de uma entrada digital seja nível lógico ALTO.
 - e) Nenhum dos itens anteriores está correto.
-

Questão 4

Raul deseja fazer um LED comutar de estado a cada 5 segundos. Sem muito critério do entendimento viu nos códigos dos slides do prof. Ascânio o seguinte ISR que fazia o LED comutar de estado a cada 5 segundos:

Rotina de interrupção de Overflow associada ao `Timer2`

```
ISR(TIMER2_OVF_vect) {  
    overflows++;  
    if (overflows == 306) {  
        overflows = 0;  
        // Toggle the LED state  
        digitalWrite(13, !digitalRead(13));  
    }  
}
```

Entretanto, ao copiar os códigos, não se atentou ao modo correto de funcionamento e configurou — sem querer — o prescaler do `Timer2` para o fator de 256, através do registrador de controle `TCCR2B` pela seguinte instrução na função `setup()`:

```
TCCR2B = 0b00000110;
```

Com isso, em vez do LED comutar de estado a cada 5 segundos, ele comutava a cada 1.25 segundos (aproximadamente). Qual dos seguintes modos de *prescaling* do `Timer2` faz com que o LED comute de estado a cada 5 segundos, como deseja Raul?

- a) `TCCR2B = 0b00000000;`
 - b) `TCCR2B = 0b00000001;`
 - c) `TCCR2B = 0b00000011;`
 - d) `TCCR2B = 0b00000111;`
 - e) `TCCR2B = 0b00000110;`
-

Questão 5

Acerca de entradas e saídas digitais no Arduino e a conexão de periféricos ao microcontrolador, avalie as seguintes assertivas:

- I. O comando `pinMode(13, OUTPUT)` configura o pino 13 como saída.
- II. Para uma conexão em série do pino+5V do Arduino a um resistor de 330Ω passando pelo ânodo de um LED, em sequência pelo cátodo do LED e finalizando no pino 13 do Arduino, o comando `digitalWrite(13, HIGH)` acenderá o LED se ele estiver apagado.
- III. Para uma conexão em série do pino+5V do Arduino a um resistor de 330Ω passando pelo ânodo de um LED, em sequência pelo cátodo do LED e finalizando no pino 13 do Arduino, o comando `digitalWrite(13, HIGH)` apagará o LED se ele estiver aceso.

IV. O comando `digitalWrite(13, !digitalRead(13))` inverte o estado do dispositivo conectado ao pino 13 qualquer que seja seu modo de conexão (ligado ao +5V ou ao GND do Arduino).

São corretas as assertivas:

- a) I e II, apenas.
- b) I e III, apenas.
- c) I e IV, apenas.
- d) I, III e IV apenas.
- e) I, II, III e IV.

Questão 6

(ENADE 2005 - 11) Apesar de todo o desenvolvimento, a construção de computadores e processadores continua, basicamente, seguindo a arquitetura clássica de von Neumann. As exceções a essa regra encontram-se em computadores de propósitos específicos e nos desenvolvidos em centros de pesquisa. Assinale a opção em que estão corretamente apresentadas características da operação básica de um processador clássico:

- a) Instruções e dados estão em uma memória física única; um programa é constituído de uma seqüência de instruções de máquina; uma instrução é lida da memória de acordo com a ordem dessa seqüência e, quando é executada, passa-se, então, para a próxima instrução na seqüência.
- b) Instruções e dados estão em memórias físicas distintas; um programa é constituído de um conjunto de instruções de máquina; uma instrução é lida da memória quando o seu operando-destino necessita ser recalculado; essa instrução é executada e o resultado é escrito no operando de destino, passando-se, então, para o próximo operando a ser recalculado.
- c) Instruções e dados estão em uma memória física única; um programa é constituído de um conjunto de instruções de máquina; uma instrução é lida da memória quando todos os seus operandos-fonte estiverem prontos e disponíveis; essa instrução é executada e o resultado é escrito no operando de destino, passando-se, então, para a instrução seguinte que tiver todos os operandos disponíveis.
- d) Instruções e dados estão em memórias físicas distintas; um programa é constituído de um conjunto de instruções de máquina; uma instrução é lida da memória quando todos os seus operandos-fonte estiverem prontos e disponíveis; essa instrução é executada e o resultado é escrito no operando de destino, passando-se, então, para a instrução seguinte que estiver com todos os seus operandos disponíveis.
- e) Instruções e dados estão em memórias físicas distintas; um programa é constituído de uma seqüência de instruções de máquina; uma instrução é lida da memória de acordo com a ordem dessa seqüência e, quando é executada, passa-se, então, para a próxima instrução na seqüência.