

EA871 – Laboratório de Programação Básica de Sistemas Digitais

Atividade 04 – 2º semestre de 2023

1. Objetivos

- Explorar os diferentes mecanismos de interrupção disponíveis no microcontrolador ATmega328P para lidar com eventos externos;
- Comparar o mecanismo de interrupção com o de varredura para lidar com eventos assíncronos.

2. Atividade de aula

Exercício 1

Botão acionando LED por meio da interrupção externa INT0

O sistema mostrado na Figura 1 é similar ao utilizado nos exercícios 2 e 3 da Atividade 3. A diferença é que, nesse caso, iremos acionar o LED por meio da interrupção externa INT0. Copie o circuito usando o link <https://www.tinkercad.com/things/32xc0TReF8j> e programe o microcontrolador de modo a mudar o estado do LED quando o botão é pressionado. É necessário tratar o transitório do botão.

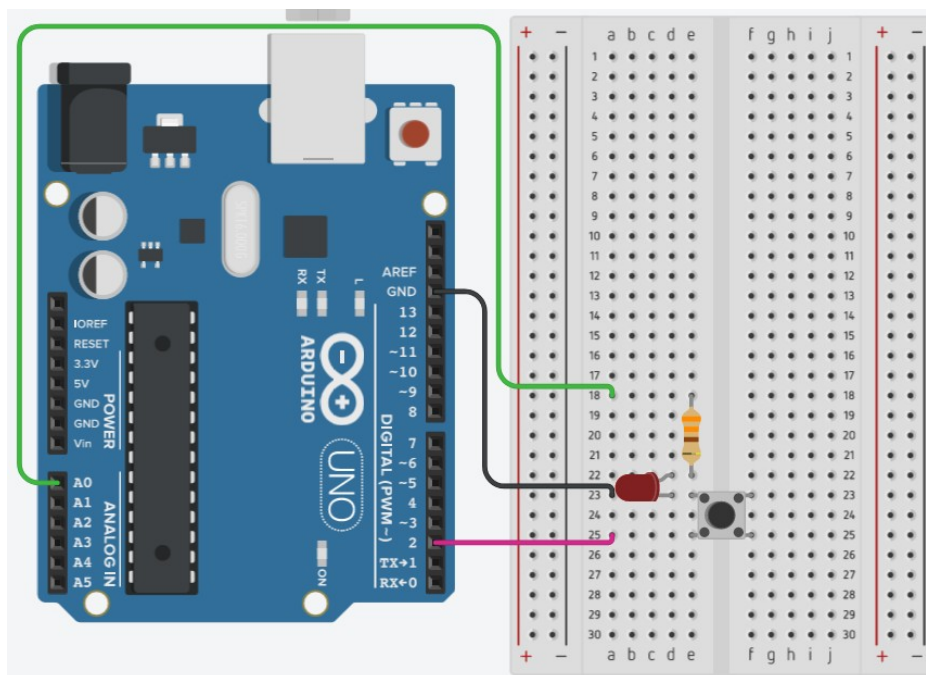


Figura 1: Circuito dos exercícios 1 e 2.

Exercício 2

Botão acionando LED por meio da interrupção externa PCINT

Sem alterar o circuito, repita o exercício anterior utilizando *Pin Change Interrupts*.

3. Roteiro da Atividade para entrega (apenas simulação)

O desafio proposto nesta atividade é a construção de um sistema que estima a frequência de rotação de um motor CC com codificador (encoder). No circuito apresentado na Figura 2, o encoder produz aproximadamente **49 bordas de subida e de descida** (pulsos) no canal B a cada volta do eixo do motor (os pulsos podem ser vistos no osciloscópio). A estimativa da frequência, em rotações por minuto (rpm), deve

ser feita através da contagem do número de pulsos produzidos pelo canal B do encoder durante um intervalo de tempo, através da equação:

$$f_{rpm} = \frac{\text{número de pulsos}}{\text{número de pulsos por volta} * \text{intervalo}},$$

em que o intervalo é medido em minutos.

Para uma contagem precisa do número de pulsos, utilize a interrupção **INT1** associada ao terminal 3 da placa. Além disso, utilize a função `_delay_ms()` da biblioteca `util/delay.h` para definir o intervalo de tempo da contagem de bordas. Como parte obrigatória da atividade, é necessário criar uma variável no código que estime a frequência do motor em rpm. O algoritmo correspondente à centena da estimativa deve ser exibido no display de sete segmentos, ou seja:

1. O display exibe 0, se $f_{rpm} < 100$;
2. O display exibe 1, se $100 \leq f_{rpm} < 200$;
3. O display exibe 2, se $200 \leq f_{rpm} < 300$;
4. O display exibe 3, se $300 \leq f_{rpm} < 400$;
5. O display exibe 4, se $400 \leq f_{rpm} < 500$;
6. O display exibe 5, se $500 \leq f_{rpm} < 600$;
7. O display exibe 6, se $600 \leq f_{rpm} < 700$;
8. O display exibe 7, se $700 \leq f_{rpm} < 800$;
9. O display exibe 8, se $800 \leq f_{rpm} < 900$;
10. O display exibe 9, se $f_{rpm} \geq 900$.

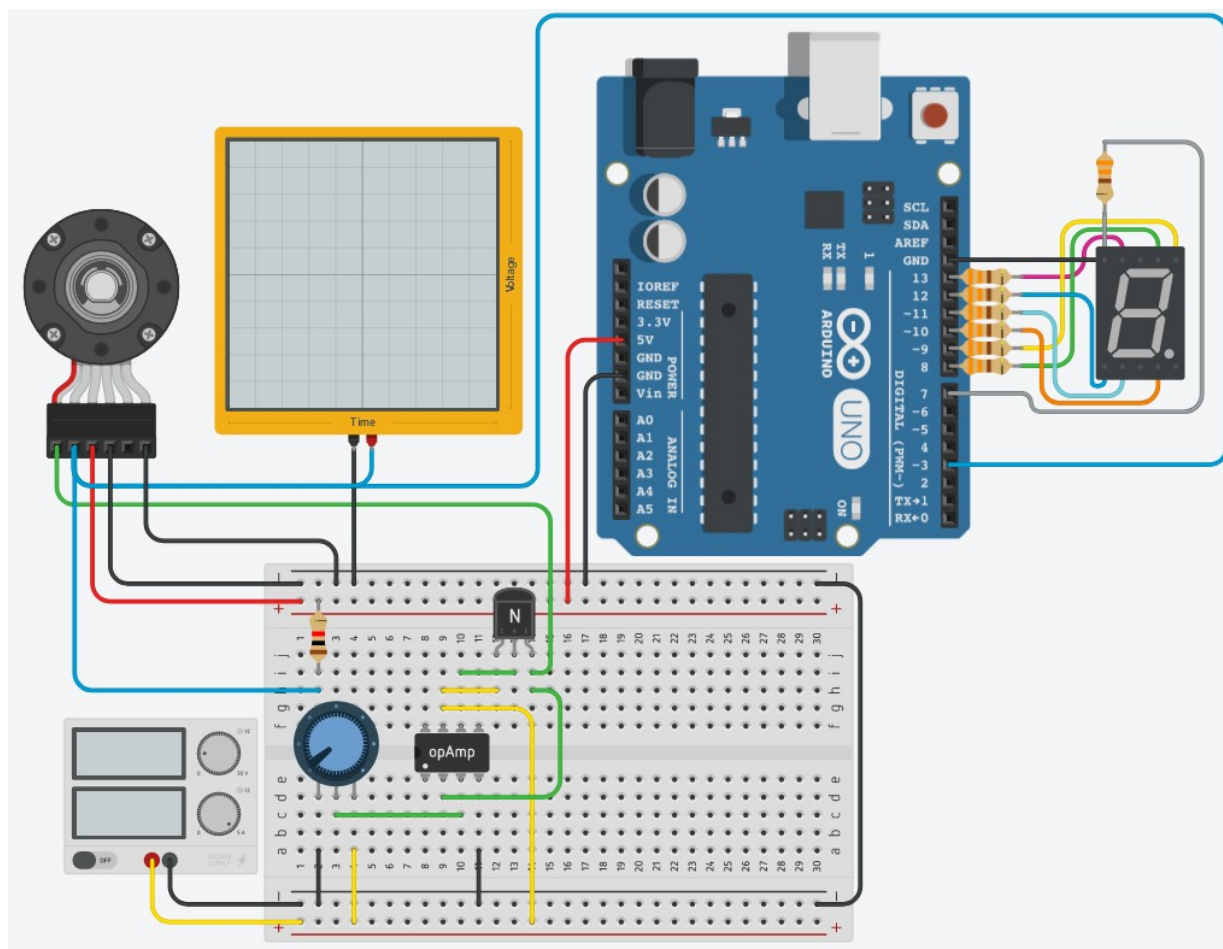


Figura 2: Montagem do sistema.

Durante a simulação, a frequência de rotação do motor pode ser alterada através do potenciômetro presente no protoboard. A frequência é mostrada no motor e deve ser usada para verificar a precisão da estimativa obtida pelo sistema.

Vamos, ademais, explorar os recursos oferecidos pela biblioteca **interrupt.h** para definir a rotina de serviço de interrupção em linguagem C (<https://onlinedocs.microchip.com/pr/GUID-317042D4-BCCE-4065-BB05-AC4312DBC2C4-en-US-2/GUID-F889605B-692F-493A-8BE7-F0FBACF1715B.html>).

Importante: Por convenção, vamos empregar os pinos 8 a 13 para acionar os segmentos de *a* a *f* e o terminal 7 da placa para acionar o segmento *g* do display. Utilizem o diagrama de pinos do Arduino UNO, disponibilizado na página do curso para determinar as portas envolvidas. A interrupção INT1, por sua vez, deverá ser configurada para detectar os eventos apropriados no terminal 3 do Arduino UNO. Notem que, diferentemente da Atividade 3, o display de sete segmentos desta atividade é catodo comum, ou seja, os segmentos são acesos com nível alto na porta à qual estão conectados e apagados com nível 0. O link para o template do Tinkercad, mostrado na figura da montagem do sistema, é <https://www.tinkercad.com/things/gV6tHmfc0jb>

Materiais de consulta

Para esta atividade, é recomendado consultar o material sobre interrupções externas e os vídeos de apoio fornecidos na página do curso.

Instruções para a submissão do trabalho

- 1) Nos comentários do código-fonte, explique o funcionamento do programa, justifique as operações e os valores carregados em todos os registradores. Programas sem comentários terão nota máxima 4,0, mesmo que funcionem.
- 2) Ao final da atividade, salve o código-fonte com nome “seu_ra.txt” (Exemplo: 025304.txt).
- 3) Faça o upload da sua solução da atividade 4 no Google Classroom.