

ROBÓTICA

Módulo 2



Semáforo Inteligente com IR

AULA 04

GOVERNADOR DO ESTADO DO PARANÁ

Carlos Massa Ratinho Júnior

SECRETÁRIO DE ESTADO DA EDUCAÇÃO

Renato Feder

DIRETOR DE TECNOLOGIA E INOVAÇÃO

Andre Gustavo Souza Garbosa

COORDENADOR DE TECNOLOGIAS EDUCACIONAIS

Marcelo Gasparin

Produção de Conteúdo

Cleiton Rosa

Simone Sinara de Souza

Validação de Conteúdo

Cleiton Rosa

Revisão Textual

Adilson Carlos Batista

Projeto Gráfico e Diagramação

Edna do Rocio Becker

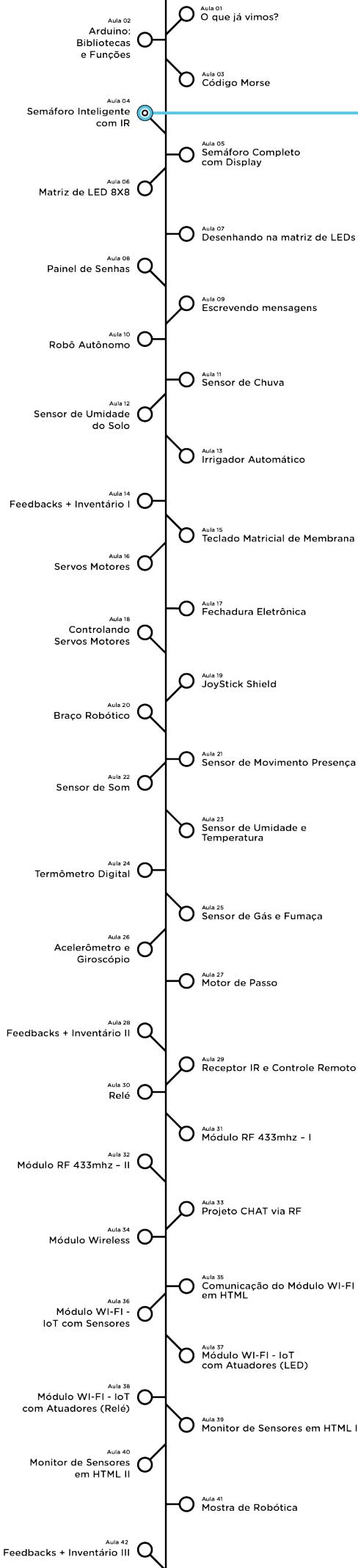
Ilustração

Jocelin Vianna

2021



Este trabalho está licenciado com uma Licença Creative Commons
Atribuição NãoComercial - Compartilhamento 4.0 Internacional



Aula 03 Código Morse

Aula 04 Semáforo Inteligente com IR

Aula 05 Semáforo Completo com Display

Sumário

Introdução	2
Objetivos desta Aula	2
Competências Gerais Previstas na BNCC	3
Habilidades do Século XXI a Serem Desenvolvidas	4
Roteiro da Aula	5
1. Contextualização	4
2. Montagem e Programação	6
3. Feedback e Finalização	16
Videotutorial	17



Introdução

No módulo 1, identificamos que os semáforos são dispositivos de sinalização que informam aos usuários de trânsito - sejam motoristas, pedestres, ciclistas, entre outros - o momento permitido à circulação, à espera e à transição entre uma ou outra circunstância.

Nesta aula, você terá a oportunidade de conhecer o funcionamento do semáforo inteligente que informa ao pedestre quando a travessia na faixa é segura.



Objetivos desta Aula

- Entender o funcionamento de um semáforo inteligente;
- Programar o semáforo inteligente, através de funções, para executar a informação ao pedestre.





Competências Gerais Previstas na BNCC

[CG02] - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

[CG04] - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

[CG05] - Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.

[CG09] - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

[CG10] - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.





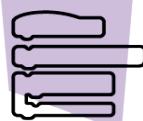
Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



Lista de Materiais

- 01 Placa Protoboard;
- 01 Placa Arduino Uno R3;
- 01 Cabo USB;
- 09 Jumpers Macho-Macho;
- 05 Resistores 220 Ohms;
- 02 LEDs vermelho 5mm;
- 02 LEDs verde 5mm;
- 01 LED amarelo 5mm;
- Notebook;
- Software Arduino IDE;
- 01 Sensor de Obstáculo IR.



Roteiro da Aula

1. Contextualização (15min):

Frequentemente, ouvimos no noticiário casos de acidentes de trânsito nos quais os pedestres são atropelados quando atravessavam a faixa de segurança, a justificativa dos condutores é que não conseguiram frear a tempo.

Para tentar reduzir estes acidentes, engenheiros de tráfego terrestre buscaram soluções no formato de semáforo inteligente, o qual, através de sensor infravermelho, identifica a presença do pedestre e aciona o fechamento do semáforo para os carros.

Mas de que forma o sensor infravermelho detecta a presença dos pedestres?

Na **Aula 30 - Sensor de Obstáculos IR**, Módulo 1, comentamos que este modelo de sensor detecta, através de radiação infravermelha, objetos e delimitações, em um raio de 2 a 30 centímetros. Seu funcionamento é simples e ocorre através de dois LEDs, o **emissor**, que permanece constantemente enviando radiação infravermelha e, ao encontrar um obstáculo em seu raio de ação, reflete um sinal que é detectado pelo outro LED, o **fotorreceptor**. Esse, por sua vez, identifica a presença do objeto. Para relembrar o funcionamento do Sensor de Obstáculo IR, retome a aula.

Nesta aula, apresentaremos um protótipo de semáforo inteligente para o uso em vias públicas secundárias que emprega sensor de obstáculo infravermelho para detectar a presença de pedestres. Seu funcionamento ocorrerá da seguinte maneira: quando uma pessoa se posicionar à frente do sensor, o sinal infravermelho será refletido e detectado pelo receptor, o qual emitirá um sinal para o fechamento do semáforo de carros.

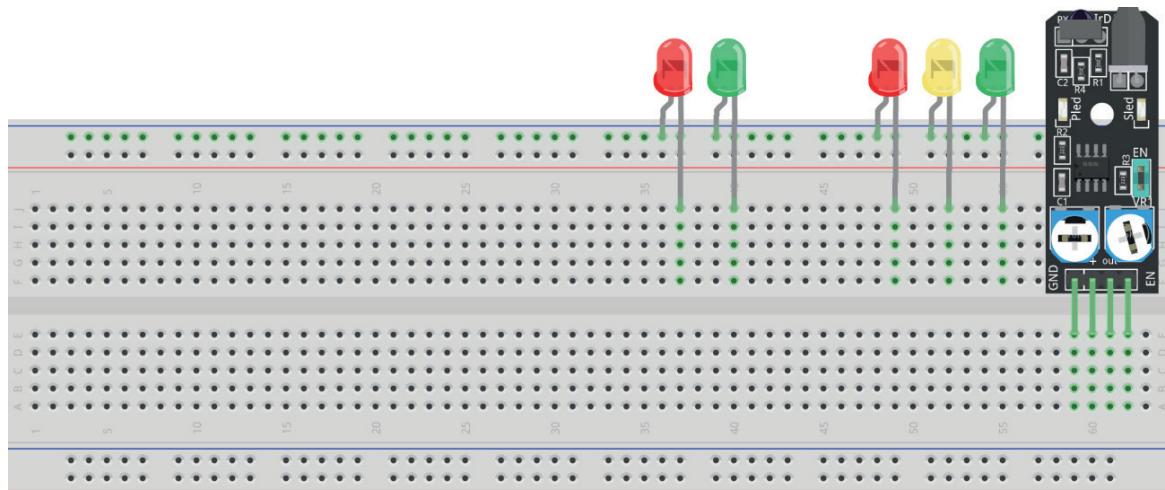


Uma vez compreendida a importância do semáforo inteligente para a travessia dos pedestres, você tem como missão montar o protótipo deste semáforo e realizar a programação para seu correto funcionamento.

2. Montagem e Programação (60min):

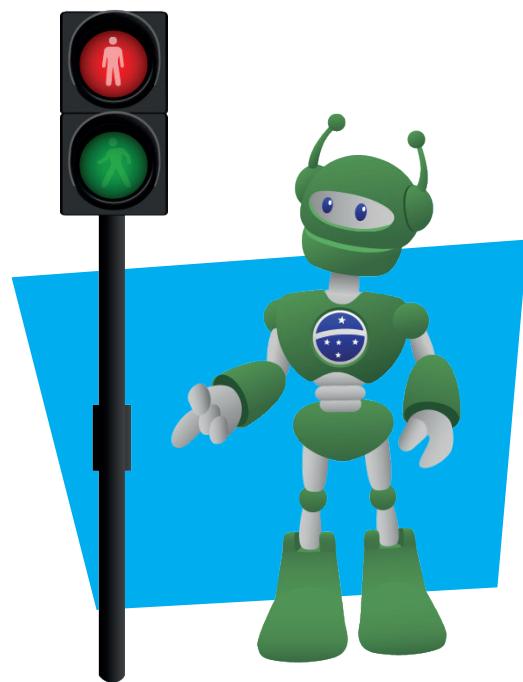
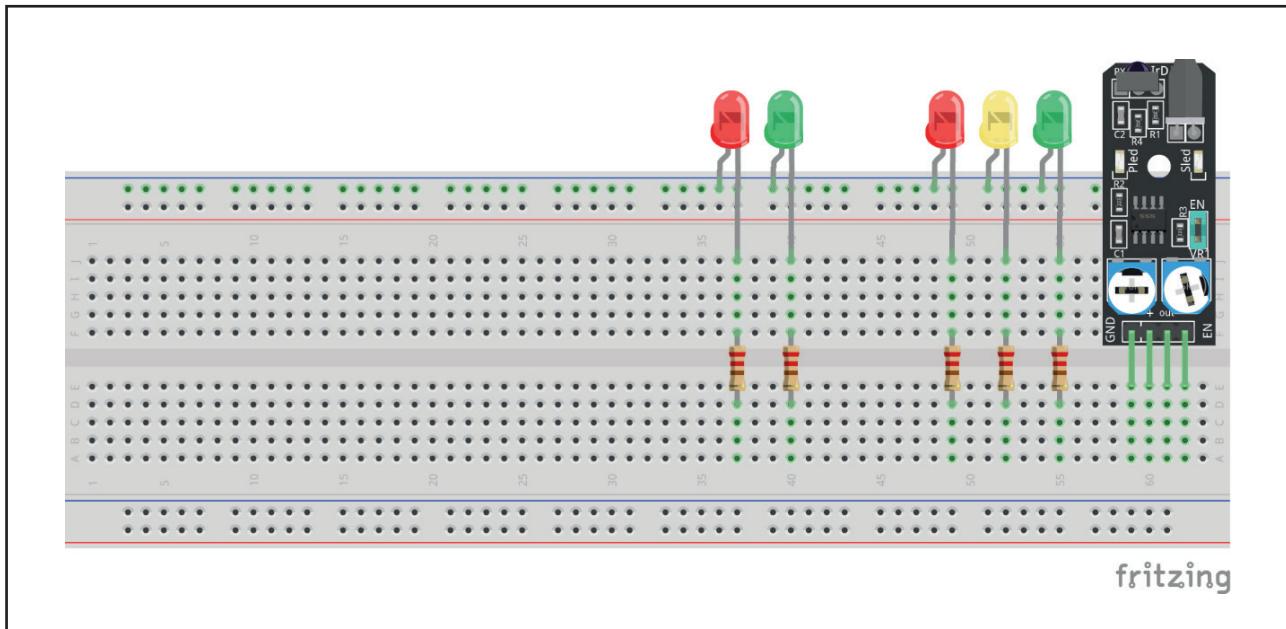
Inicie a montagem dos componentes eletrônicos encaixando na Protoboard, o Sensor de Obstáculo IR e os LEDs, conforme as posições indicadas pela figura 1.

Figura 1 – Inserindo os componentes eletrônicos na Protoboard



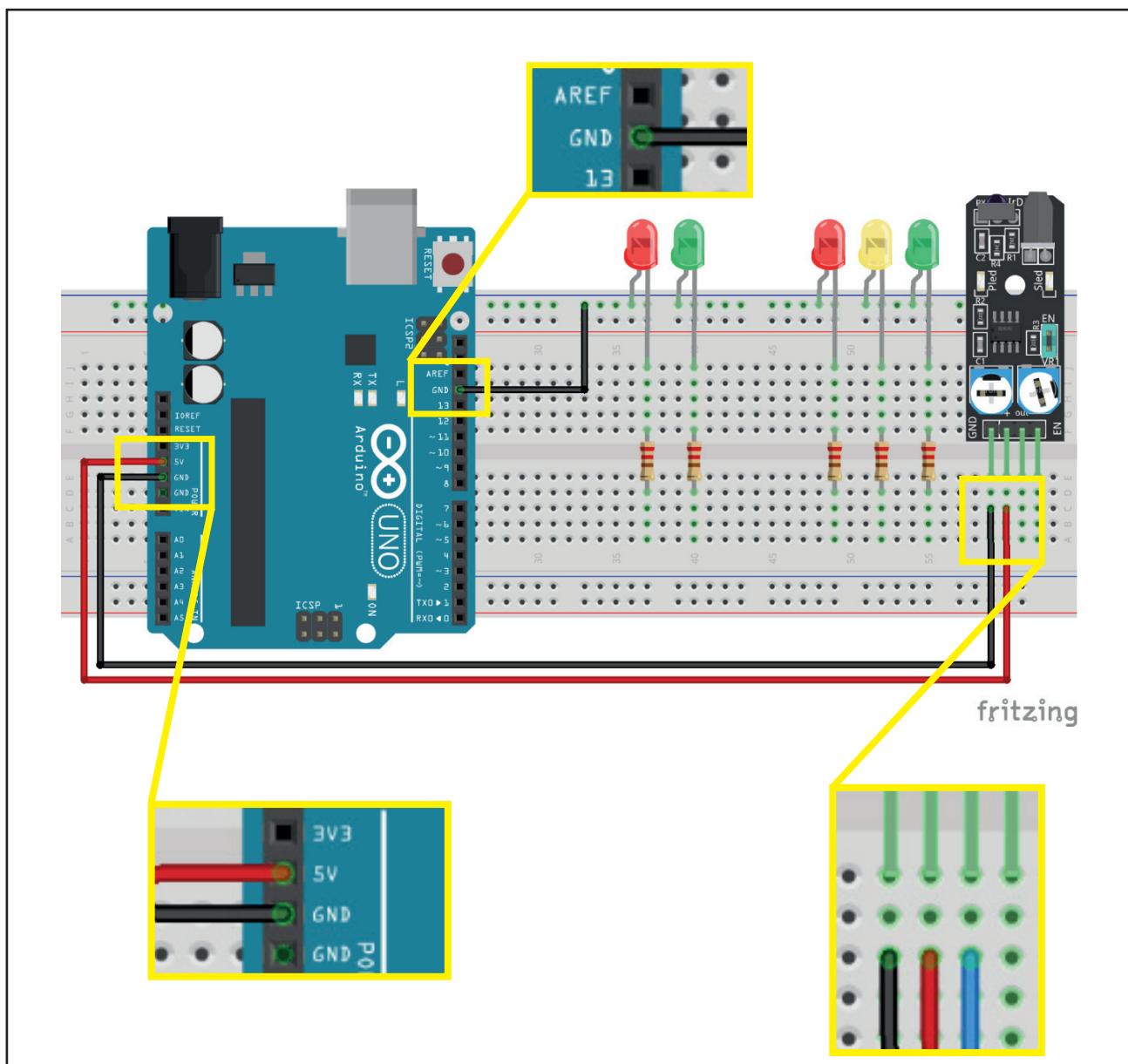
Conecte 5 resistores de 220 Ohms aos terminais positivos dos LEDs, como mostra a figura 2.

Figura 2 - Conectando resistores na Protoboard



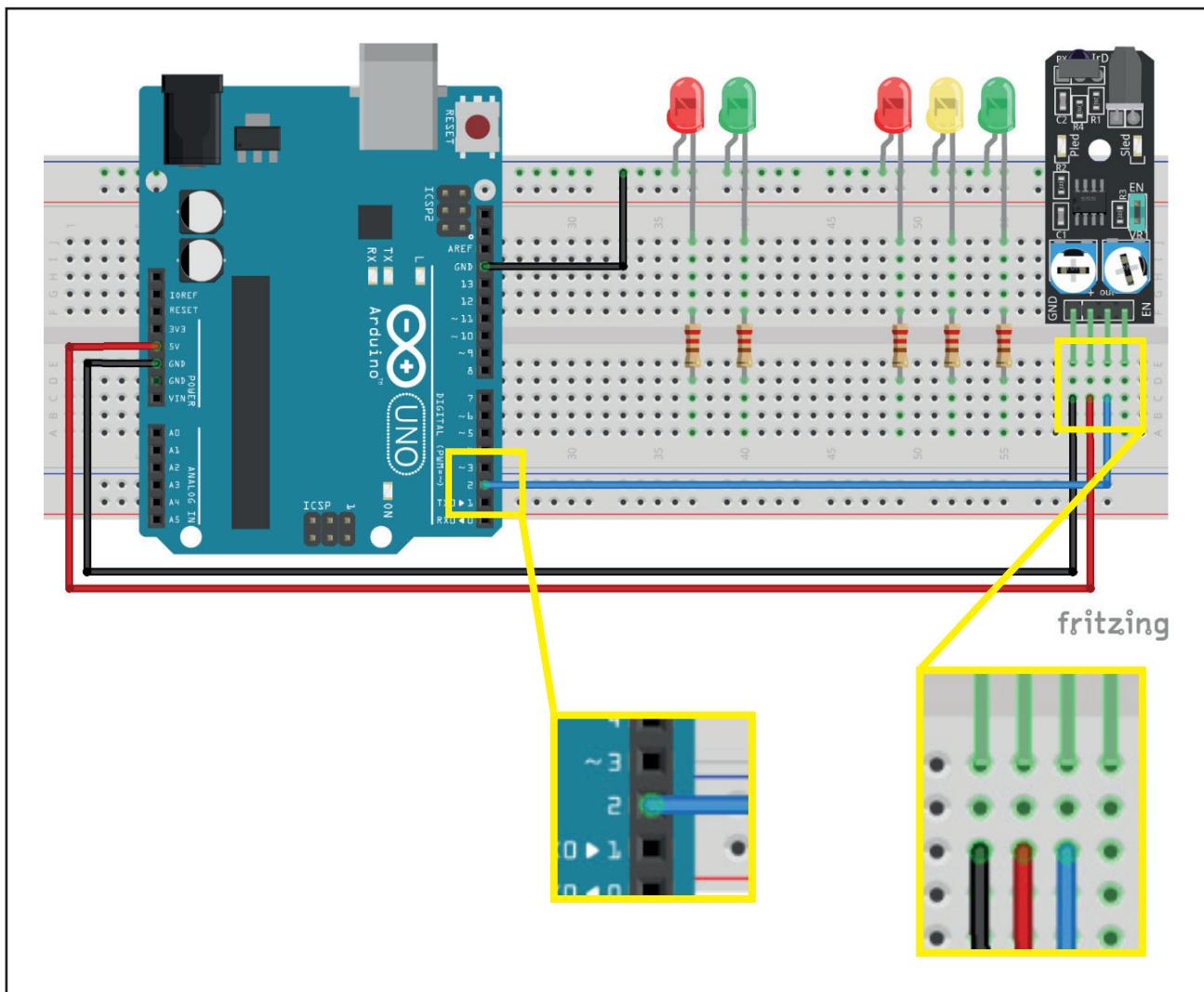
Interligue, através de 1 jumper Macho-Macho, um pino GND da placa Arduino com um dos furos da linha lateral azul da Protoboard (aquele que os LEDs estão interligados). Conecte mais 2 jumpers entre os pinos 5V e GND do Arduino com os pinos (+) e GND do sensor de obstáculo IR, conforme apresentado na figura 3.

Figura 3 - Conectando a placa Arduino à placa Protoboard e os componentes eletrônicos



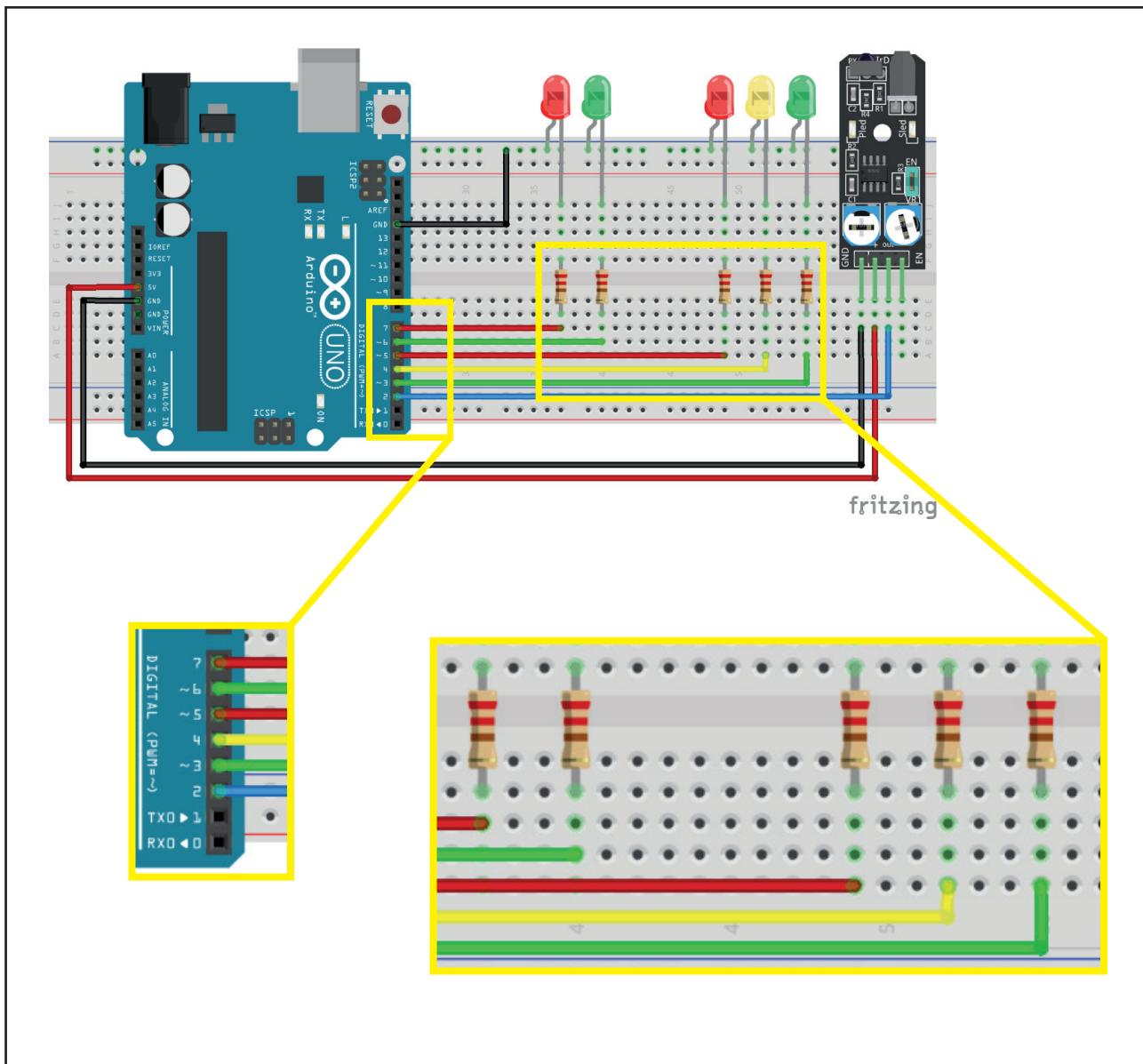
Utilizando mais 1 jumper, conecte o pino digital 2 do Arduino ao pino Sinal do sensor, como indicado na figura 4.

Figura 4 - Conectando o pino de dados do sensor ao Arduino



Finalizando a montagem, utilize mais 5 jumpers para conectar os 5 LEDs aos pinos digitais 3, 4, 5, 6 e 7 do Arduino, como indicado na figura 5.

Figura 5 – Conectando os LEDs aos pinos digitais do Arduino





Agora, vamos programar!

Com os componentes eletrônicos montados, vamos programar por codificação o funcionamento do semáforo inteligente para pedestres.

Linguagem de programação por código

Para iniciar a programação, conecte a placa Arduino ao computador, através de um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

No software IDE, crie um sketch e lembre-se de selecionar a porta que o computador atribuiu ao Arduino; então, escreva ou copie e cole o código-fonte de programação, conforme apresentado no quadro 1:

Atenção!

Ao copiar o código diretamente do pdf, evite quebra da página (e consequentemente erros na compilação), copiando o código por partes.



Quadro 1- Programação por codificação do semáforo inteligente

```
/* Programa: Semáforo inteligente. */  
  
/* Definições dos pinos para os LEDs e sensor. */  
#define PIN_SENSOR 2  
#define PIN_LED_VERDE_CAR 3  
#define PIN_LED_AMARELO_CAR 4  
#define PIN_LED_VERMELHO_CAR 5  
#define PIN_LED_VERDE_PED 6  
#define PIN_LED_VERMELHO_PED 7
```

```
int modo = 0; /* Variável que seleciona o modo de operação do
semáforo 0: Modo aberto para carros e 1: Modo travessia de
pedestres. */
void setup()
{
    /* Configura o pino do sensor como ENTRADA. */
    pinMode(PIN_SENSOR, INPUT);
    /* Loop para configurar os pinos 3 a 7 como SAÍDA. */
    for (int i = 3; i <= 7; i++) {
        pinMode(i, OUTPUT);
    }
    /* Ativa a função de interrupção no pino que o sensor está
ligado. */
    attachInterrupt(digitalPinToInterrupt(PIN_SENSOR),
sensorAcionado, FALLING);
}
void loop()
{
    /* Switch com dois modos de operação: modo aberto para
carros e modo travessia de pedestre. Enquanto o sensor
não detectar um pedestre, o modo de operação permanece
em 0, ou seja, aberto para a passagem dos carros. Caso o
sensor detecte um pedestre, a função attachInterrupt irá
imediatamente alterar o modo de operação para 1, ou seja,
entrará no modo de travessia dos pedestres. */
    switch (modo) {
        case 0:
            modocarros();
            break;
        case 1:
            travessia();
            break;
    }
}
```

```
/* Após o sensor detectar o pedestre, será executada essa
função, ela altera o modo de operação do semáforo. */
void sensorAccionado() {
    modo = 1;
}

/* Função que ativa os LEDs para o modo "Travessia de
pedestres" */
void travessia() {
    /* Aguarda +1 segundo após o sensor detectar o pedestre */
    delay(1000);

    /* Aciona o amarelo do semáforo dos carros */
    digitalWrite(PIN_LED_VERDE_CAR, LOW);
    digitalWrite(PIN_LED_AMARELO_CAR, HIGH);
    delay(3000);
    /* Abre o semáforo de pedestres. */
    digitalWrite(PIN_LED_VERDE_PED, HIGH);
    digitalWrite(PIN_LED_VERMELHO_PED, LOW);
    digitalWrite(PIN_LED_AMARELO_CAR, LOW);
    digitalWrite(PIN_LED_VERMELHO_CAR, HIGH);
    delay(5000);
    /* Apaga o verde dos pedestres e começa a piscar o
vermelho, indicando que o tempo de travessia está acabando.
*/
    digitalWrite(PIN_LED_VERDE_PED, LOW);
    for (int i = 0; i <= 5; i++)
    {
        digitalWrite(PIN_LED_VERMELHO_PED, HIGH);
        delay(300);
        digitalWrite(PIN_LED_VERMELHO_PED, LOW);
        delay(300);
    }
}
```

```
modo = 0; /* Após a travessia de pedestre, altera para o
modo de passagem de carros: 0. */
}
/* Função que ativa os LEDs para o modo "Aberto para carros"
*/
void modocarros() {
    digitalWrite(PIN_LED_VERDE_PED, LOW);
    digitalWrite(PIN_LED_VERMELHO_PED, HIGH);
    digitalWrite(PIN_LED_VERDE_CAR, HIGH);
    digitalWrite(PIN_LED_AMARELO_CAR, LOW);
    digitalWrite(PIN_LED_VERMELHO_CAR, LOW);
}
```



Com o código-fonte inserido no Arduino IDE, compile o programa pressionando o botão **Verificar** para examinar se não há erros de sintaxe. Estando o código correto, pressione o botão **Carregar** para realizar a transferência do programa para o Arduino.

Após a transferência do programa, o protótipo do semáforo inteligente deve entrar em funcionamento, detectando a presença do pedestre e acionando o fechamento do semáforo de carros. Perceba

que, ao aproximar um objeto do sensor, haverá mudança na sequência de acendimento dos LEDs. O LED verde do semáforo de pedestres acenderá, liberando a via para a travessia desses.



Desafio:

Que tal inserir no protótipo o componente Buzzer passivo para sonorizar a abertura e fechamento do semáforo inteligente?



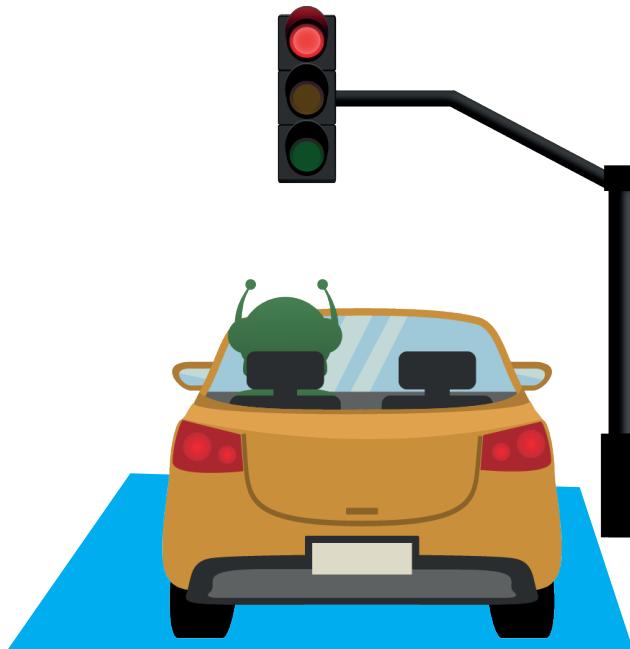
E se...?

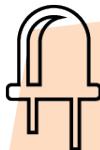
O projeto não funcionar, se atente a alguns dos possíveis erros:

- a.** Verifique se os jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim a conexão;
- b.** Verifique se os jumpers estão ligados aos pinos corretos no Arduino;
- c.** Verifique se os LEDs estão conectados corretamente e não invertidos;
- d.** Verifique se a programação está adequada a cada porta digital.

3. Feedback e Finalização (15min):

- a.** Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.
- b.** Analise seu projeto desenvolvido, de modo a atender aos requisitos para funcionamento de um semáforo: sequência correta do acendimento das cores verde, amarelo e vermelho para os veículos e, de modo simultâneo, para os dois semáforos.
- c.** Reflita se as seguintes situações ocorreram:
 - i.** Colaboração e Cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?
 - ii.** Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?
- d.** Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.





Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/a04robotica2>

Acesse, também, pelo QRCode:



