



DISPOSITIVO AUTÔNOMO PARA MAPEAMENTO E SOLUÇÃO DE LABIRINTOS UTILIZANDO ESP8266 E ALGORITMO DE BUSCA EM LARGURA

Nome: João Gonçalves

Orientador: Me. Marcelo Christiano da França Júnior



Estrutura da apresentação

1 Problema de pesquisa

2 Introdução

3 Revisão de literatura

4 Metodologia

5 Resultados

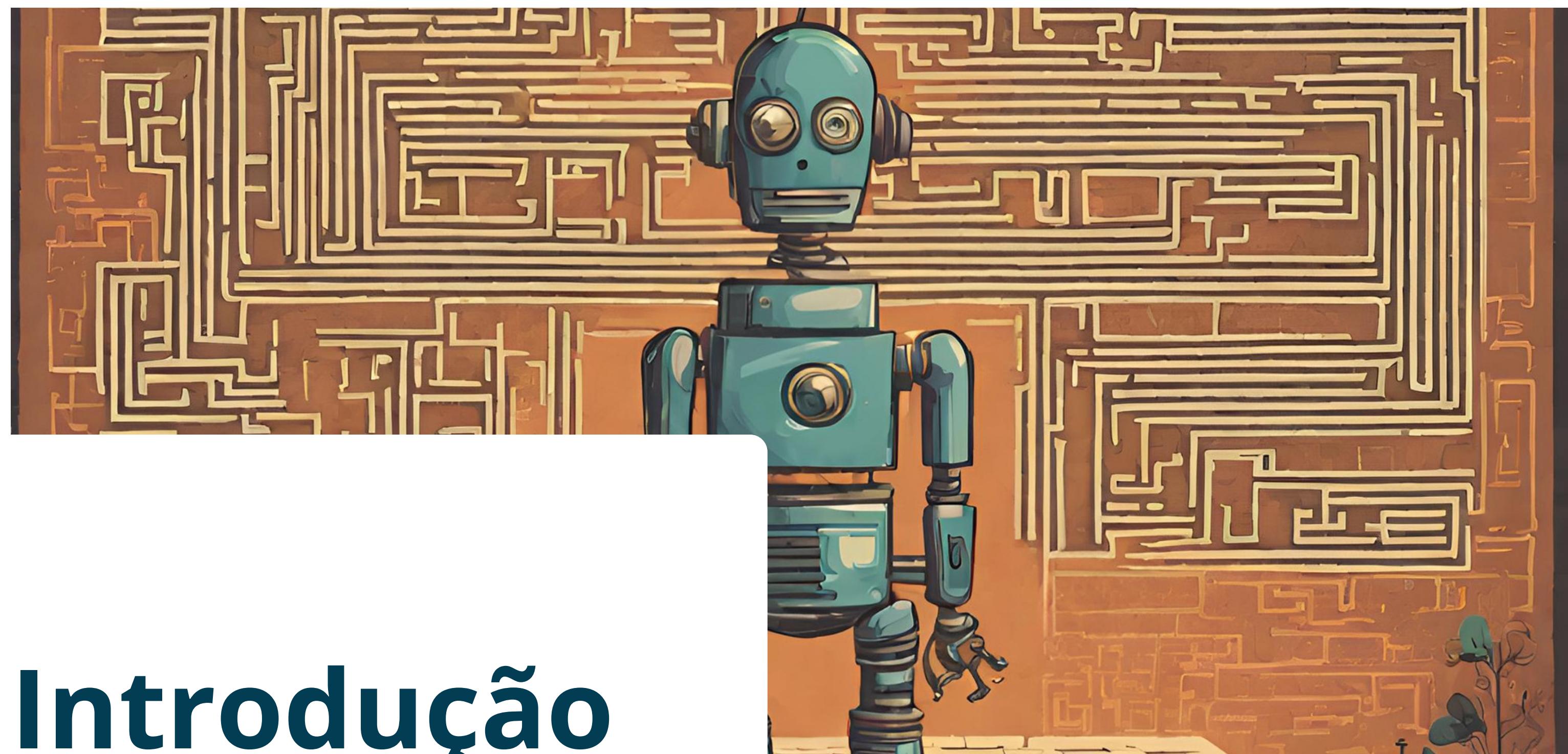
6 Conclusão





Problema de pesquisa

Como percorrer um caminho sem ter conhecimento do percurso, utilizando ESP8266 e algoritmos de busca em largura?



Introdução

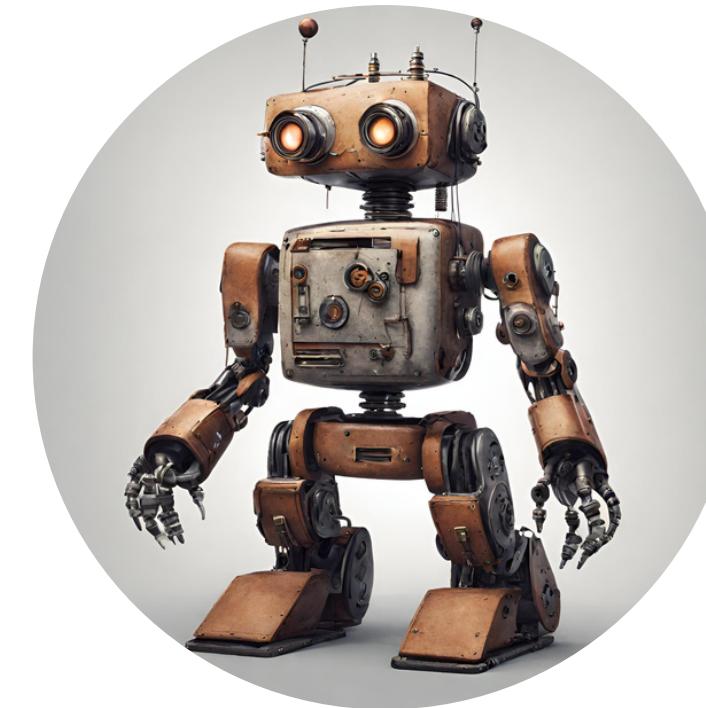


Contextualização



Tipos de Labirintos

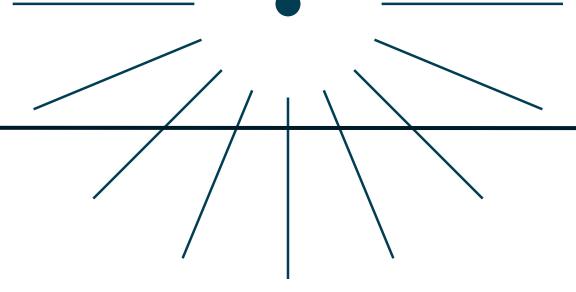
Meandros e maze



Robô

Rabota e robotnik

Justificativa



O ato de resolver um labirinto, para muitas pessoas, pode parecer algo muito complexo e trabalhoso, pois requer muito tempo e esforço.

Percorrer de um ponto A para um ponto B sem possuir conhecimento do caminho

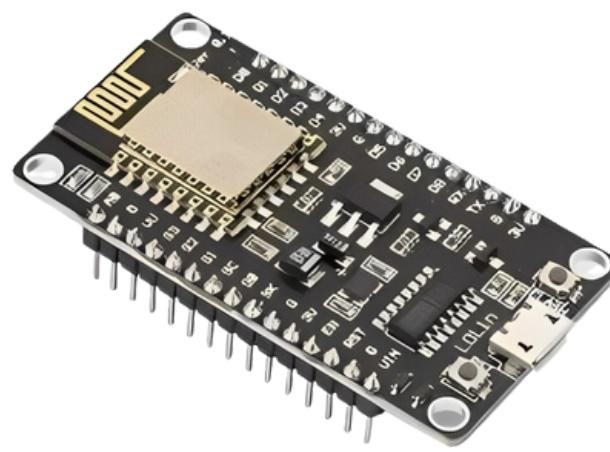




Objetivo Geral

Desenvolver um protótipo de um dispositivo autônomo para mapeamento e solução de labirintos utilizando ESP8266 e algoritmos de busca em largura.

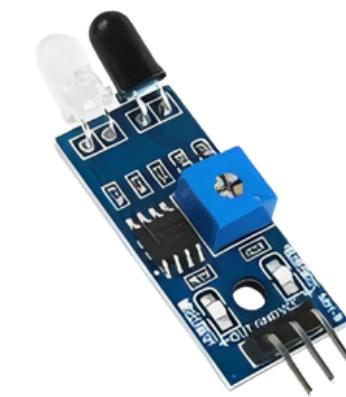
Objetivos específicos



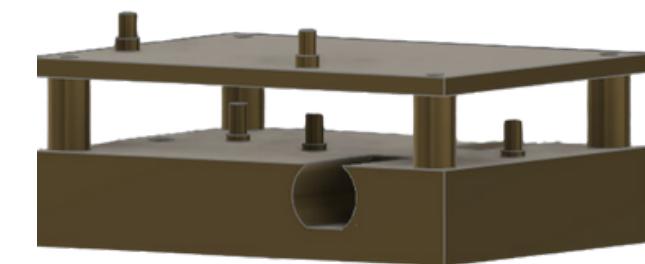
ESP8266



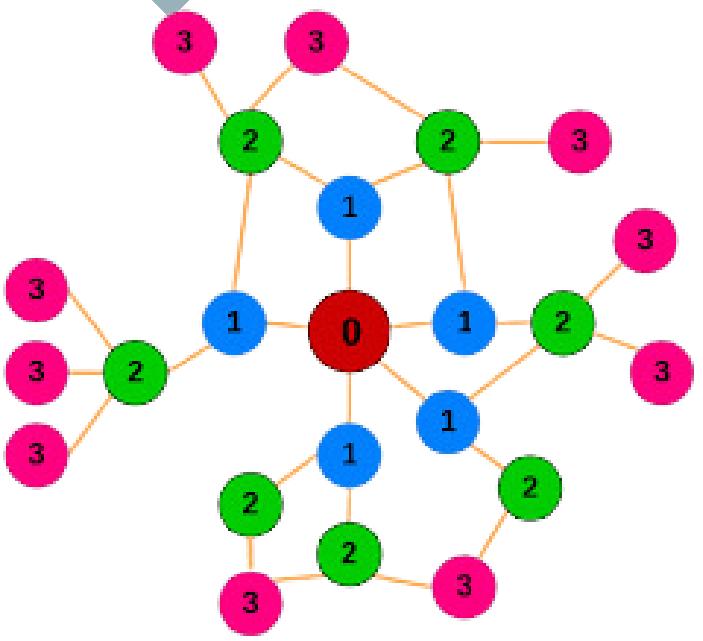
Motores



Sensor



Protótipo



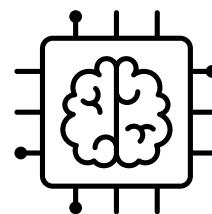
Algoritmo

Revisão da literatura

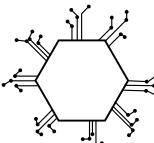


Tópicos abordados

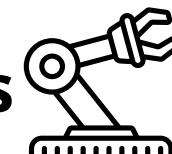
01 Microcontrolador



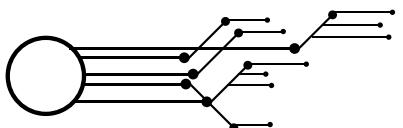
02 Sensores



03 Atuadores

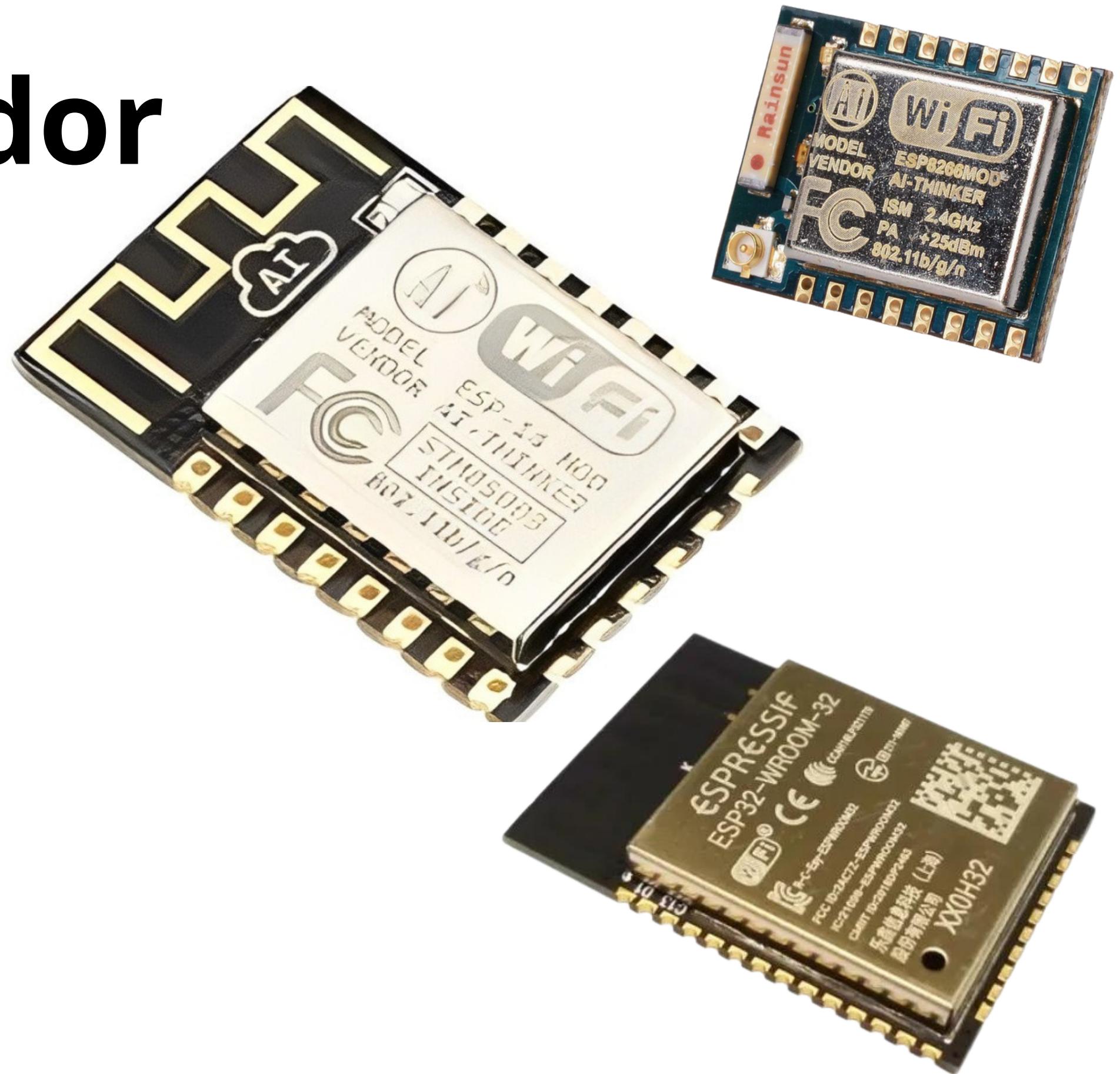


04 Algoritmo de busca



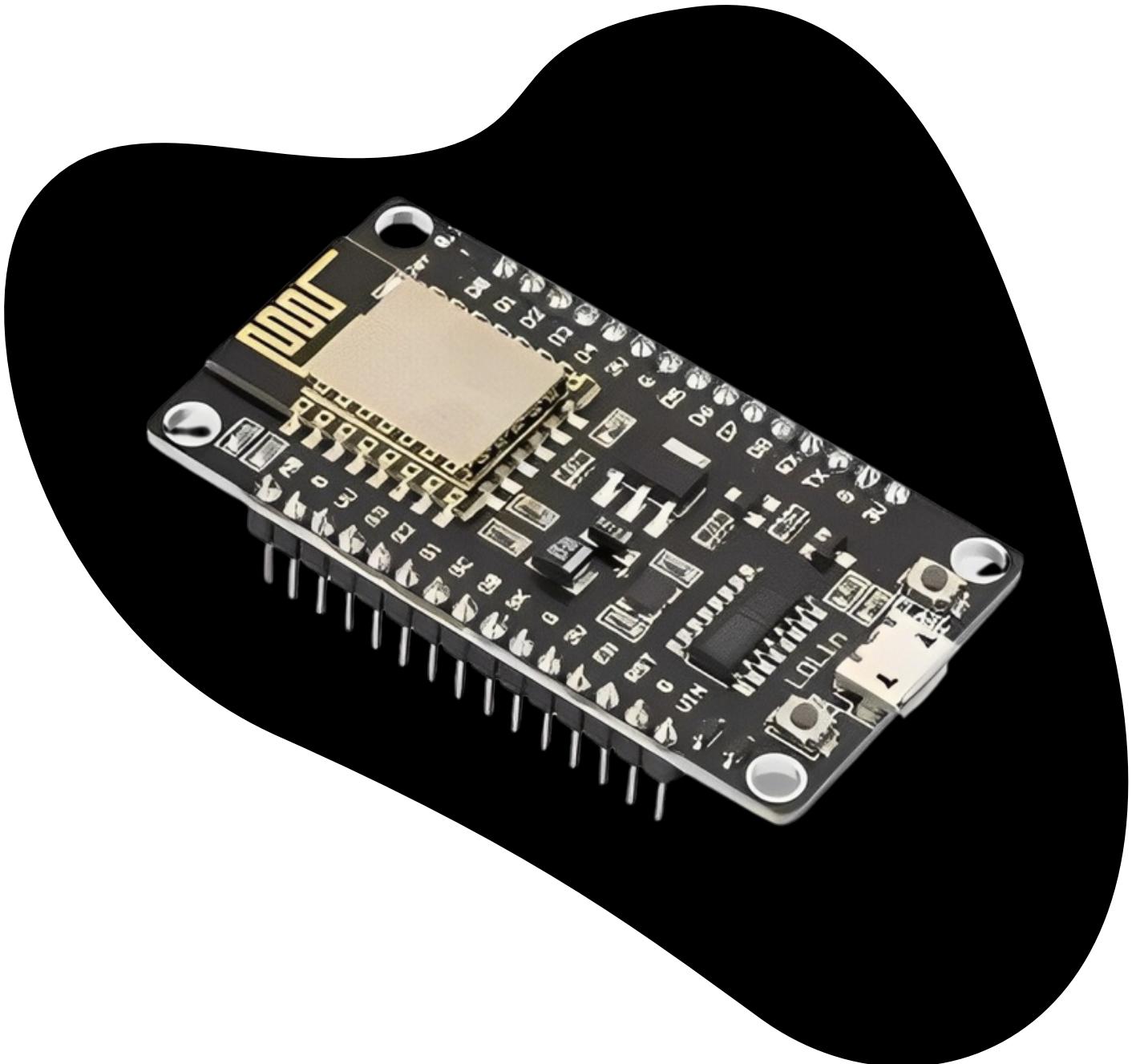
Microcontrolador ESP8266

Produzido pela fabricante chinesa Espressif, é um microcontrolador de 32 bits que funciona na frequência padrão de 80 MHz, podendo chegar a 160 MHz. É possível encontrar diversos módulos deste microcontrolador ESP-12E, ESP-07, ESP32, entre outros.



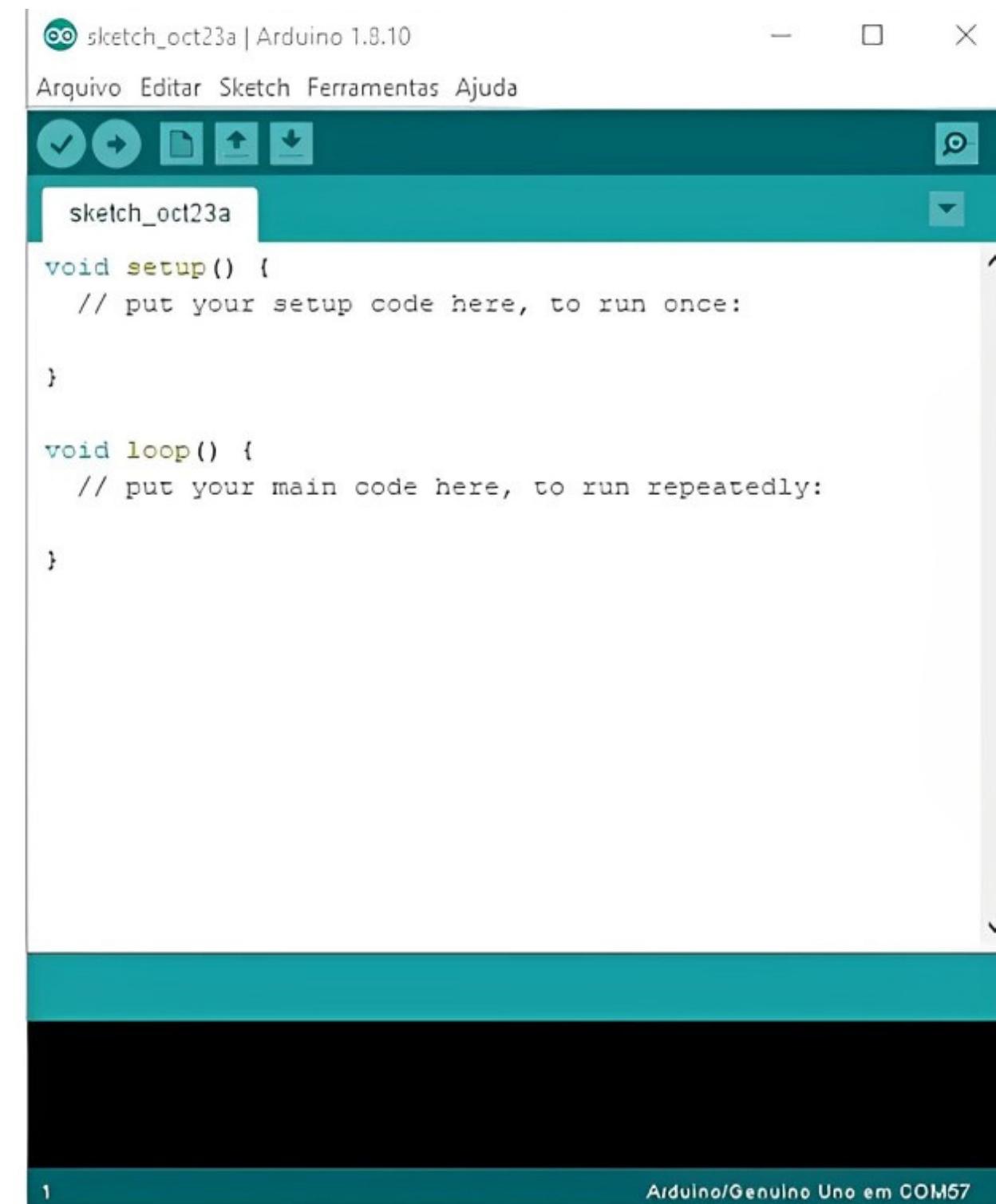
NodeMCU

Plataforma open source da família ESP8266 criada para ser utilizada no desenvolvimento de projetos IoT (Internet of Things) e também surgiu para facilitar a construção de projetos utilizando o ESP8266.



Programação

Uma das maneiras de programar o NodeMCU é através da utilização da IDE do Arduino, esta IDE é open source, através desse ambiente podemos usar bibliotecas que utilizam a linguagem C++ entre outras ferramentas fornecidas pelo ambiente.



The screenshot shows the Arduino IDE interface. The title bar reads "sketch_oct23a | Arduino 1.8.10". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". Below the menu is a toolbar with icons for save, upload, and search. The code editor window contains the following C++ code:

```
void setup() {
  // put your setup code here, to run once:

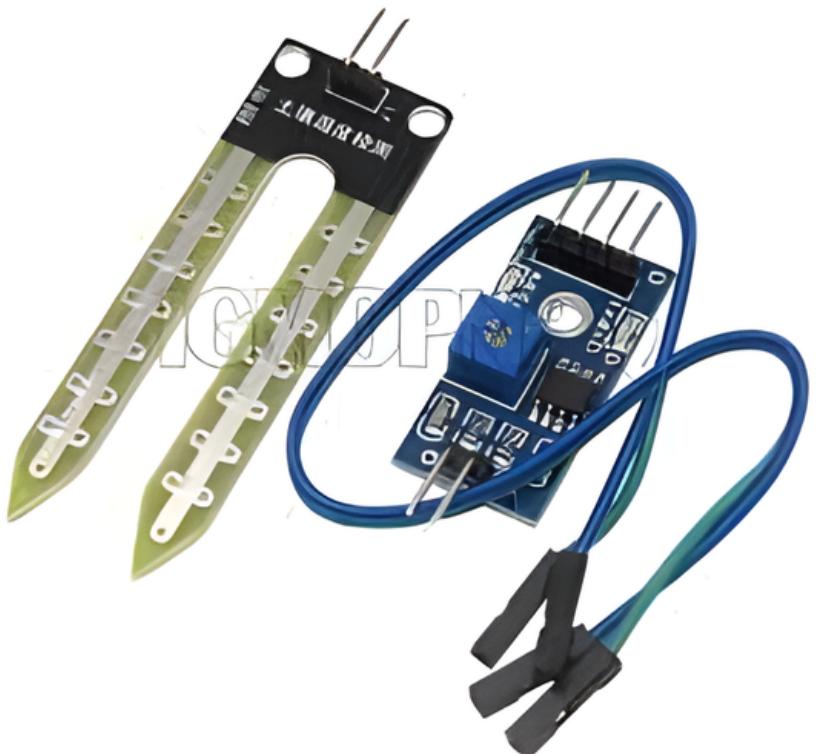
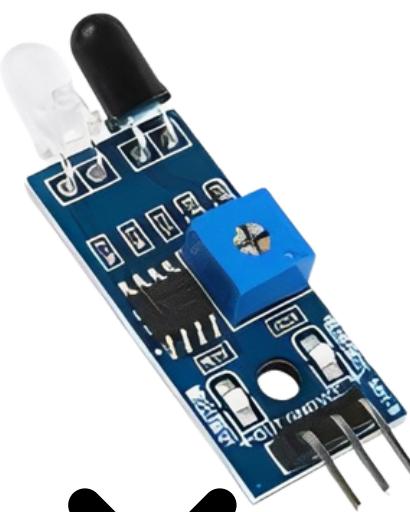
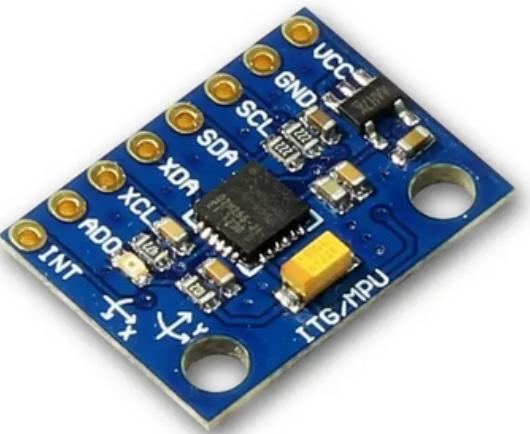
}

void loop() {
  // put your main code here, to run repeatedly:

}
```

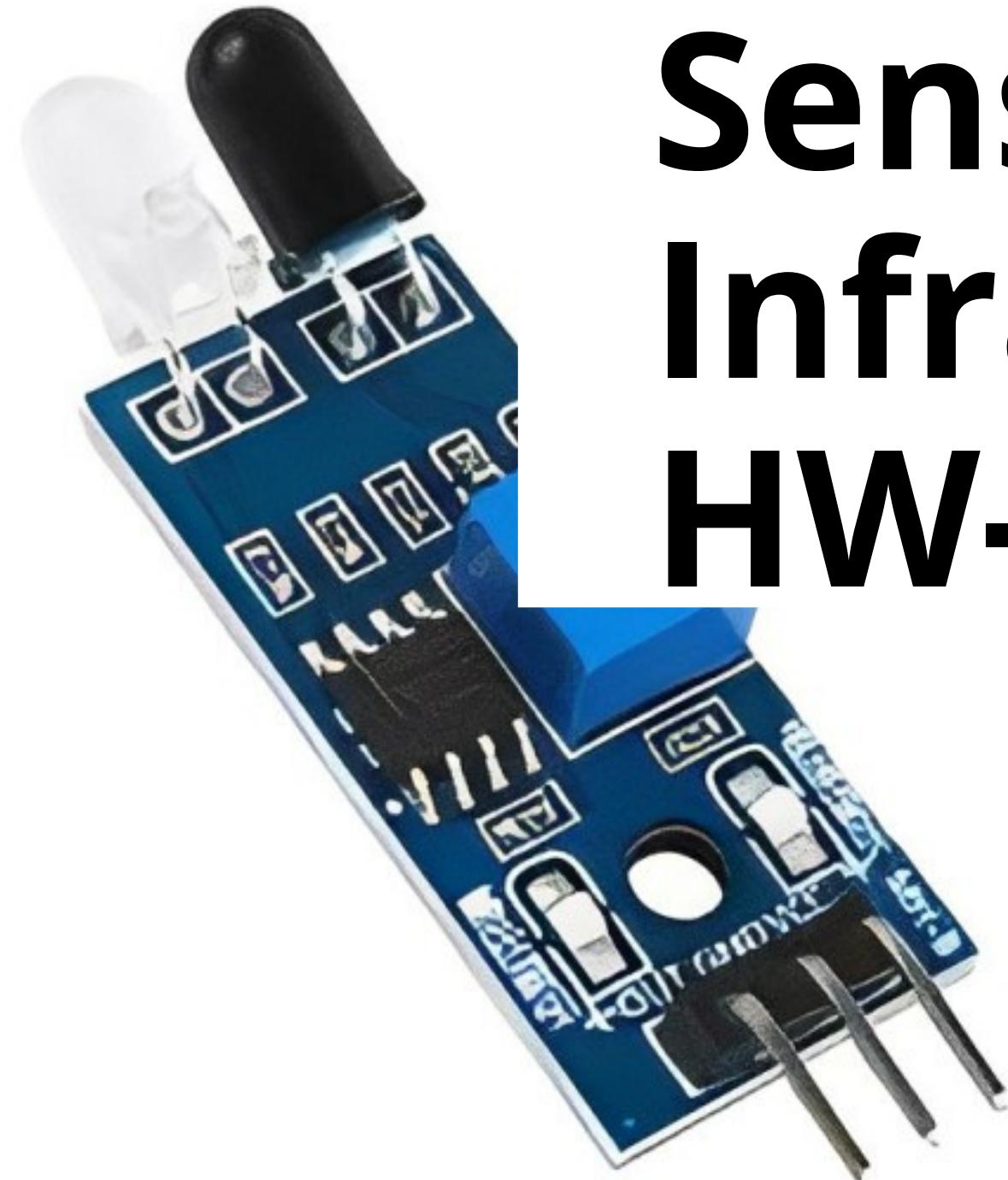
The status bar at the bottom right shows "Arduino/Genuino Uno em COM57".

Sensores



Sensores são dispositivos eletrônicos utilizados para obtenção de dados, efetuando a função de medir algum estímulo recebido pelo ambiente externo.

Sensor reflexivo Infravermelho HW-201



Módulo eletrônico capaz de detectar objetos que estejam em sua frente. Por meio de um LED emissor infravermelho responsável por emitir a luz infravermelha e um LED receptor infravermelho para detectar o sinal infravermelho.

Sensor MPU-6050

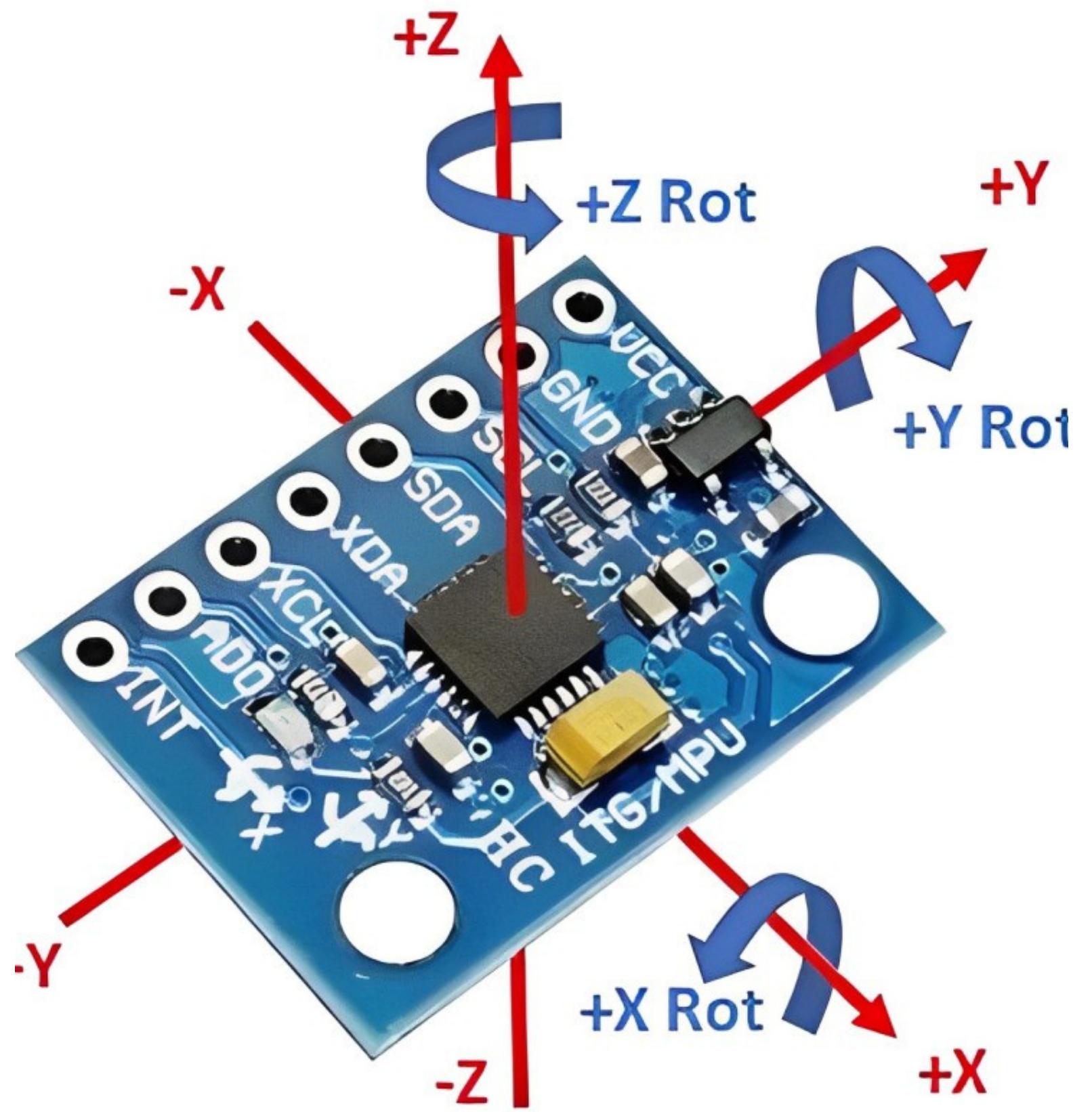
Combina um giroscópio e um acelerômetro.

01

Giroscópio é um dispositivo que mede a variação do ângulo, realizando a conversão de quão rápido o sensor está girando em graus por segundo.

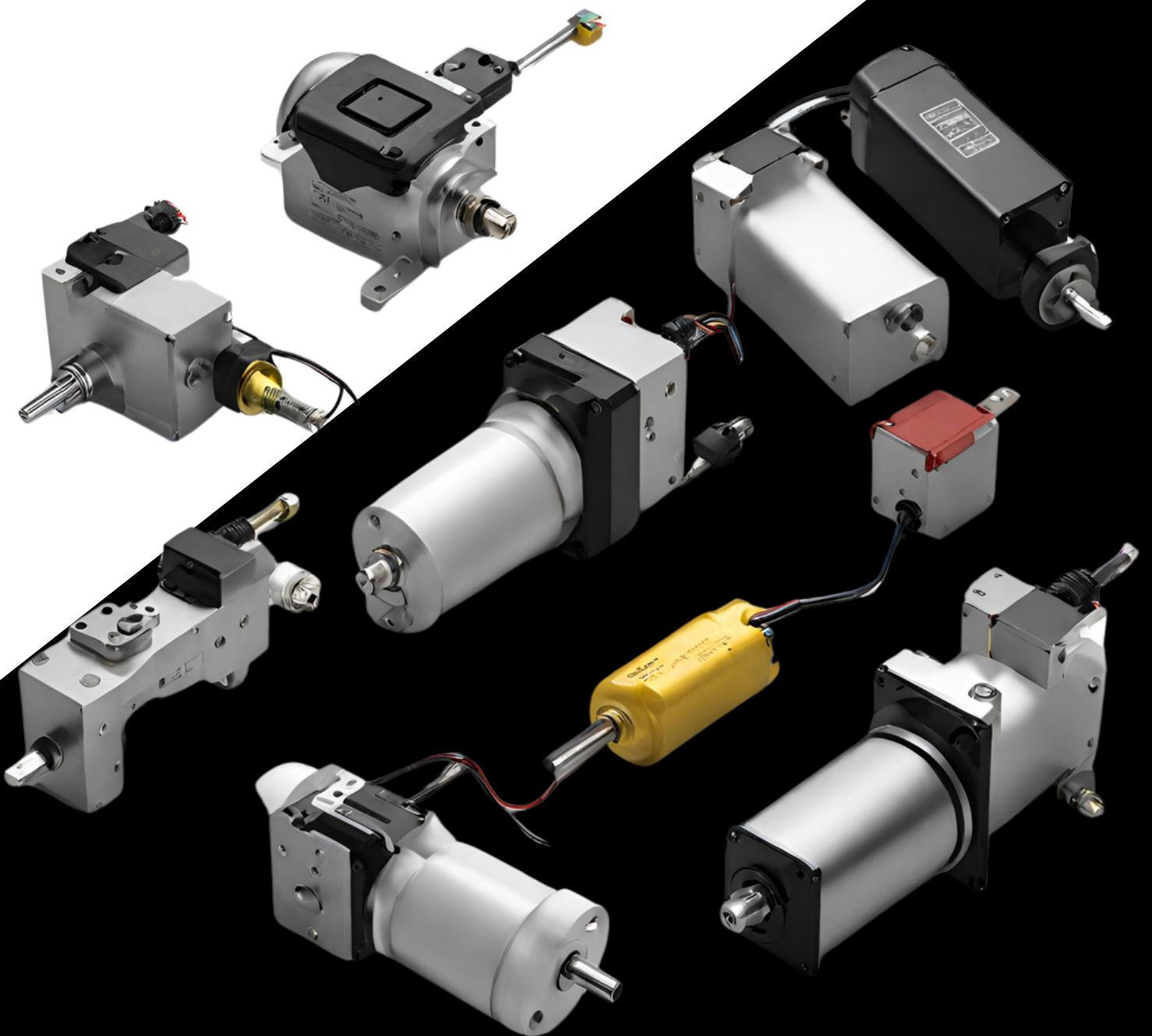
02

O acelerômetro exerce a função de medir acelerações, tendo como referência forças externas que agem sobre o mesmo



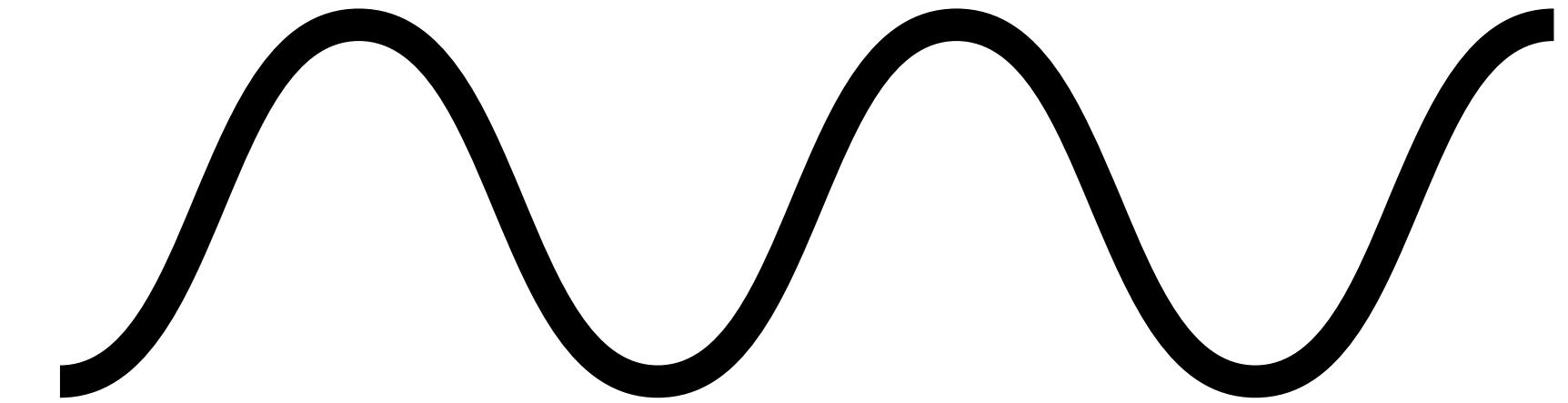
Atuadores

Dispositivos que exercem a função de converter energia elétrica, hidráulica ou pneumática em potência mecânica.



Motores elétricos

Corrente Alternada



São os motores mais utilizados, por terem um custo mais baixo, pois a distribuição de energia elétrica é feita principalmente em corrente alternada, e também por funcionarem com uma velocidade fixa proporcionando grandes potências.

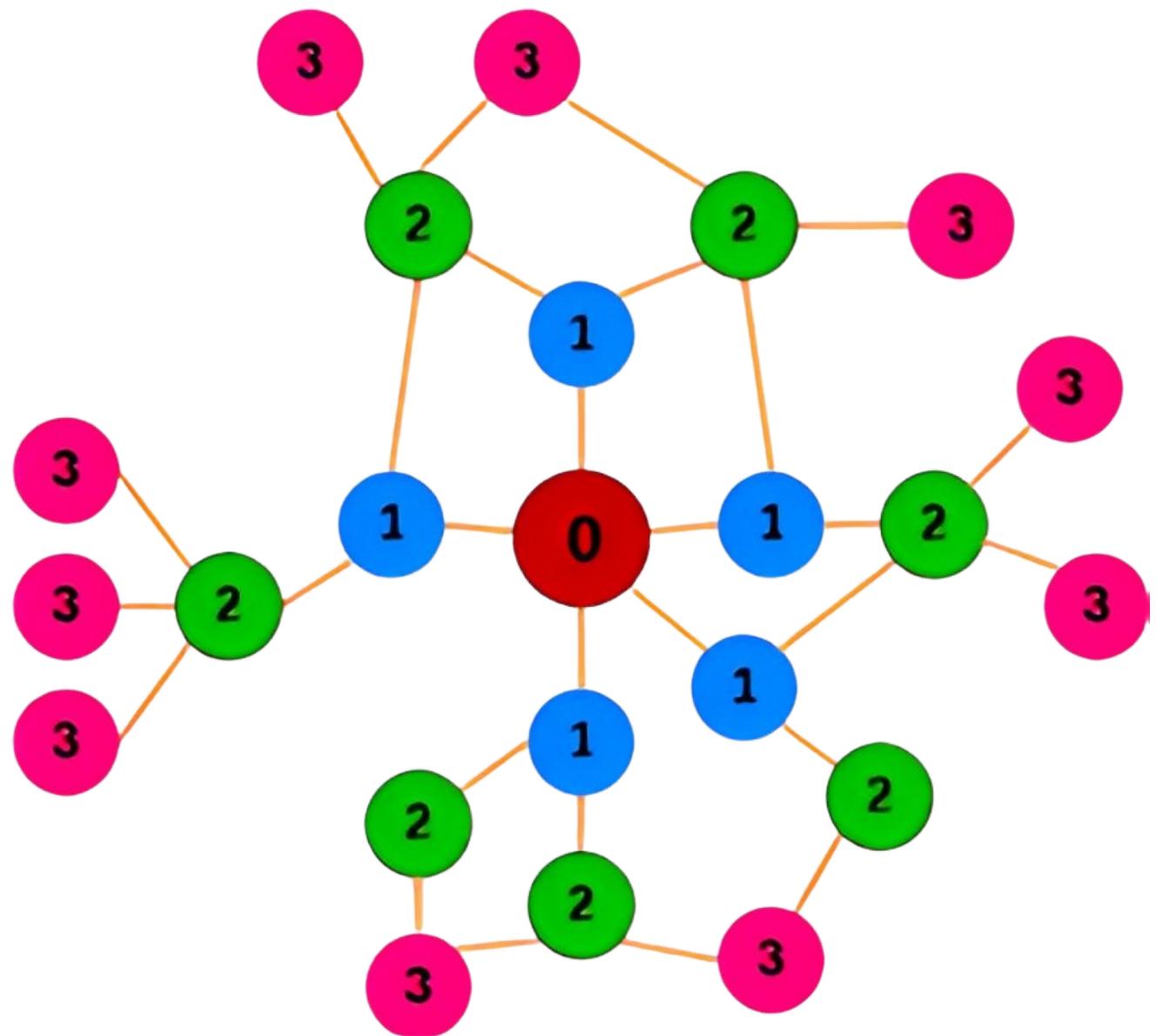
Motores elétricos

Corrente Contínua

Custo mais elevado, pois necessitam de uma fonte de corrente contínua e de uma manutenção frequente. Entretanto ao contrário dos motores de corrente alternada este pode funcionar com velocidades ajustáveis.

ALGORITMO DE BUSCA EM LARGURA

Utilizado para determinar o menor caminho de um ponto a outro, manualmente, fazendo a busca em todos os caminhos possíveis.



Metodologia

Características



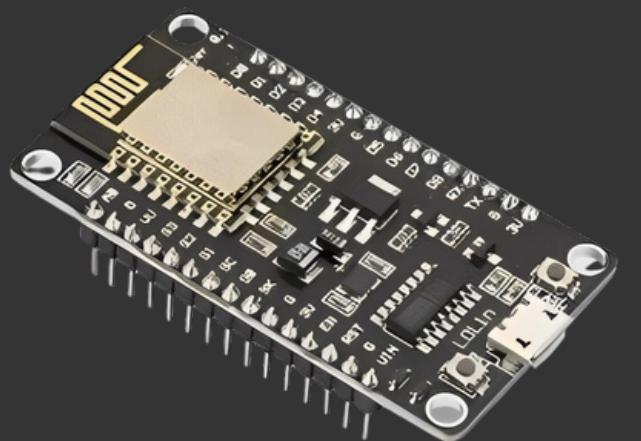
Abordagem Quali-quantitativa

Natureza aplicada

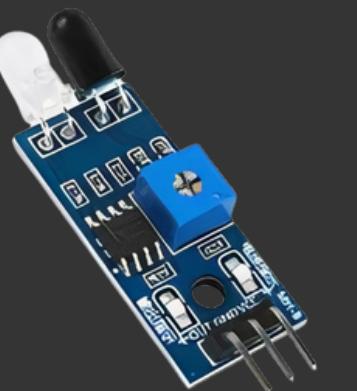
Objetivos exploratória

Pesquisa bibliográfica

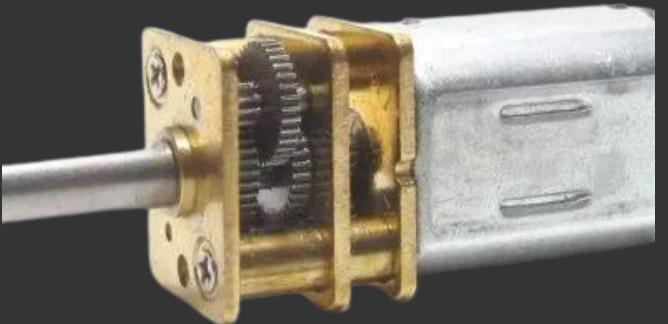
Componentes



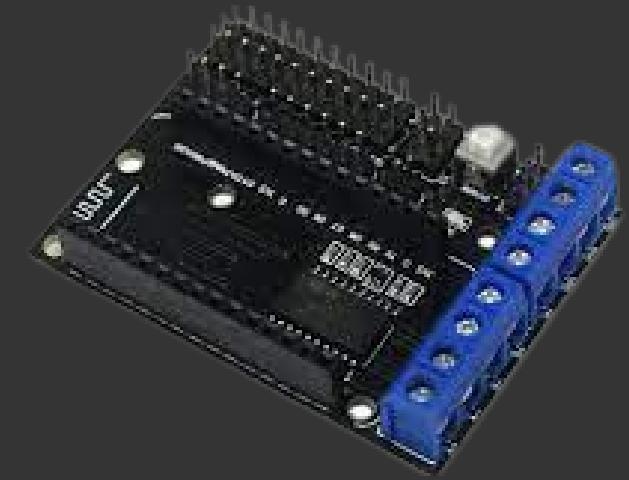
Microcontrolador



Sensor

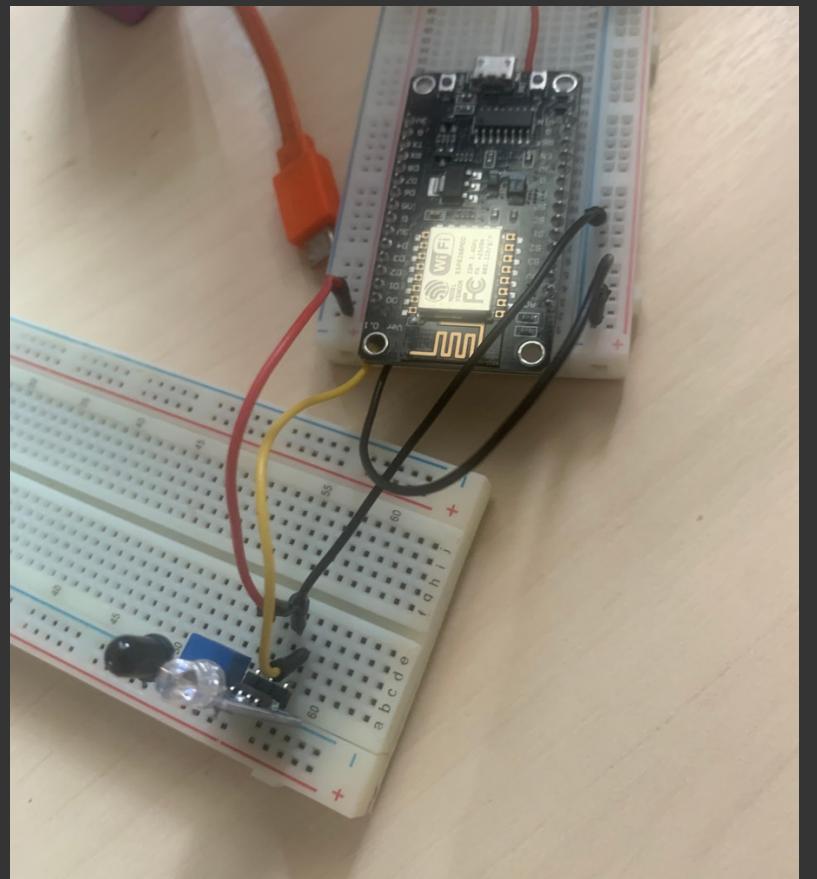
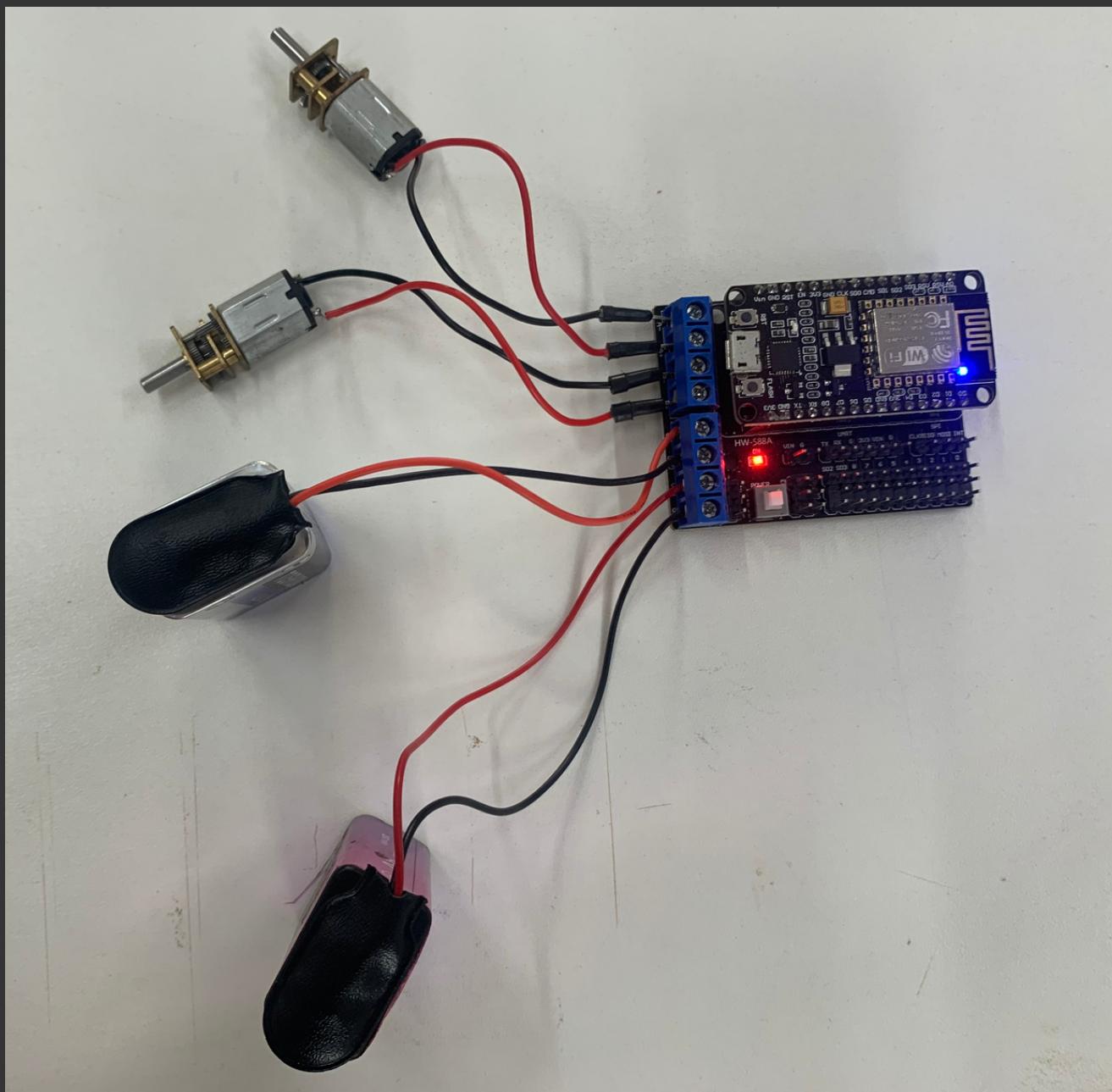


Motor



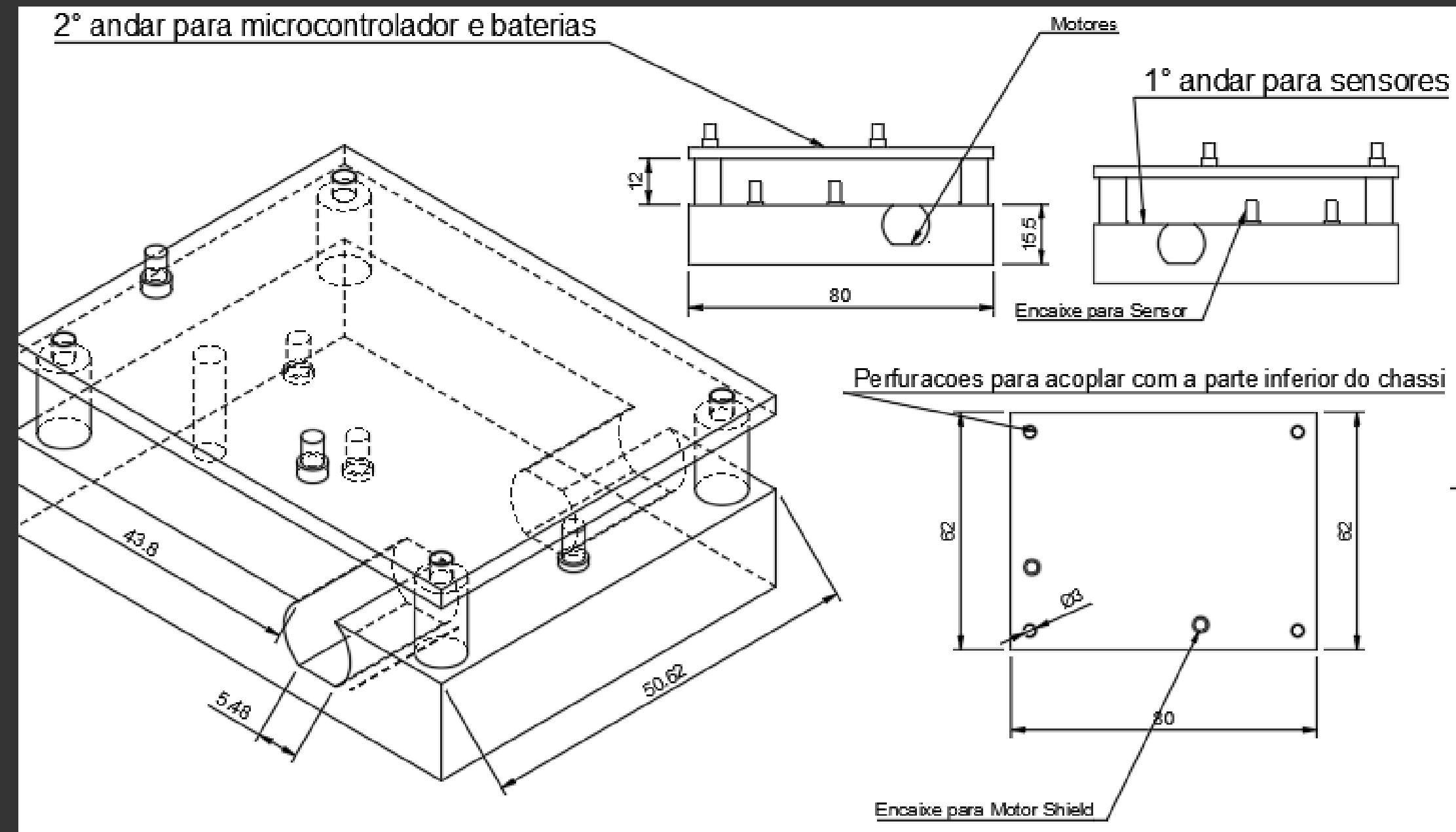
Motor shield

Teste com motores



Teste com sensor

Estrutura





Labirinto

O labirinto foi planejado para possuir 6 linhas e 6 colunas, dispondo de paredes em suas laterais.

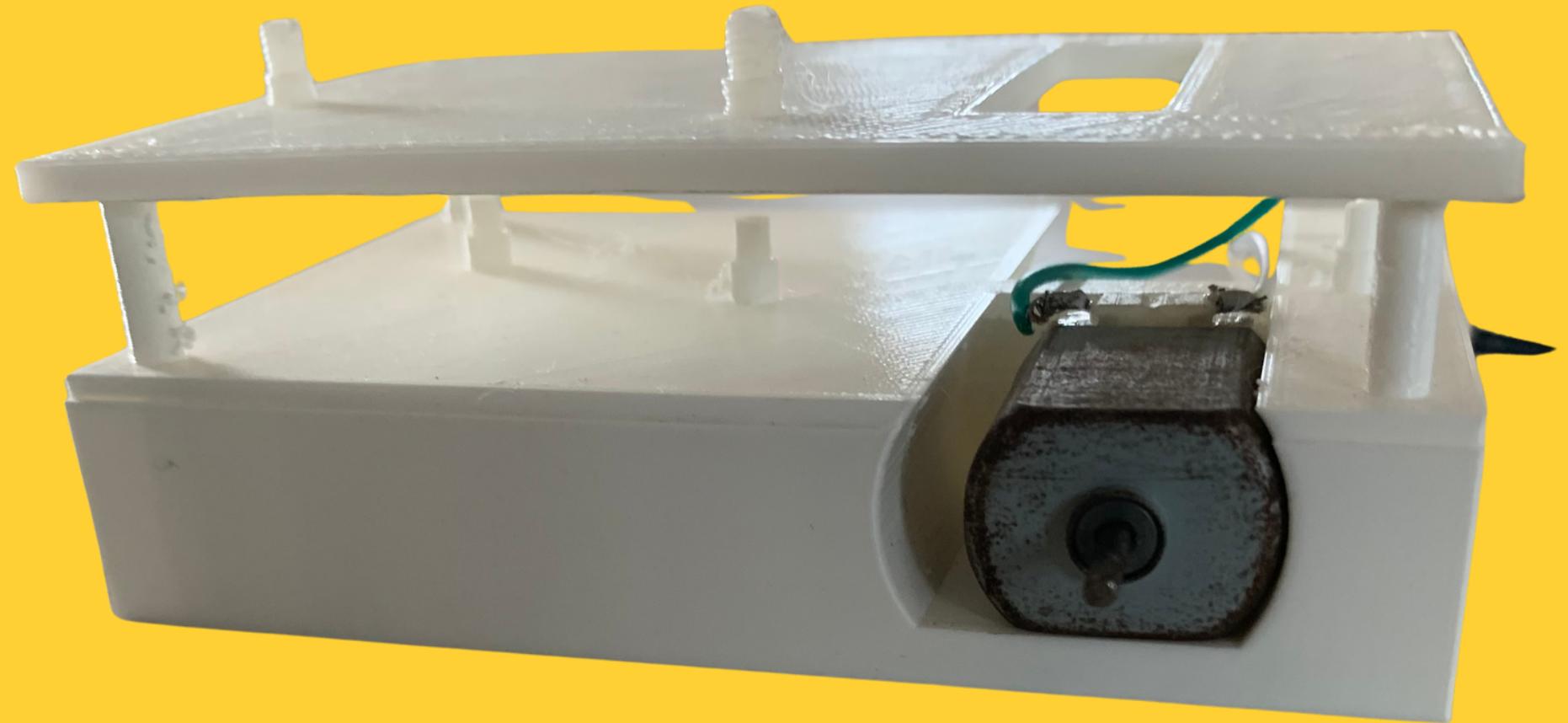
Possui medida total de 90x90 cm, havendo em seu piso perfurações a cada 15 cm para demarcar local de acoplamento das paredes.



Resultados

—

Construção do protótipo.

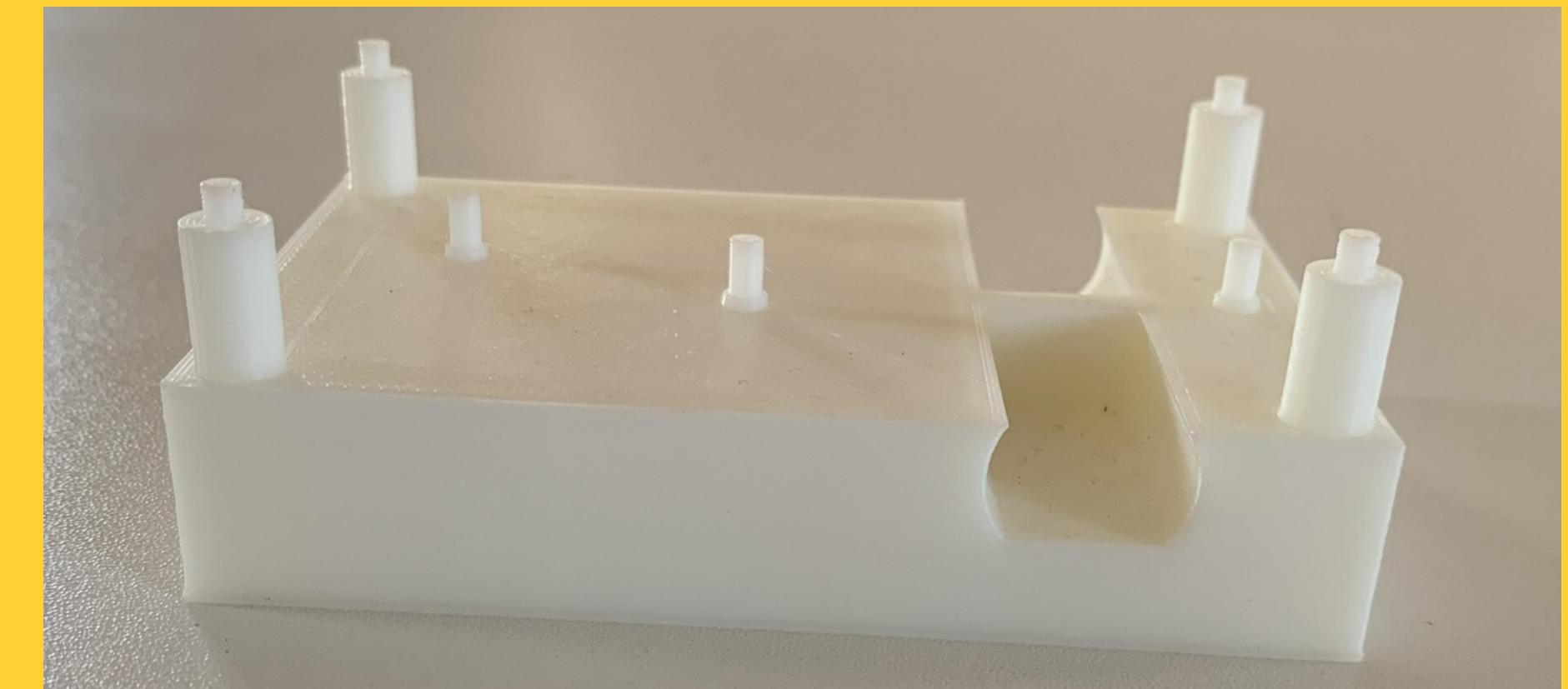
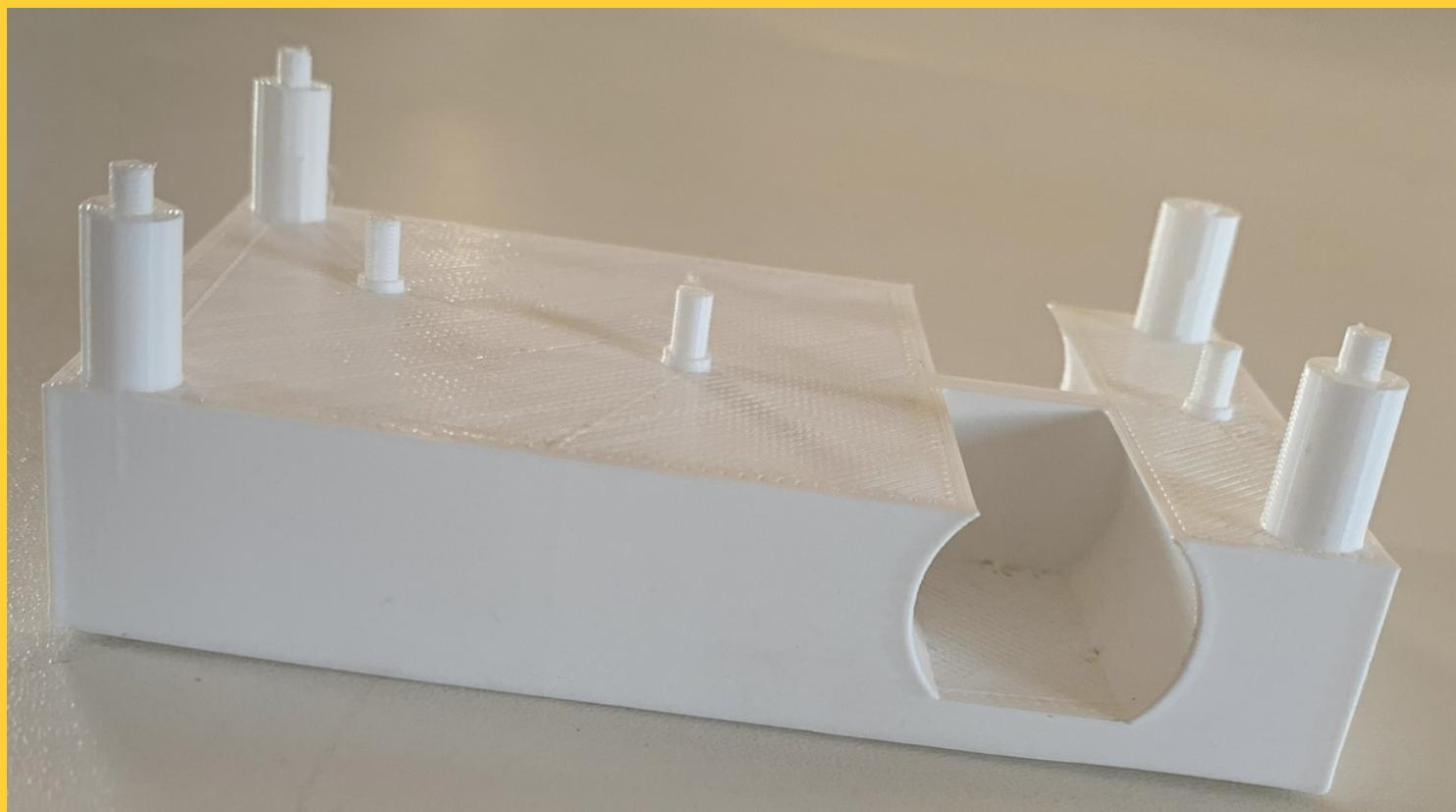


Primeiro protótipo

Após a impressão do modelo notou-se que certas medidas como a de encaixe dos motores estavam inadequadas para uso.

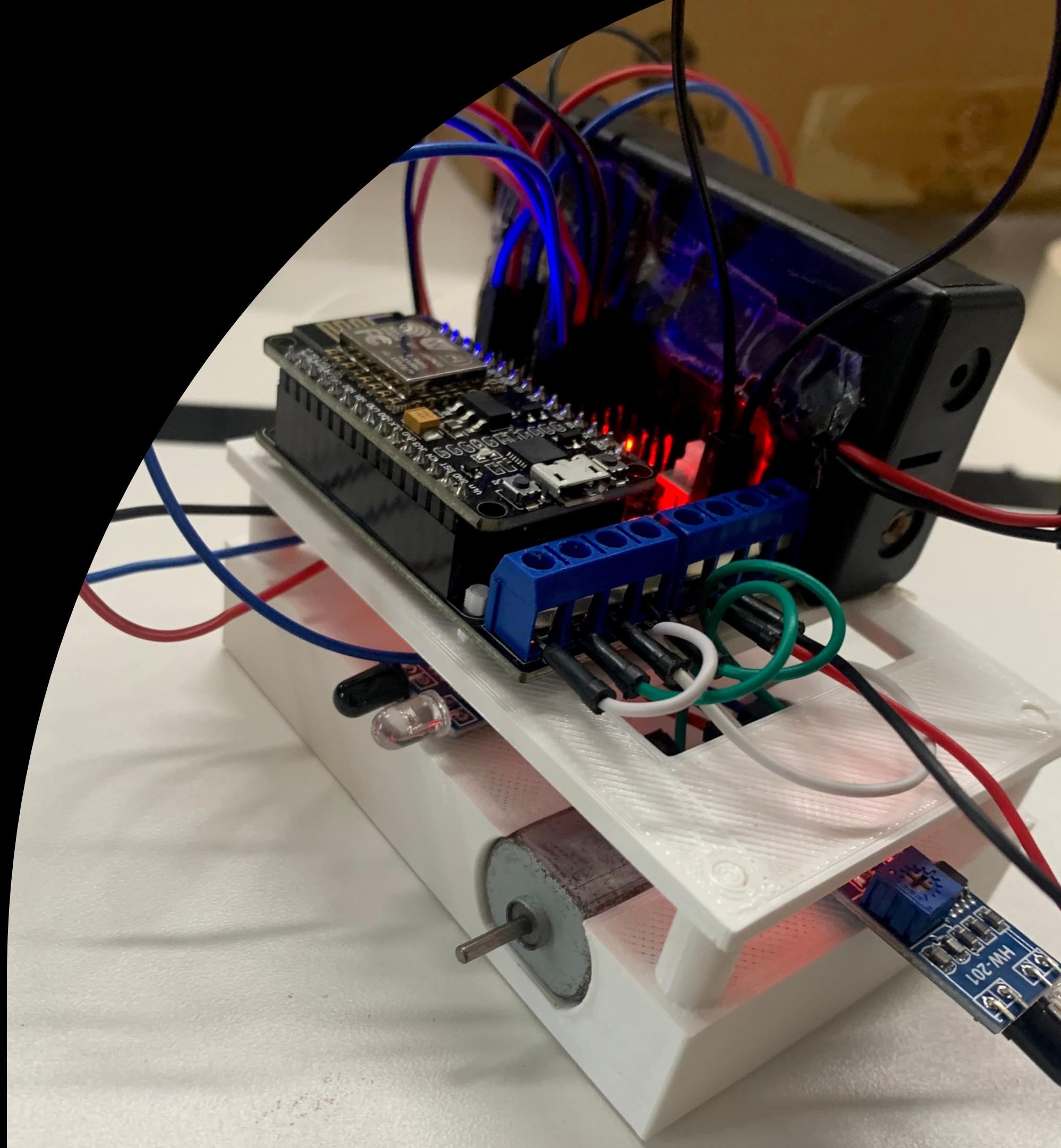
ENTRETANTO...

Ao decorrer do trabalho ocorreram cerca de 9 erros referentes a alguma medida, sendo assim necessário a impressão de 9 vezes da estrutura para serem alcançadas as medidas exatas para o segundo protótipo.



Segundo protótipo

Testes foram realizados com este protótipo a fim de confirmar as capacidades desse modelo, sem demora, notou-se que os motores utilizados não seriam eficazes para tal trabalho por possuírem pouco torque e baixa compatibilidade com rodas disponíveis, assim sendo necessário trocá-los por outra opção que fornecesse maior desempenho e maior uniformidade.



Componentes adquiridos

Motor 300 RPM

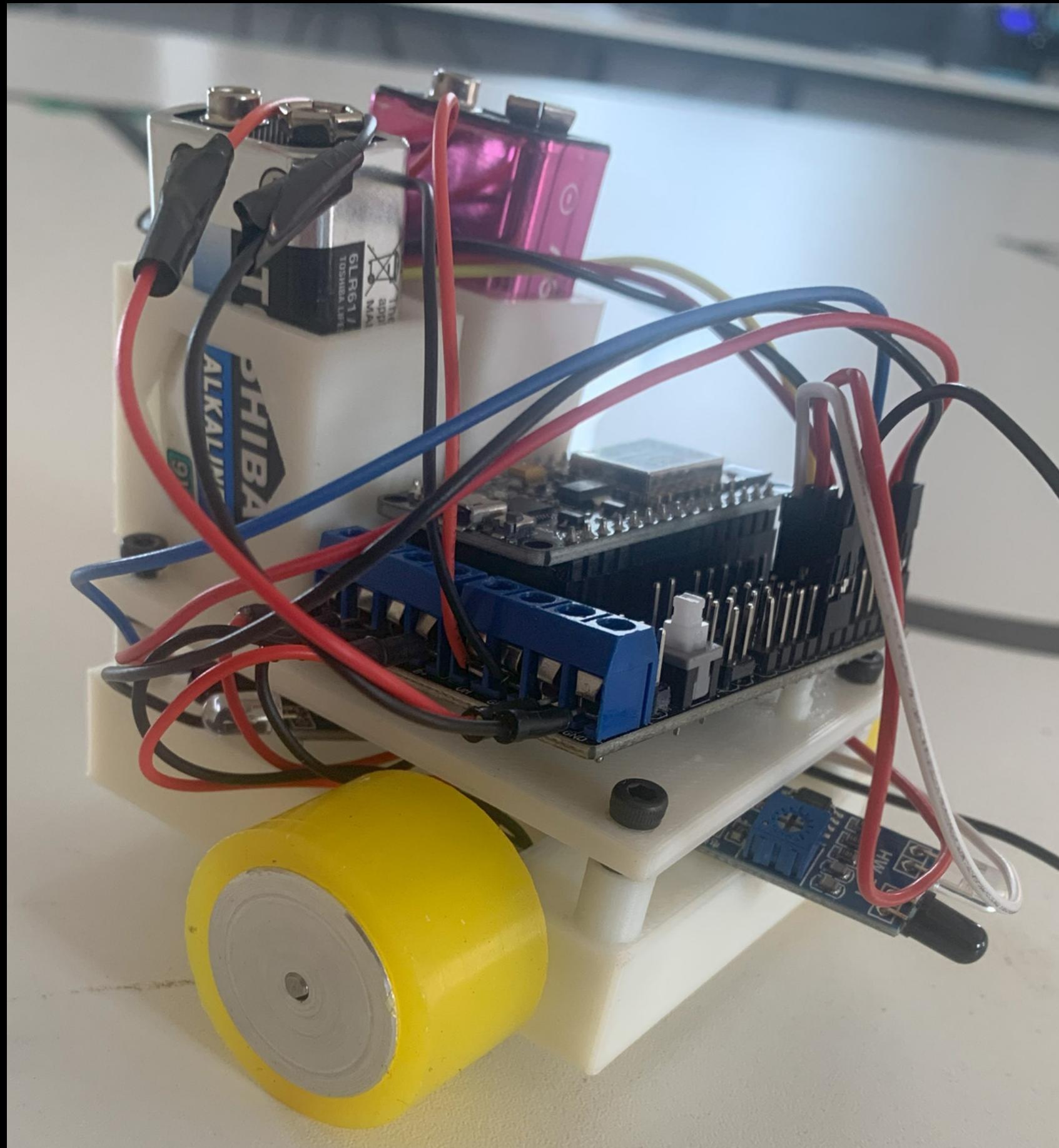


**Roda Macia -
32mm**



Bateria 9 Volts

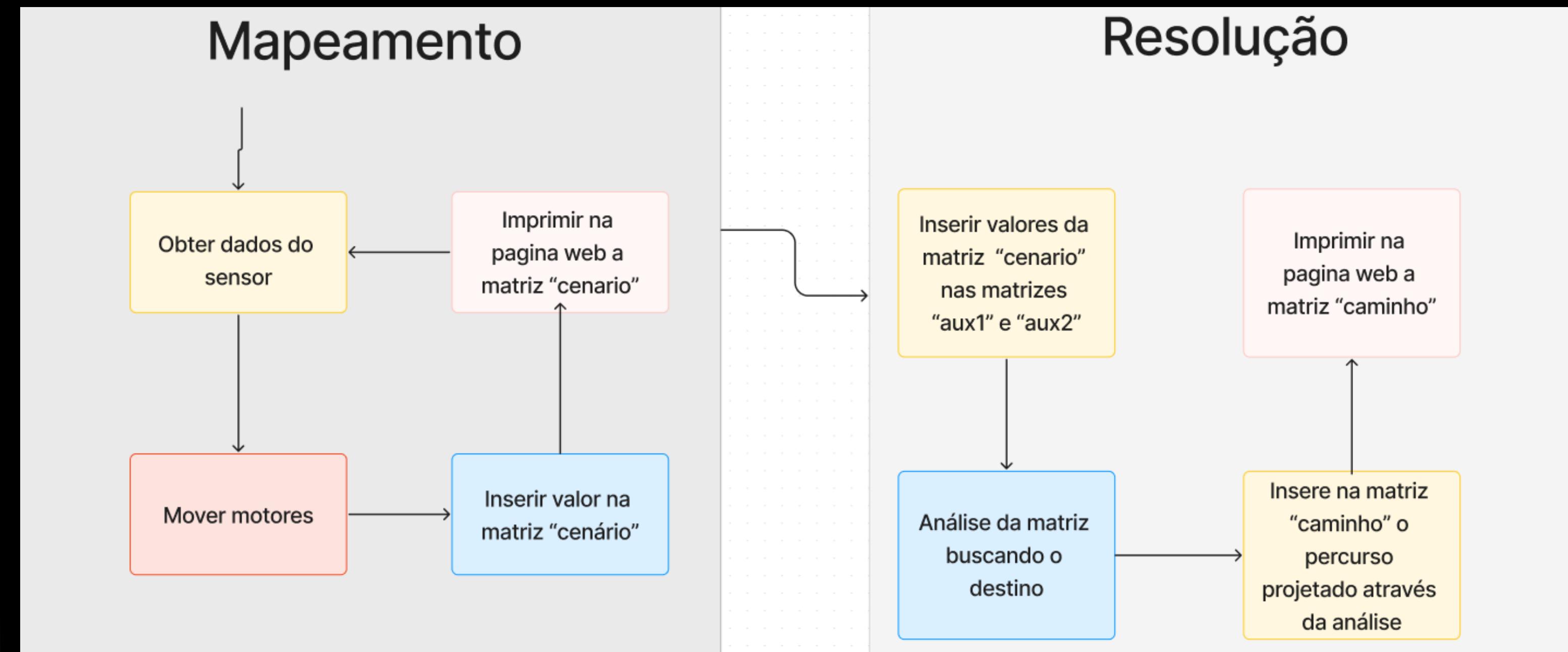




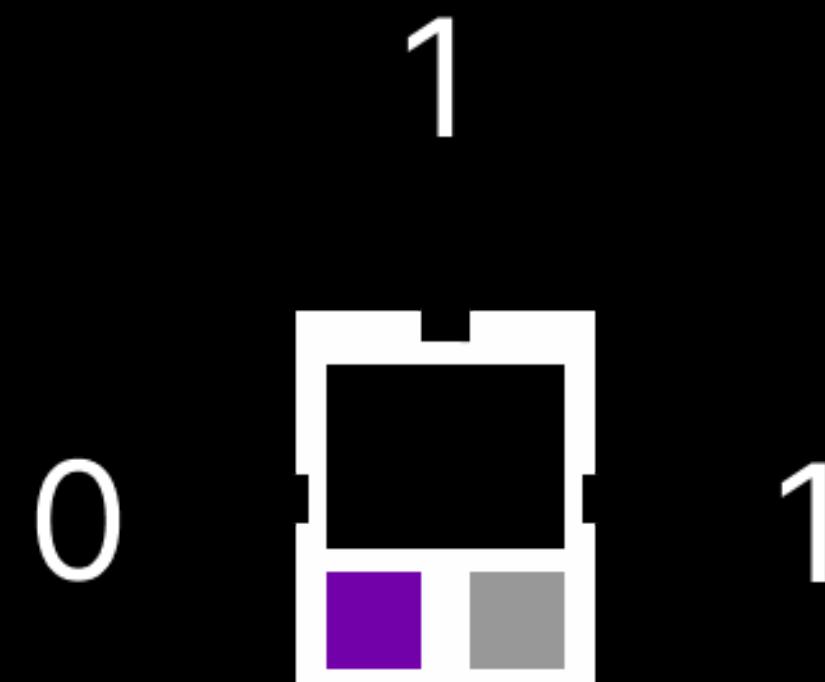
Terceiro protótipo

Com o terceiro protótipo montado, iniciou a fase da programação, para assim se realizar testes de bancada e se necessário buscar melhorias do protótipo.

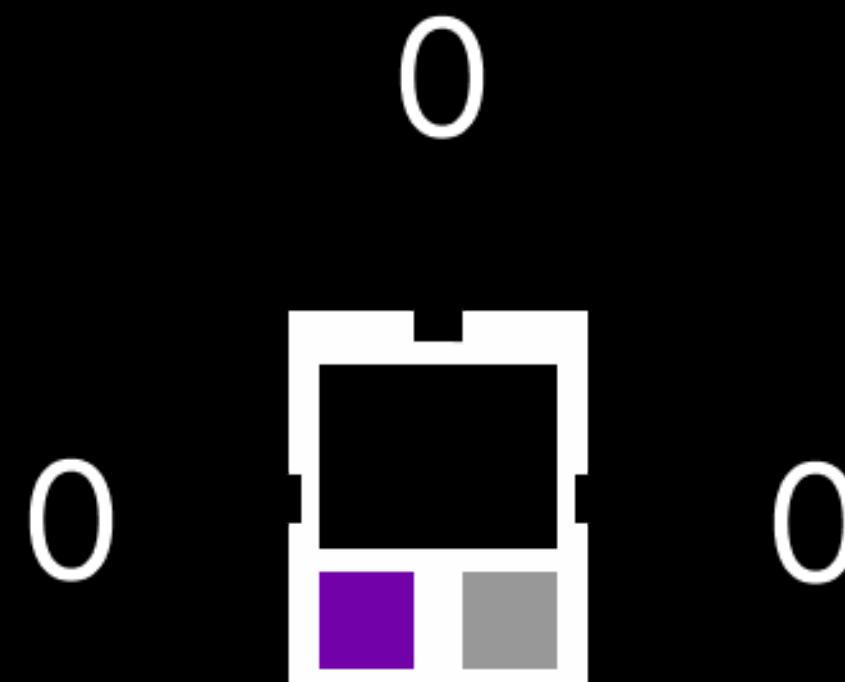
Fluxograma



Exemplos dados dos sensores

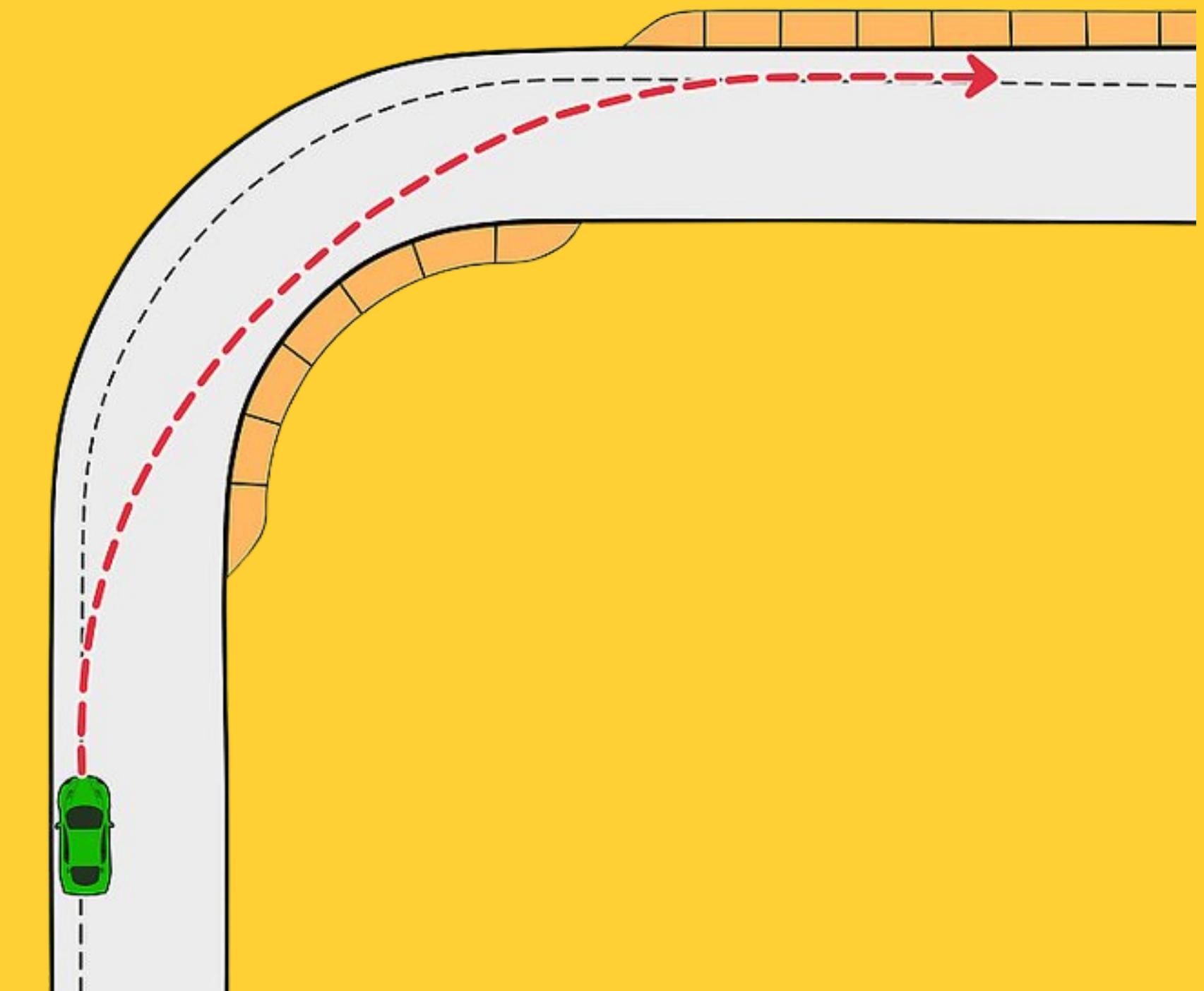


Parede à esquerda

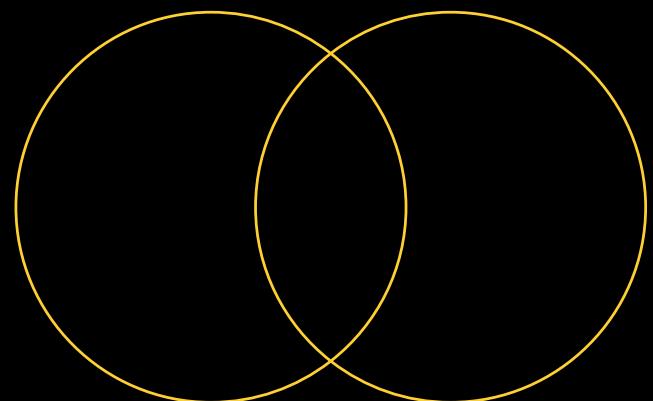


Parede em todos os lados

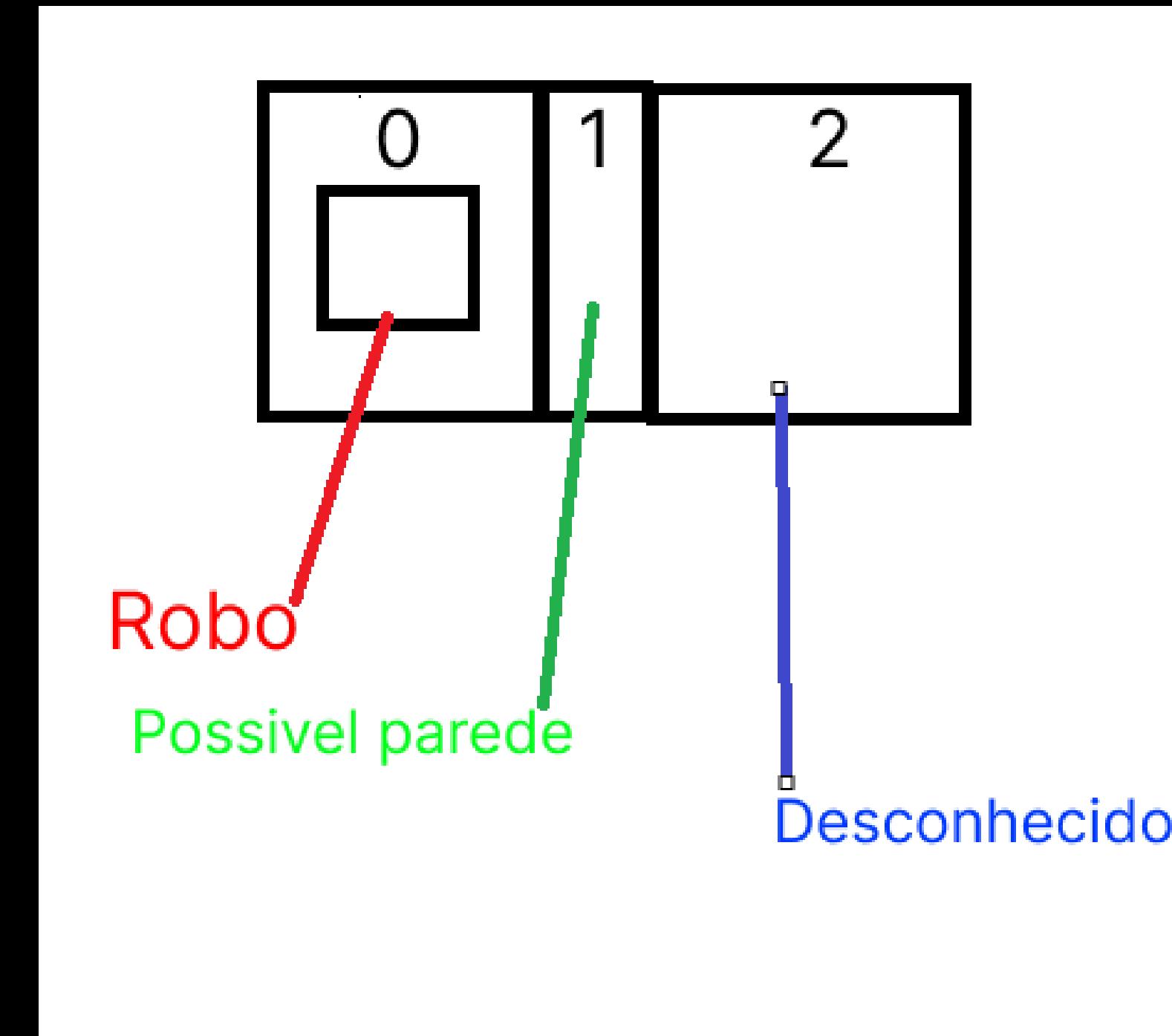
Movimento



Inserir valor



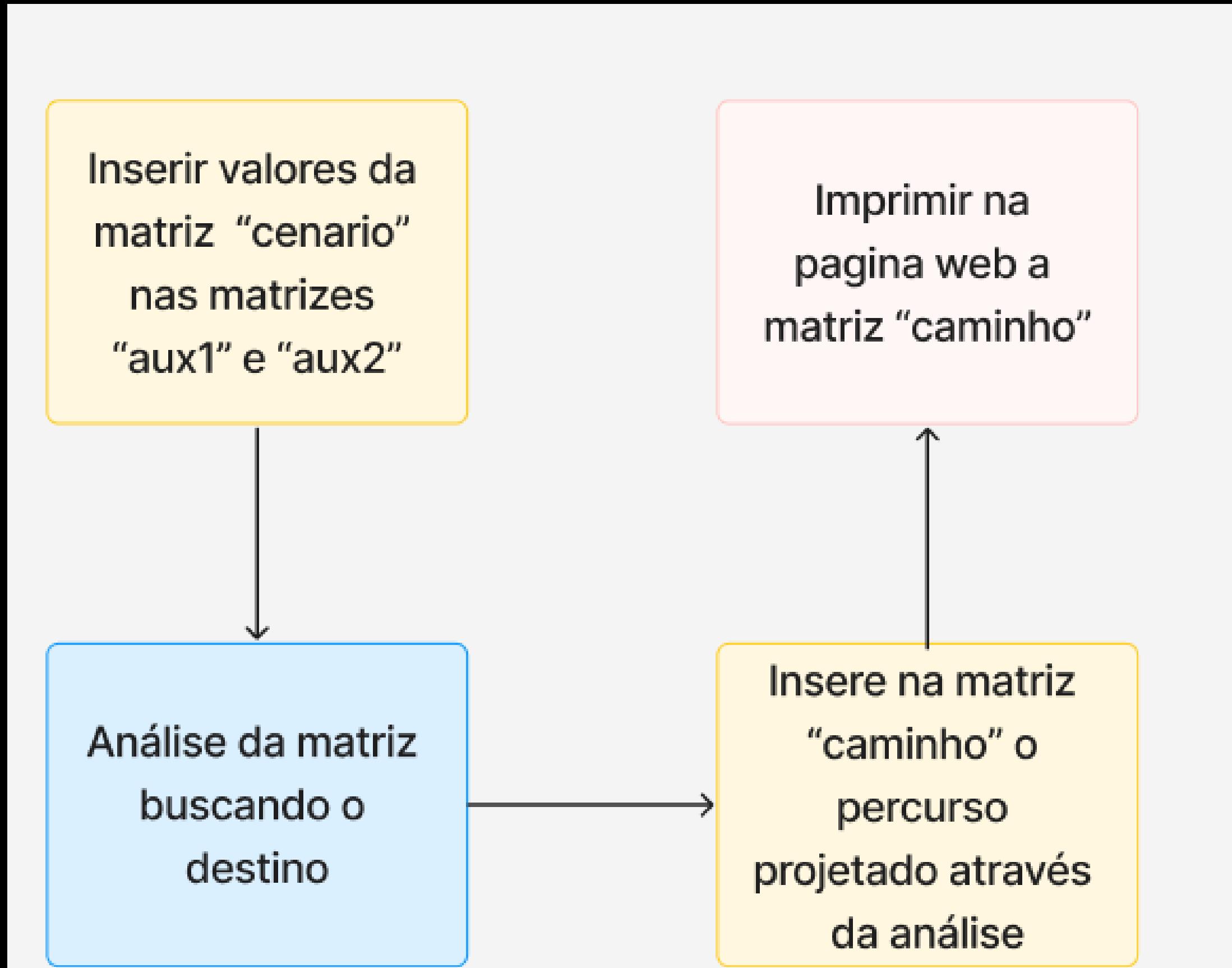
todos os números ímpares matriz
são possíveis paredes, esta
característica ocorre em razão de
que é feita o mapeamento através
das paredes próximas.



PÁGINA WEB

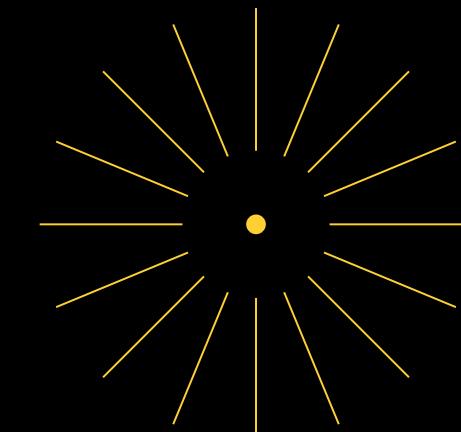
Visualização do mapeamento em tempo real.

Resolução



Inserção de valores nas matrizes auxiliares

```
for(x=0;x<tamanho;x++)
{
    for(y=0;y<tamanho;y++)
    {
        if(cenario[x][y] == -99)
        {
            origem[0] = x;
            origem[1] = y;
            matriz[x][y]=0;
            aux[x][y]=0;
        }
        if(cenario[x][y] == 999)
        {
            destino[0] = x;
            destino[1] = y;
            matriz[x][y]=999;
            aux[x][y]=999;
        }
        if(cenario[x][y] == 1)
        {
            matriz[x][y]=-60;
            aux[x][y]=-60;
        }
        if(cenario[x][y] == 0)
        {
            matriz[x][y]=-30;
            aux[x][y]=-30;
        }
    }
}
```



Buscar o destino

```
if (matriz[a][b+1]==-60)
| { matriz[a][b+1]=indice+1; aux[a][b+1]=indice+1;    }
if (matriz[a+1][b]==-60)
| { matriz[a+1][b]=indice+1; aux[a+1][b]=indice+1;    }
if (matriz[a-1][b]==-60)
| { matriz[a-1][b]=indice+1; aux[a-1][b]=indice+1;    }
if (matriz[a][b-1]==-60)
| { matriz[a][b-1]=indice+1; aux[a][b-1]=indice+1;    }

if(matriz[a+1][b]==999 || matriz[a][b+1]==999 || matriz[a-1][b]==999 || matriz[a][b-1]==999 )
| { Serial.println("Destino encontrado"); achou=1; }
cont--;
```

Elaboração do Caminho

```
if (matriz[a-1][b]==busca)
| { caminho[a-1][b]=busca+1;      a=a-1;    b=b;   }
else if (matriz[a+1][b]==busca)
| { caminho[a+1][b]=busca+1;      a=a+1;    b=b;   }
else  if (matriz[a][b+1]==busca)
| { caminho[a][b+1]=busca+1;      a=a;      b=b+1;  }
else  if (matriz[a][b-1]==busca)
| { caminho[a][b-1]=busca+1;      a=a;      b=b-1;  }
```



Resultados Terceiro protótipo

PAGINA WEB

Visualização do caminho

Resultados Terceiro Protótipo

Problemas

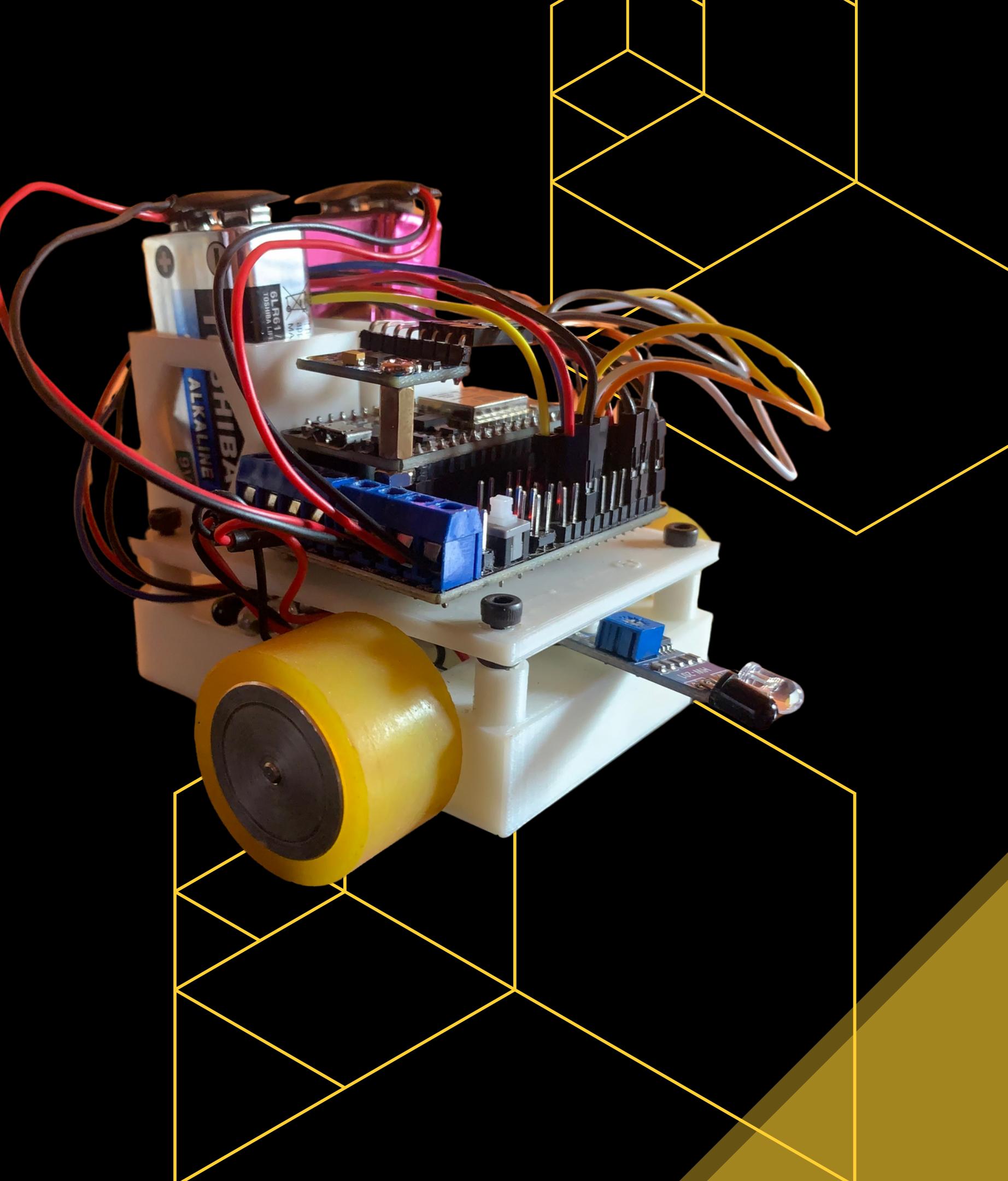
- 1 Locomoção

Soluções

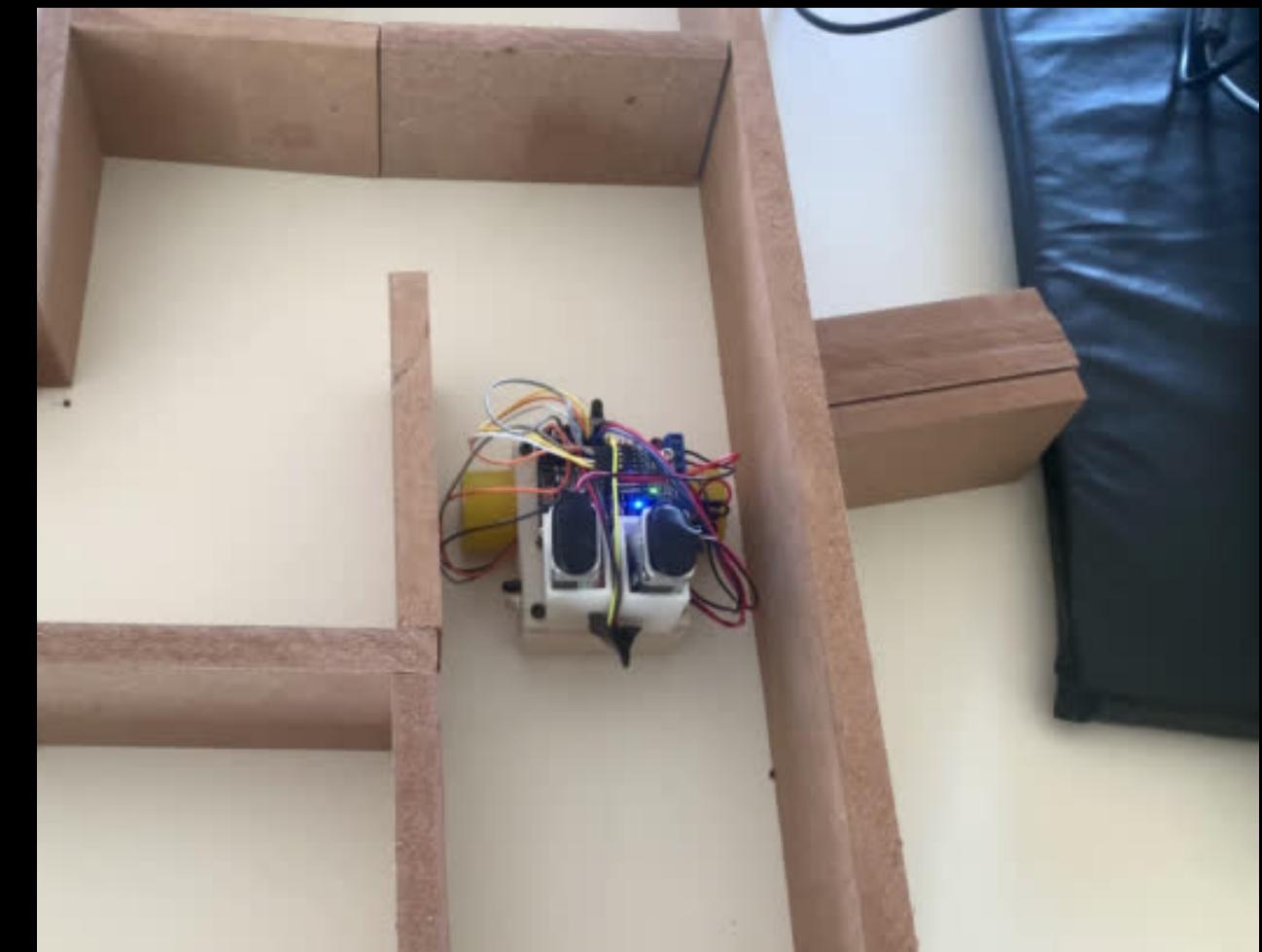
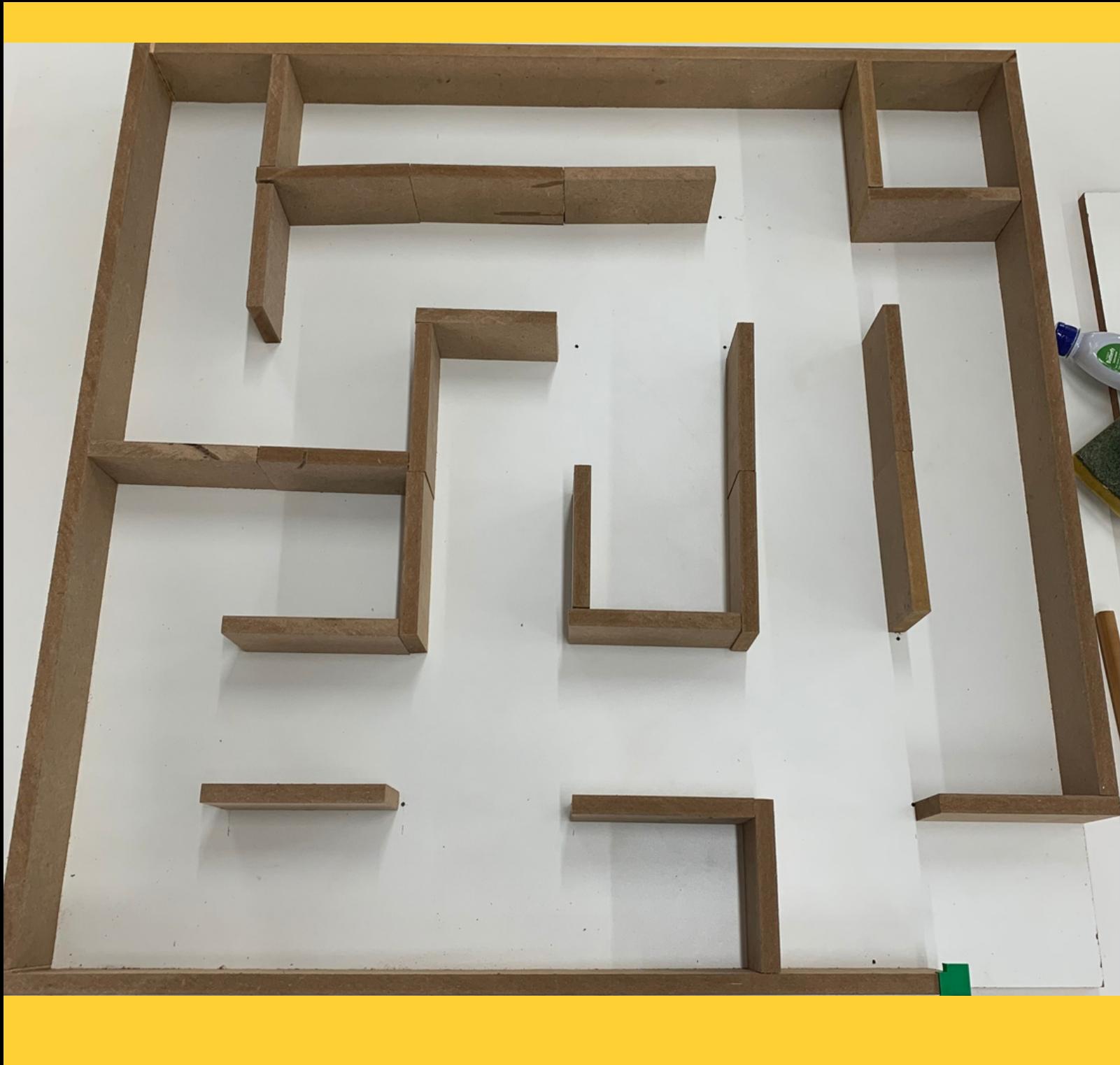
- 1 Motores de 30 RPM
- 2 Sensor MPU 6050

Quarto Protótipo

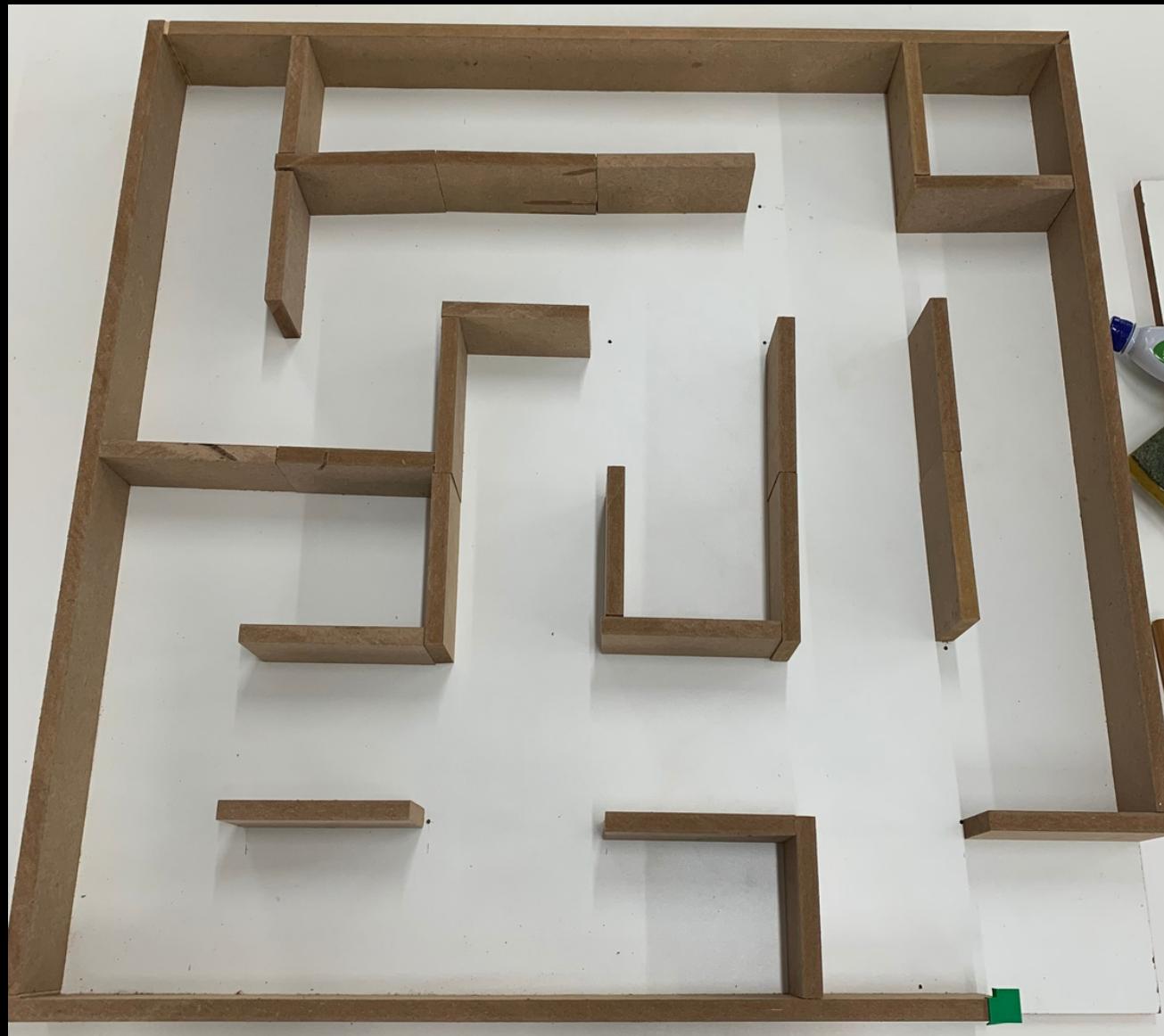
Após a implementação do sensor MPU-6050, foi realizado testes de bancadas a fim de validar o protótipo elaborado.



Validação do protótipo

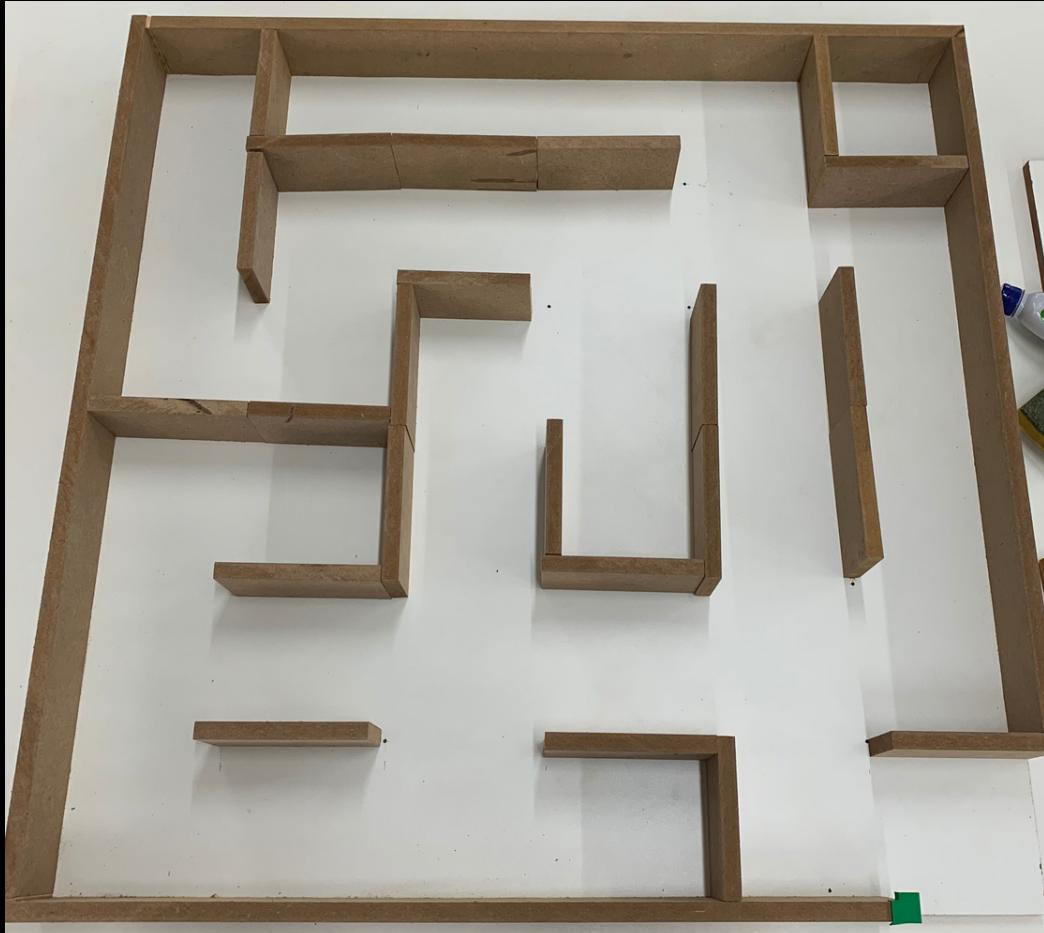


Resultados



```
|0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0| | | | | | | | |
|0||-99||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0|
|0||1||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0|
|0||1||0||1||1||1||1||1||1||0||0||0||0||0||0||0||0|
|0||1||0||1||1||0||0||0||0||0||0||1||0||0||0||0||0|
|0||1||1||1||1||0||0||0||0||0||1||0||0||0||0||0||0|
|0||1||1||1||1||0||0||0||0||0||1||0||0||0||0||0||0|
|0||0||0||0||0||0||0||0||0||0||1||0||0||0||0||0||0|
|0||0||0||0||0||0||0||0||0||0||1||1||0||0||0||0||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||0||0||0||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||0||0||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||1||0||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||1||1||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||1||1||1||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||1||1||1||1||0|
|0||0||0||0||0||0||0||0||0||0||1||1||1||1||1||1||1||1||1||999||0|
|0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0||0|
```

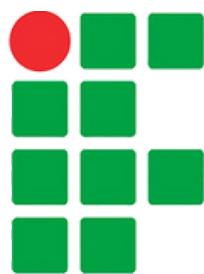
Resultados



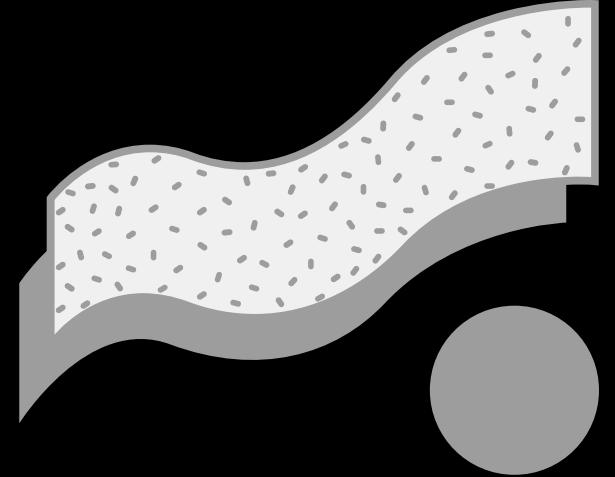
Conclusão

Os objetivos deste trabalho foram atingidos com êxito, portanto, expondo dessa forma que é possível realizar o desenvolvimento de um protótipo autônomo capaz de mapear e solucionar labirintos utilizando tecnologia do ESP8266 e algoritmos de busca em largura.

Devido ao foto do protótipo final ser compacto e didático, pode ser usado para fins de aprendizado sobre algoritmos de busca aplicados em situações do cotidiano, tendo em vista que o protótipo realiza o mapeamento dos possíveis trajeto necessário para alcançar o destino e logo após procurar o melhor caminho.



Trabalhos Futuros



Desenvolver um sistema para o mapeamento utilizando visão computacional;

Implementar algoritmo Proporcional integral derivativo com base nos valores de ângulos obtidos para uma melhor locomoção entre as paredes;

Realizar o desenvolvimento de um aplicativo que exibe o mapeamento do labirinto em tempo real e após apresente o melhor caminho;

