



Projeto 09

Controle Serial – Prática

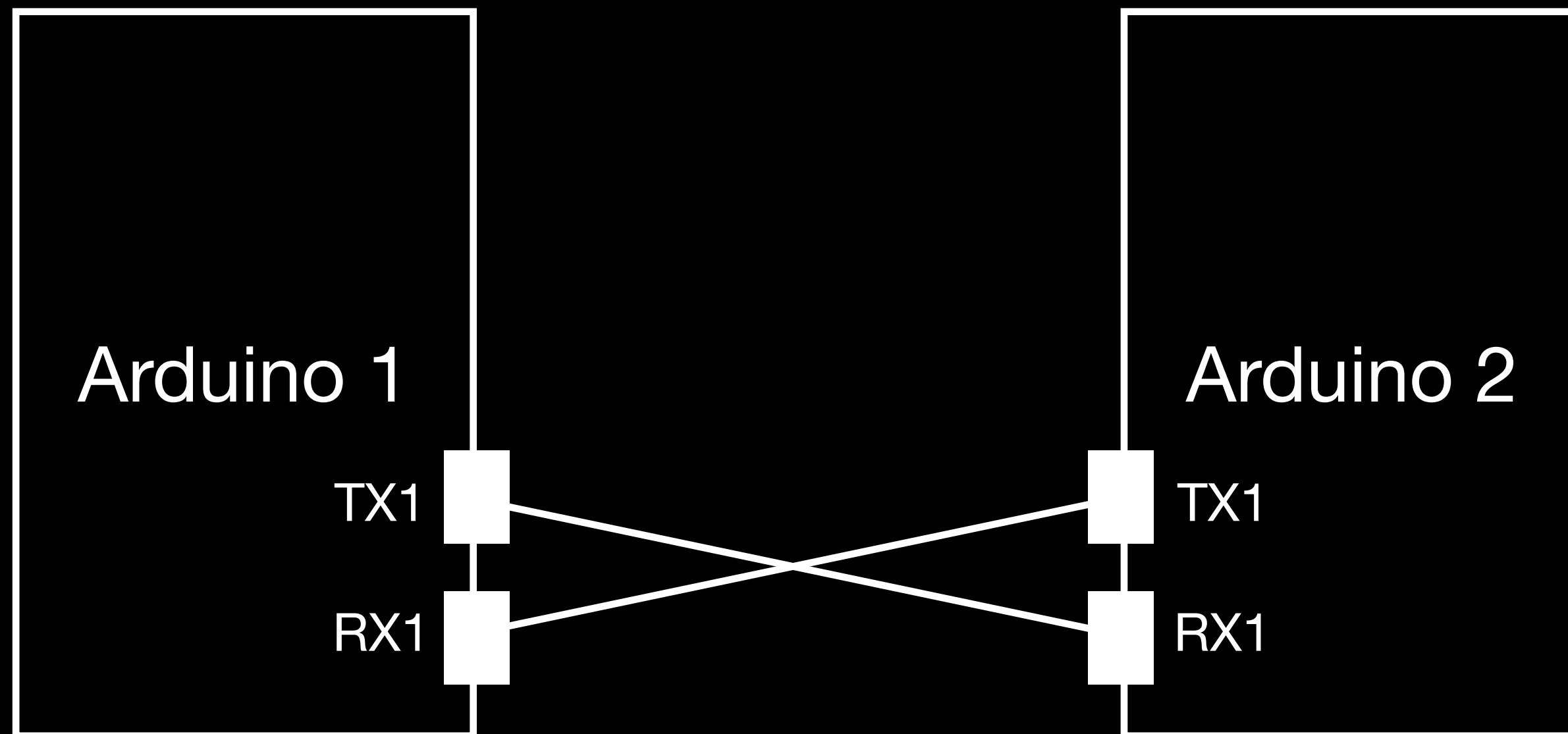
Jan K. S. – janks@puc-rio.br

ENG1419 – Programação de Microcontroladores

Testes Iniciais

Testes Iniciais 1

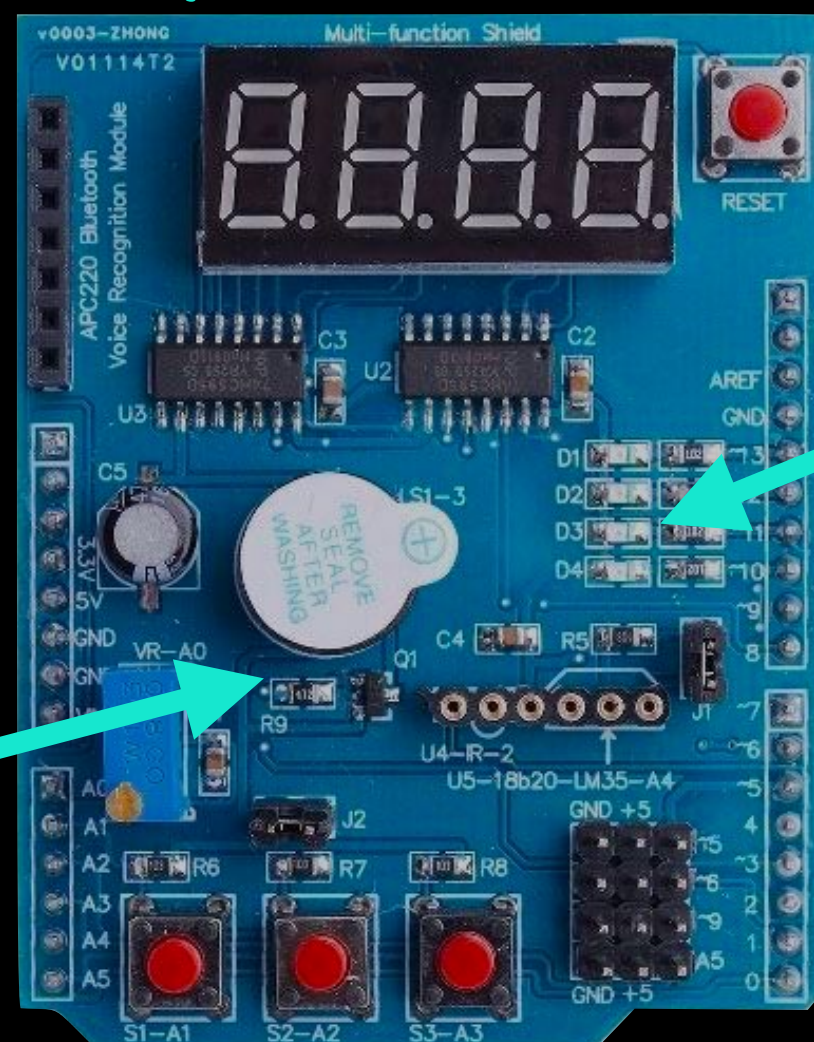
Testes Iniciais 2



Comunicação entre Arduinos

Testes Iniciais 1

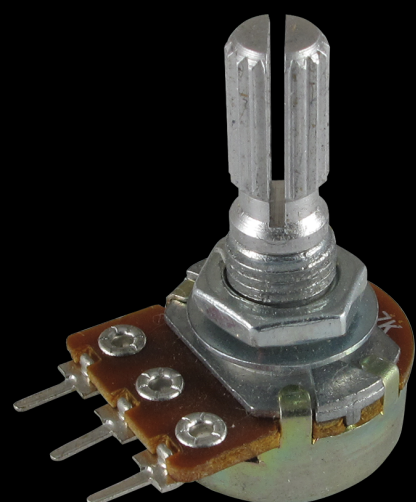
pinos 4, 7 e 8



pino 3

pinos 13,
12, 11 e 10

pinos A1, A2 e A3



pino A10

Testes Iniciais 2



motor 3
(esquerda)



motor 4
(direita)



pino A11
(esquerda)



pino A12
(direita)

Pinos Usados pelos Componentes



Testes Iniciais 01

Ao receber o texto "tocar", **toque a campainha ativa durante 200 ms.**

↳ DICA: use a função delay.

Mapeie o valor do potenciômetro para valores inteiros de 0 a 255. Exiba esse valor do display de 7 segmentos.

Ao apertar o Botão 1, **envie o texto "frente N"**, onde N é o de 0 a 255 mapeado pelo potenciômetro. Ao apertar o Botão 2, **envie o texto "tras N"**.

Ao receber o texto "contagem X", **exiba o valor X no display de 7 segmentos durante 1 segundo.**

↳ DICA: use as funções startsWith e substring.

Quando sua dupla e a dupla vizinha terminarem, **mude a comunicação para Serial1**, e teste todos os itens entre os Arduinos.



Testes Iniciais 02

Cada vez que o **valor digital do sensor ótico 1** mudar, **envie o comando "tocar"** para a Serial.

↳ DICA: crie uma variável global para armazenar o estado anterior do sensor.

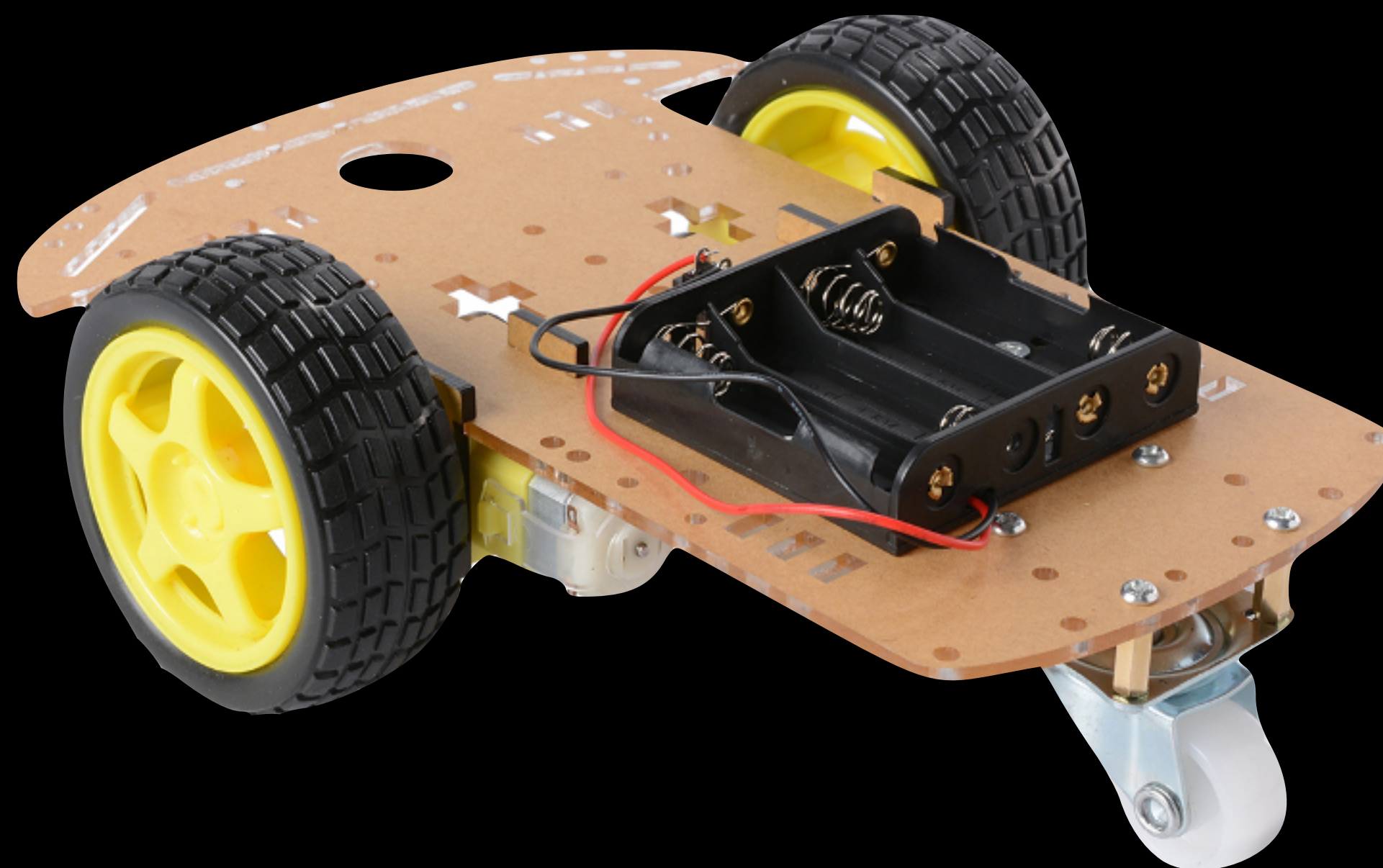
Ao receber o comando "frente N" da Serial, **gire o Motor 3 para frente com a velocidade N**. Ao receber "tras N", **gire para trás com a velocidade N**.

↳ DICA: use a função `startsWith` e `substring`.

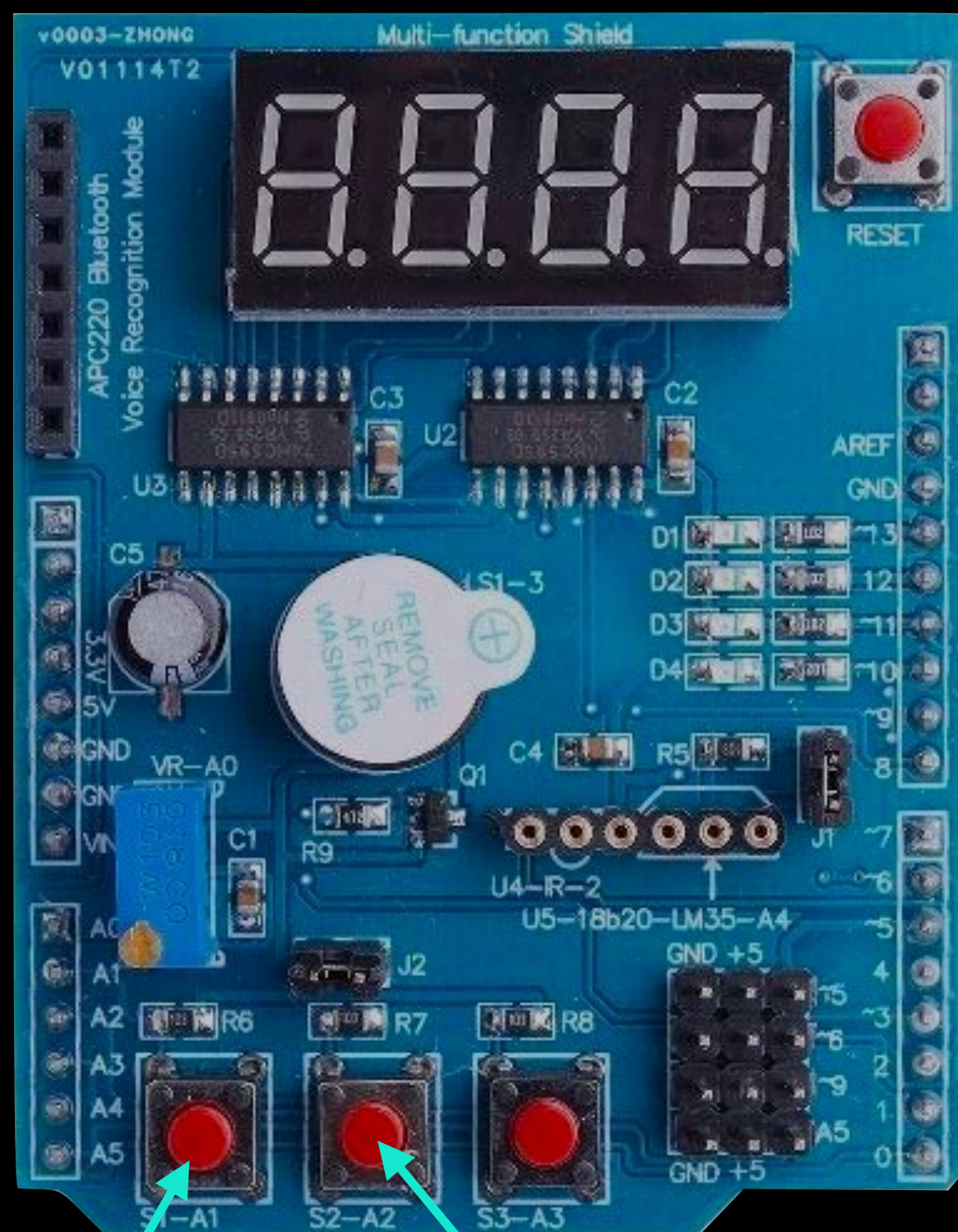
Cada vez que o valor analógico do sensor ótico 2 ultrapassar o limiar de 800, **aumente uma vez a contagem global X**. Em seguida, **envie o texto "contagem X"** (com o valor de X) pela Serial.

Quando sua dupla e a dupla vizinha terminarem, **mude a comunicação para Serial1**, e teste todos os itens entre os Arduinos.

Implementação



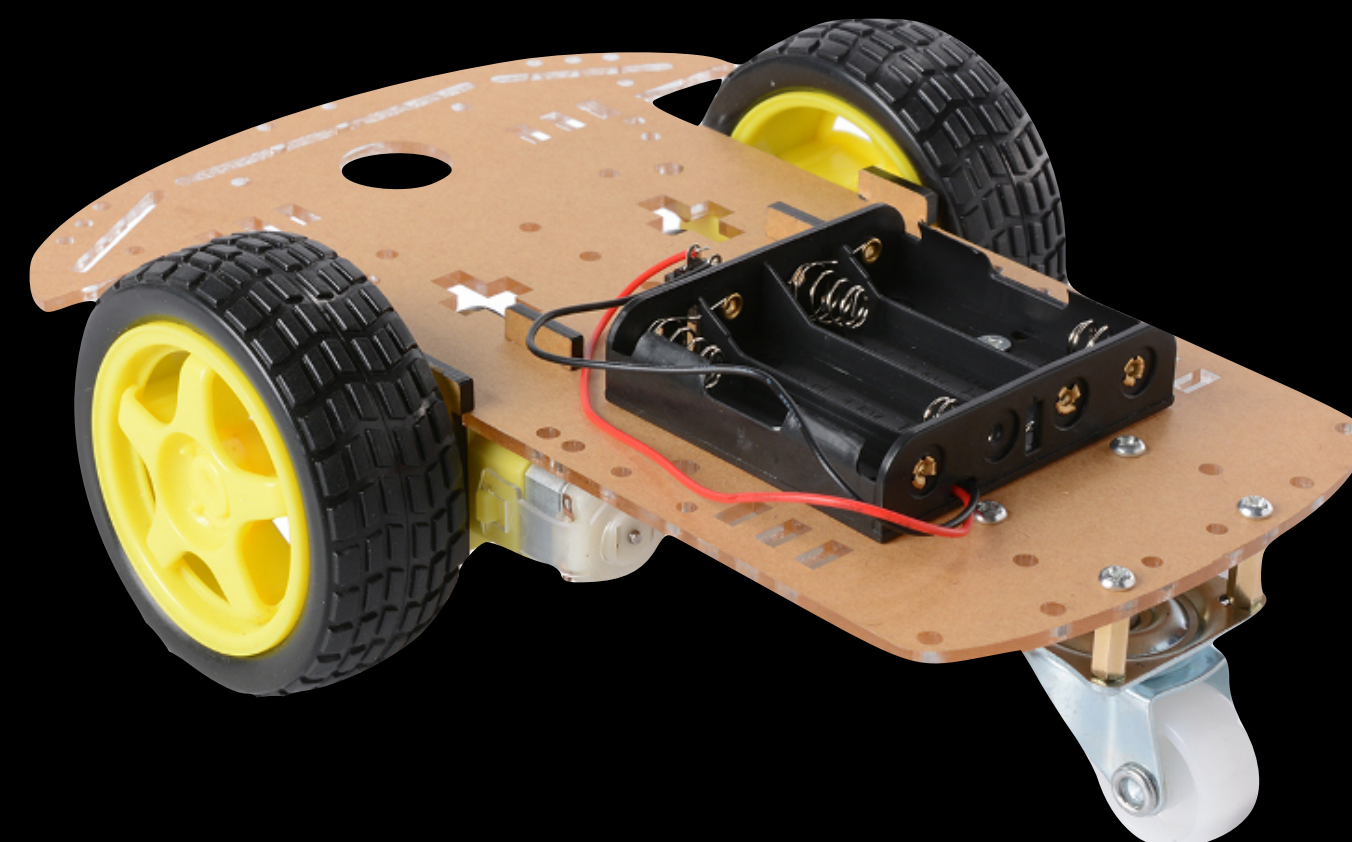
Carrinho de Duas Rodas



Andar / Parar

Direção

serial



Controle Manual do Carrinho



Implementação 01

Ao apertar o Botão 1, alterne o comando atual para "frente", "trás", "esquerda" e "direita", exibindo no display de 7 segmentos.

Ao apertar o Botão 2, envie o comando atual pela serial. Ao soltar o Botão 2, envie o comando "parar".

Ao receber os valores dos dois sensores, acenda/apague os LEDs 1 e 2 de acordo com o que foi lido (aceso = cor clara, apagado = cor escura).



Implementação 02

Controle os motores 3 e 4 do carro devidamente ao receber os comandos "frente", "trás", "esquerda", "direita" e "parar" via Serial.

↳ DICA: defina as velocidades dos motores como 155 na esquerda e 160 na direita.

A cada 100 ms, envie os valores do sensores óticos pela serial.

↳ DICA: use a biblioteca TimerOne.

Aperfeiçoamento



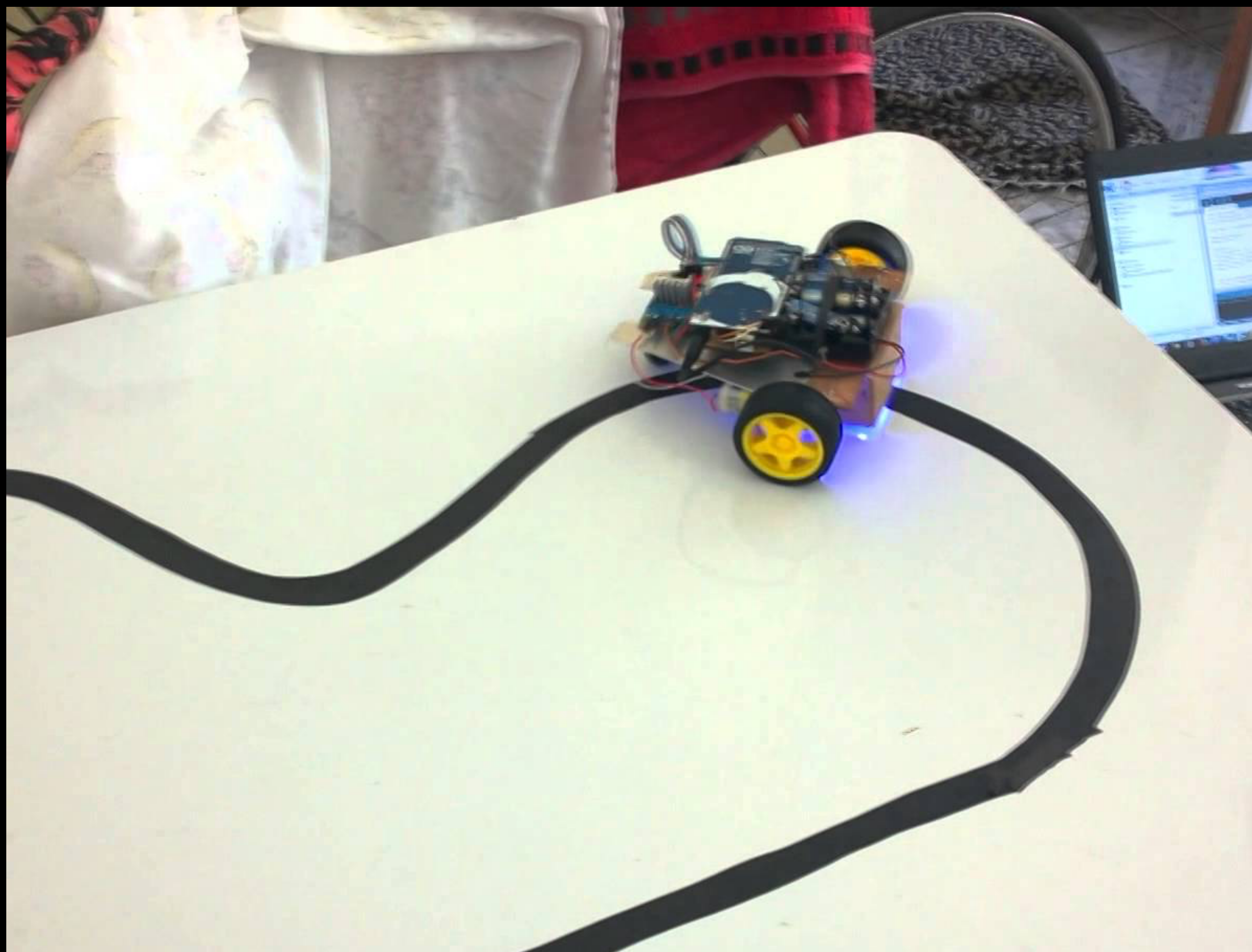
09b_implementacao.ino

cópia
----->



09c_aperfeicoamento.ino

Cópia do Código da Implementação para o Aperfeiçoamento



Seguidor de Linha

sensor 1



sensor 2



Dois Sensores Rastreando a Linha Preta



Aperfeiçoamento 01

Ao apertar o Botão 3, **alterne entre os modos "manual" e "automático"**. Neste último caso, o envio de comandos de direção deve ser bloqueado e o texto "auto" deve ser exibido no display.

Caso os valores recebidos dos sensores fiquem iguais a LOW (cor branca) durante 1 segundo, **volte ao modo manual**.

↪ DICA: use a biblioteca TimerOne, criando e cancelando um timer de acordo com os valores recebidos.



Aperfeiçoamento 02

Ao receber o comando "automático" da serial, controle os motores de modo que o **carrinho mantenha o curso da linha preta**. Garanta que o carrinho nunca fique parado no modo automático.

↳ DICA: controle a direção de acordo com os valores dos dois sensores. Não complique o algoritmo de controle.

Ao receber o comando "manual", **pare o carrinho e volte a obedecer os comandos** anteriores de direção.



janks.link/micro/projeto09.zip

Material do Projeto 09