Projeto 08

Controle Analógico – Prática

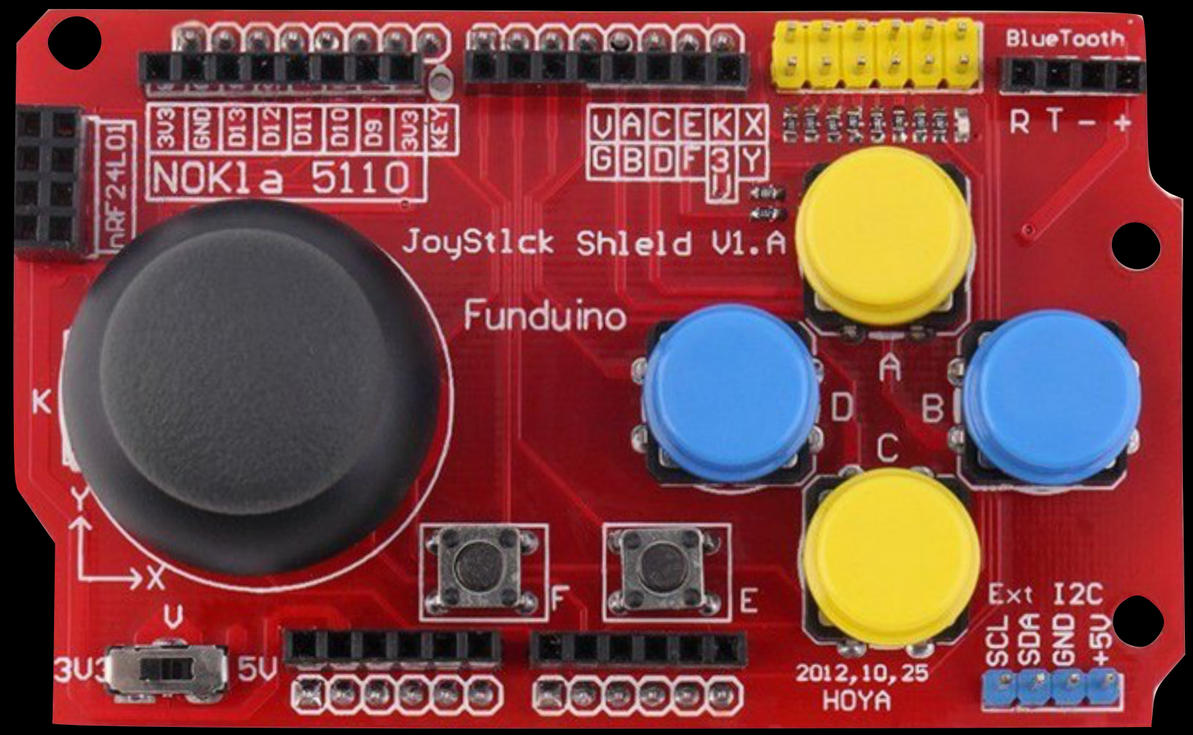
Jan K. S. – janks@puc-rio.br

ENG1419 – Programação de Microcontroladores

Testes Iniciais

A1

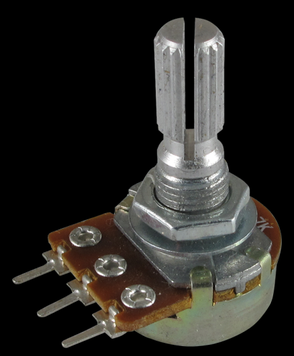
A0



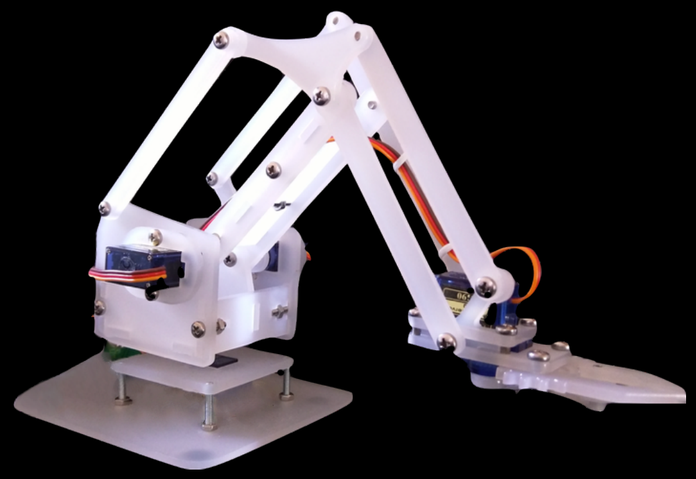
2

5 3

4



A5



base: 12

ombro: 11

cotovelo: 10

garra: 9

Pinos Usados pelos Componentes

```
#include <EEPROM.h>
```

```
#include <Servo.h>
```

~~`#include <meArm.h>`~~

Teste da EEPROM e de Controle de Servo (Sem Usar meArm)

Testes Iniciais

Crie uma variável global indicando quantas vezes o Botão B (Direita) foi apertado, imprimindo a contagem via serial.
↳ DICA: use a `GFButton`.

Ao apertar o Botão B (Direita), salve na EEPROM a contagem no endereço 0. Ao iniciar o programa, carregue essa contagem da memória como o valor inicial.

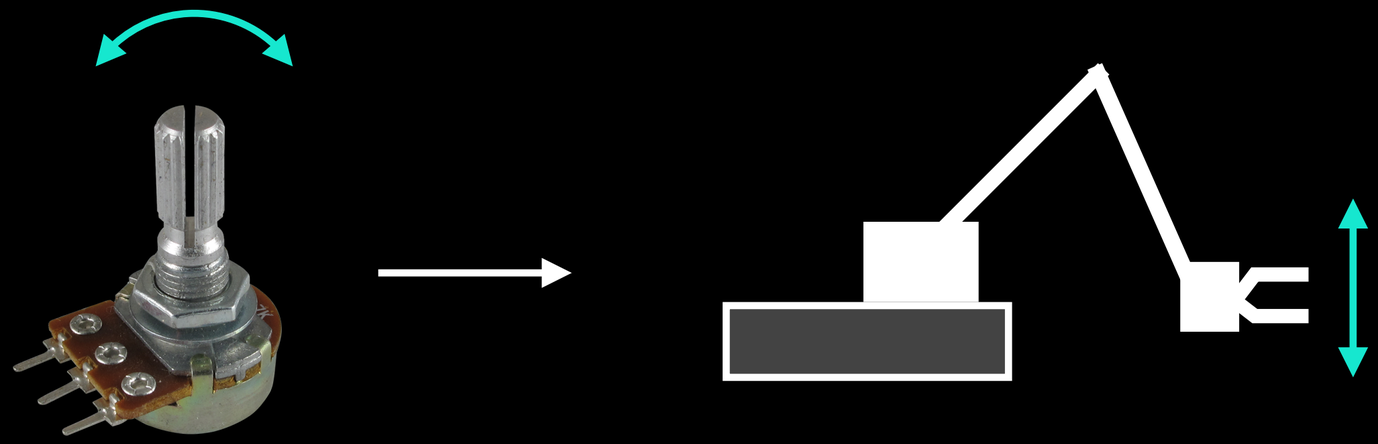
Ao girar o potenciômetro, varie o ângulo do servo da base entre 0 e 180°.
↳ DICA: use a função `map`.

Crie uma variável global para o ângulo do servo do ombro. Se o Botão A (Cima) estiver apertado, diminua essa variável gradualmente até 45. Se o Botão C (Baixo) estiver apertado, aumente essa variável gradualmente até 135.
↳ DICA: use as funções `digitalRead` e `delay` dentro do loop principal. Verifique o valor imprimindo na serial.

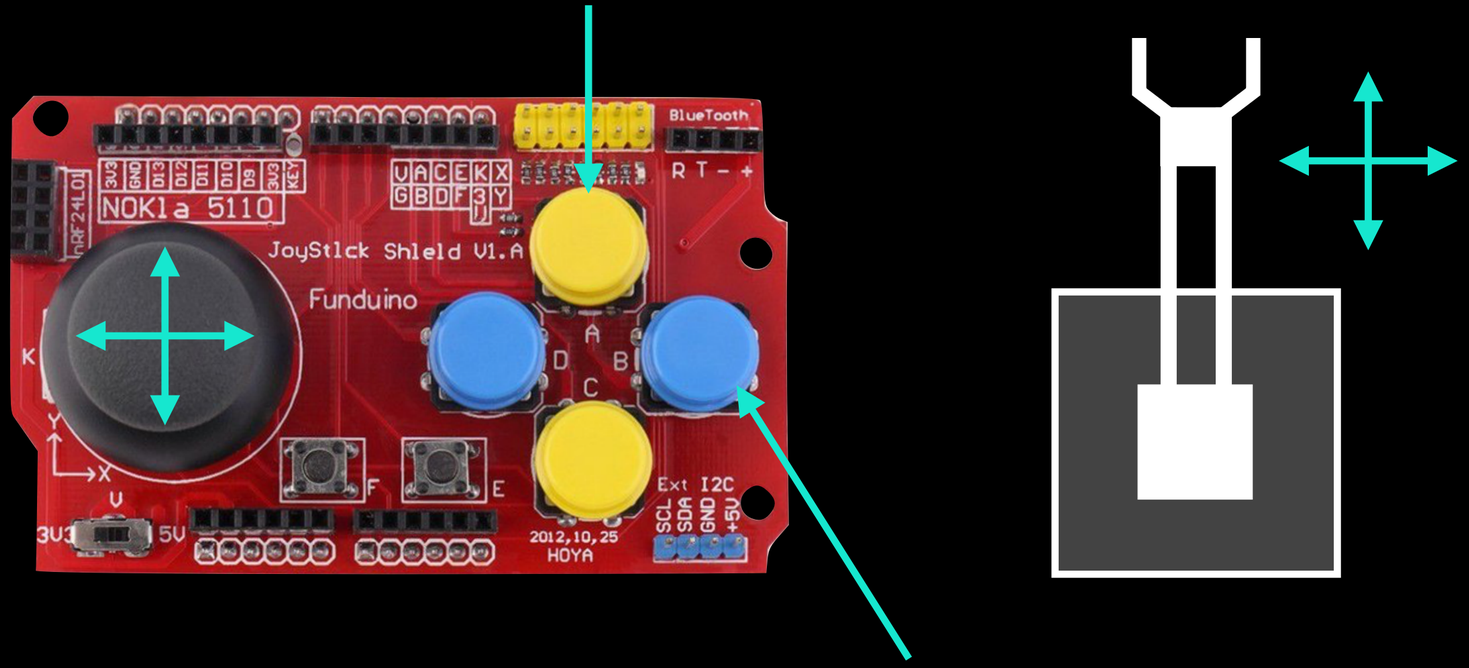
Associe o valor da variável acima ao ângulo do servo do ombro. Teste cuidadosamente o movimento.

Implementação

Controle do Braço Mecânico



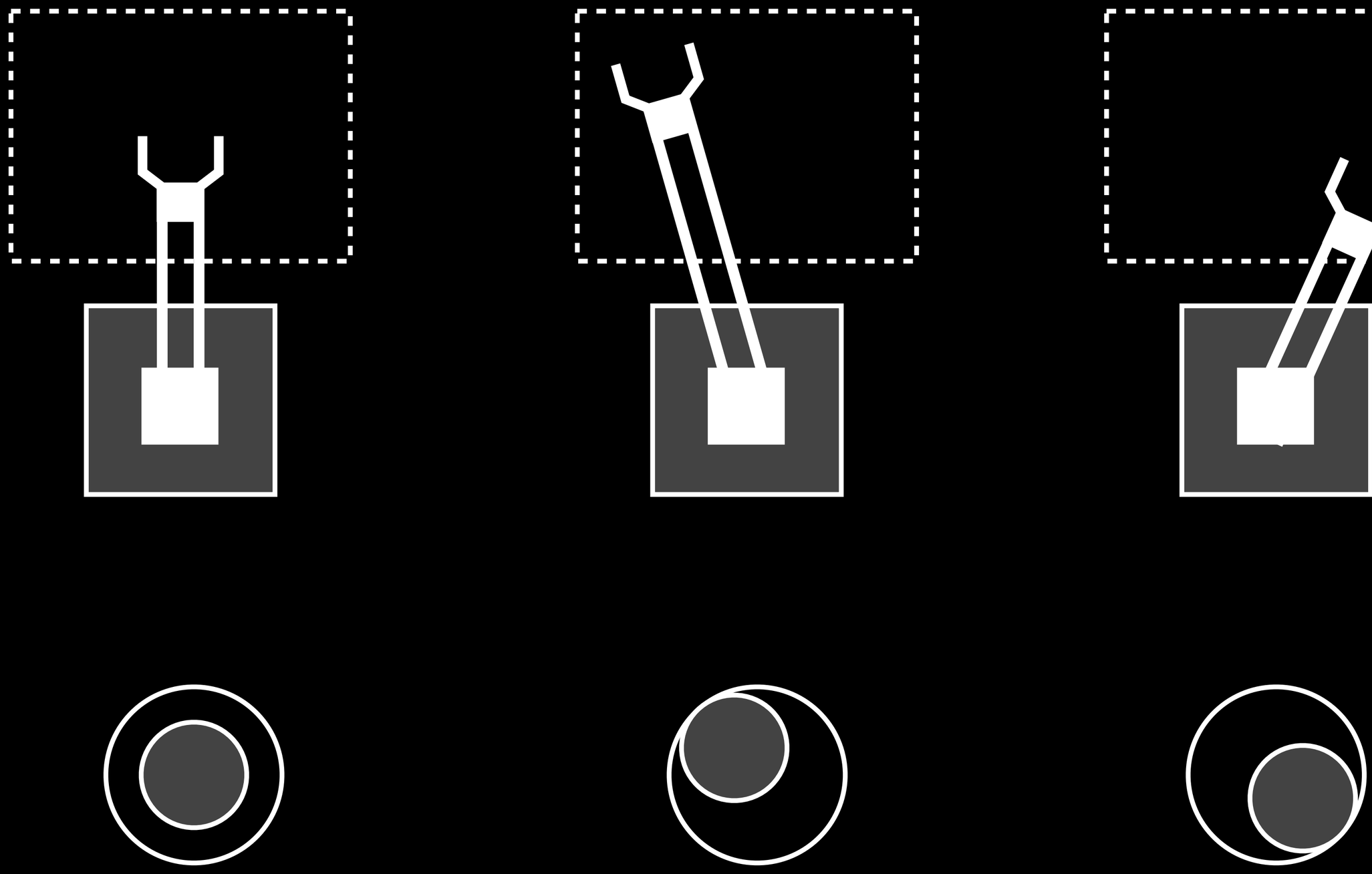
abre/fecha garra



modo absoluto / relativo

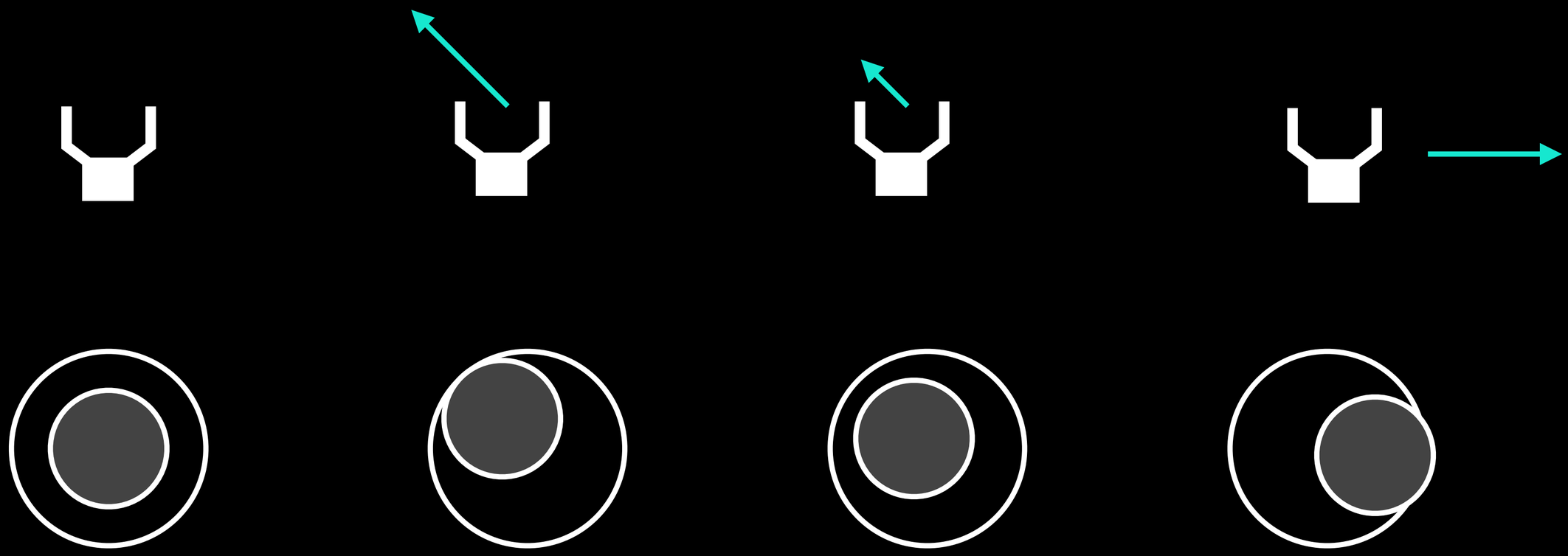
Controle Analógico de 3 Coordenadas

modo absoluto = posição



"Modo Absoluto": Ajuste da Posição

modo relativo = direção

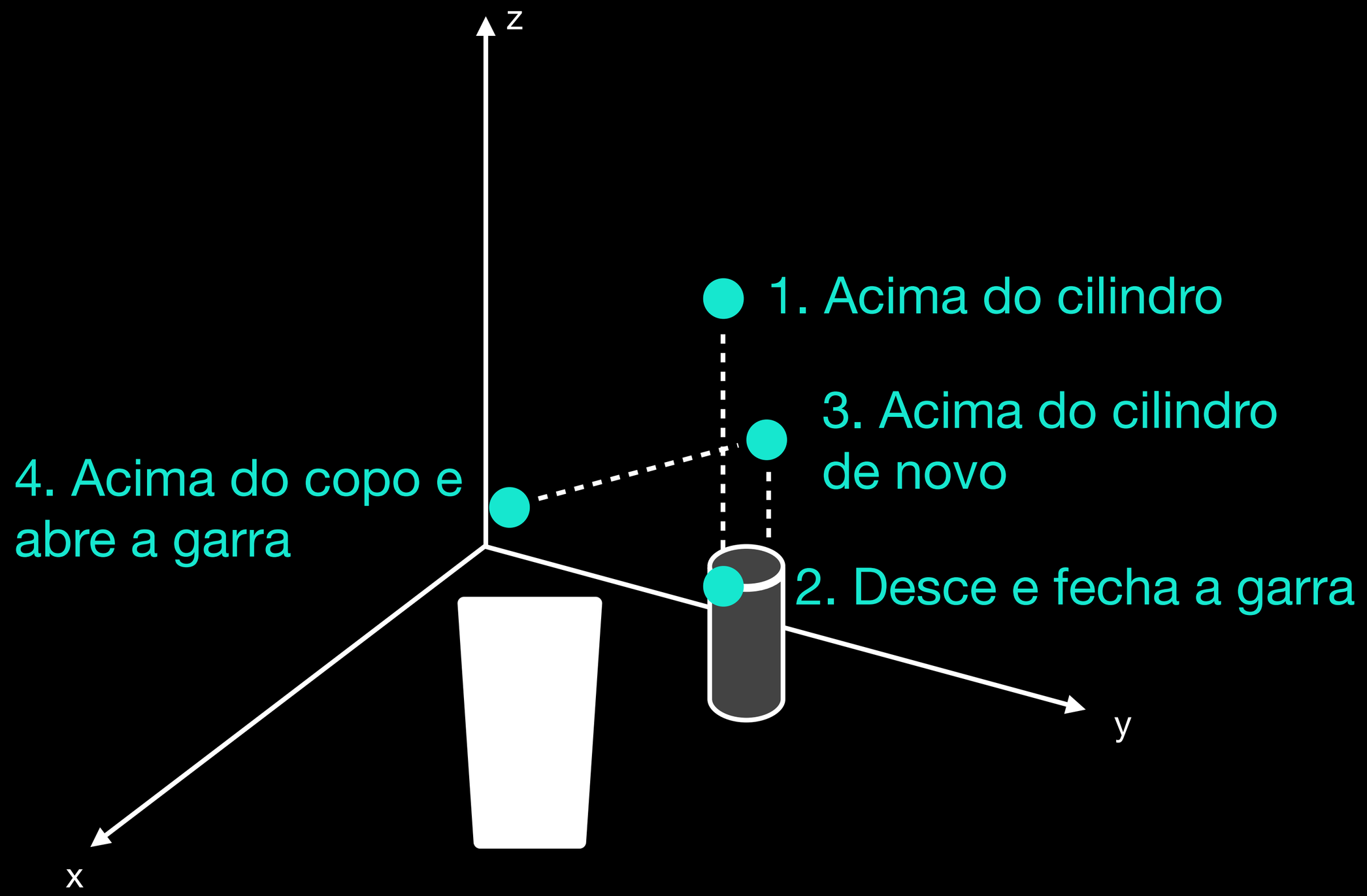


"Modo Relativo": Ajuste da Direção de Movimento

Movimento relativo

1. *Usar variáveis globais para X e Y*
2. *Mapear eixos do joystick para valores entre -10 e 10*
3. *Usar valores como incremento de X e Y, com um delay de 50ms*
4. *Garantir que X e Y não ultrapassem os limites do braço*
5. *Imprimir variáveis pela Serial, verificando os valores*
6. *Usar X e Y para posicionar a garra*

Algoritmo para Movimento Relativo



Trajeto Desejado: Colocar o Pino dentro do Copo

Implementação

Ao iniciar o programa, mova suavemente a garra para o ponto (0, 100, 0).
↳ DICA: use a biblioteca `meArm`.

Ao apertar o Botão A (Cima), abra a garra. Ao soltar, feche a garra.

Ao mexer no potenciômetro, varie a altura do braço entre -30 e 100.

Ao mexer no joystick, mova a garra no plano XY no "modo absoluto", variando X entre -150 e 150 e Y entre 100 e 200.

Ao apertar o Botão B (Esquerda), alterne o movimento do joystick entre "modo absoluto" e "modo relativo". Neste último caso, ajuste as variáveis globais X e Y seguindo o algoritmo indicado.

Associe as variáveis do item anterior à posição do braço.

Use o braço para mover o cilindro para o copo.

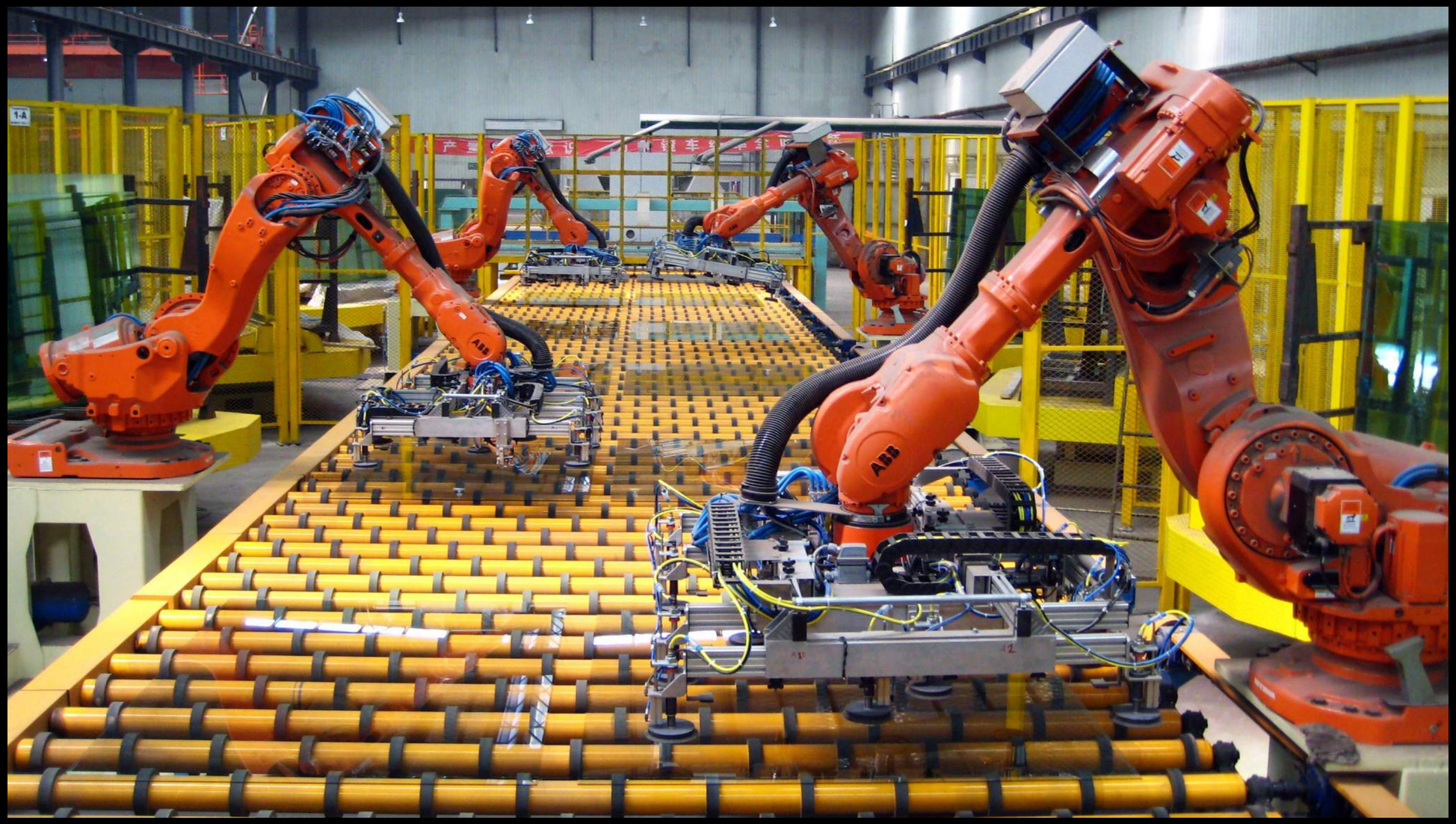
Aperfeiçoamento

08b_implementacao.ino

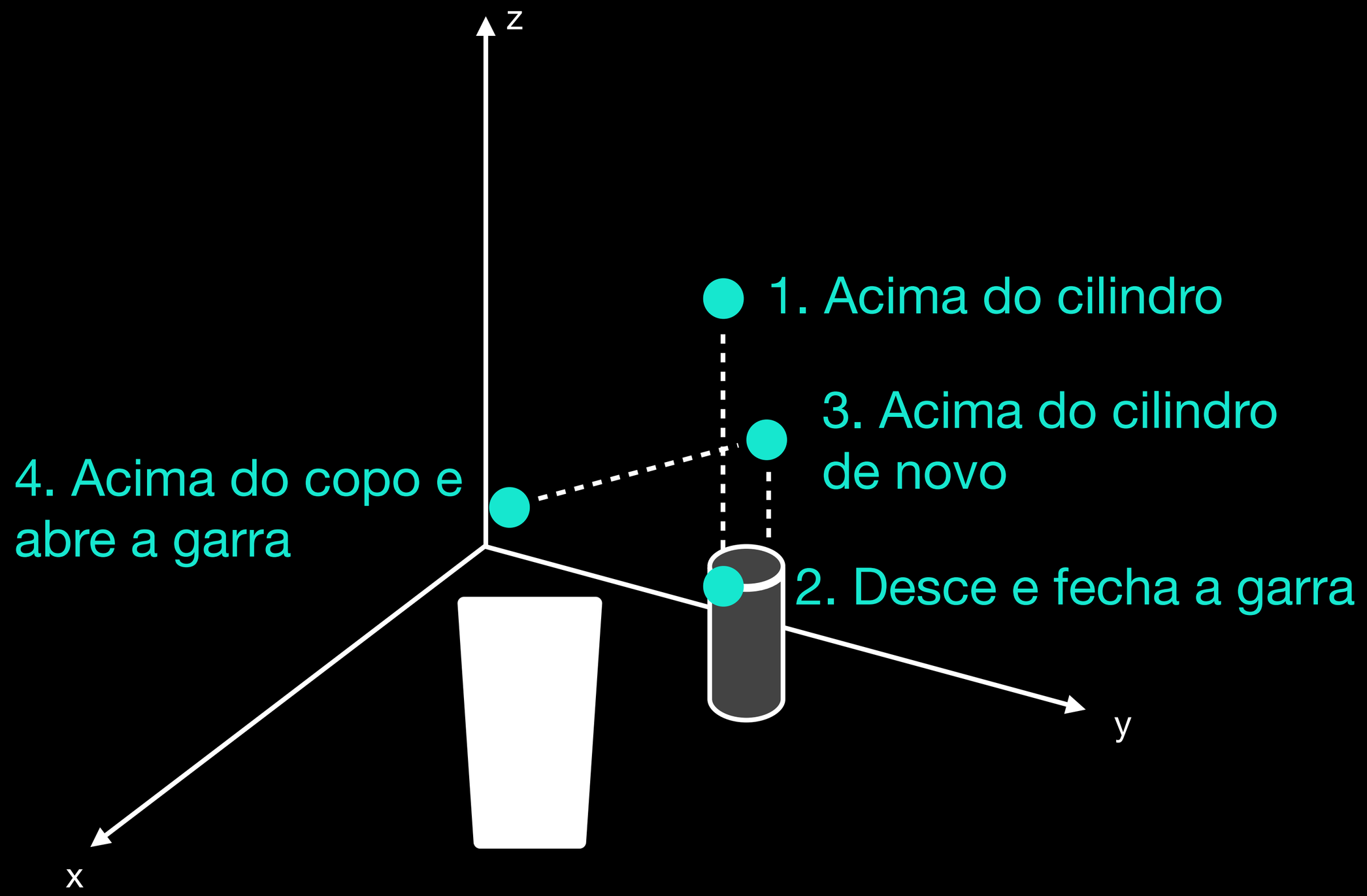
cópia

08c_aperfeicoamento.ino

Cópia do Código da Implementação para o Aperfeiçoamento



Controle Automático do Braço



Trajeto Desejado: Colocar o Pino dentro do Copo

Botão C (Baixo)

move o braço para a posição

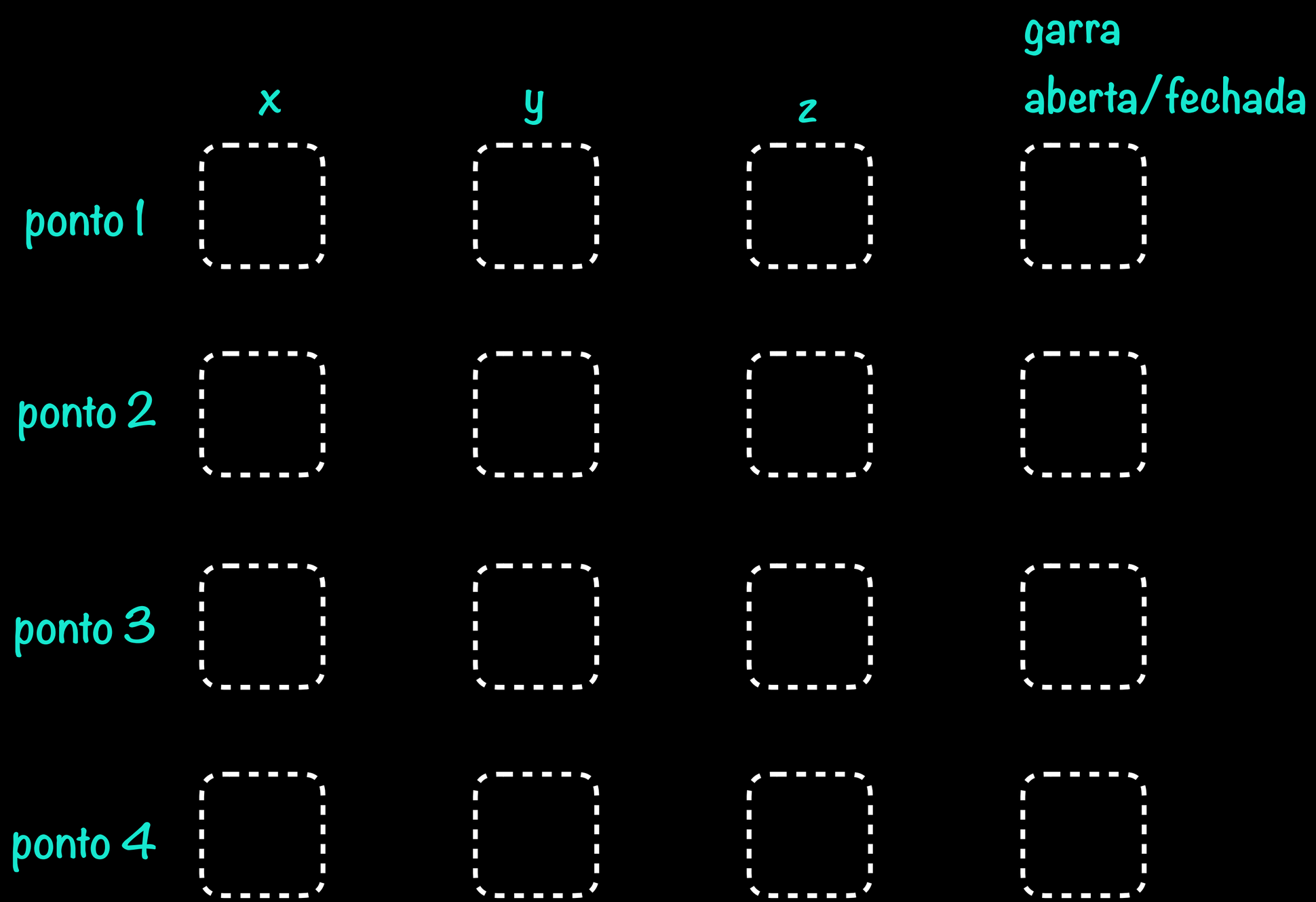
salva coordenadas e garra na matriz (linhas de 0 a 3)

Botão D (Esquerda)

move braço para as coordenadas salvas

Trajeto Desejado: Colocar o Pino dentro do Copo

```
float pontosSalvos[4][4];
```



Armazenamento dos Pontos

Aperfeiçoamento

Ao apertar o Botão C (Baixo), salve as coordenadas e o estado da garra (aberto/fechado) na matriz 4x4, variando o ponto atual de 0 a 4.

Ao apertar o Botão D, mova o braço para cada uma das 4 posições salvas, com intervalos de 500 ms entre cada ponto.

Ao salvar o ponto, guarde a matriz dentro da EEPROM. Ao iniciar o programa, carregue a matriz da EEPROM.

janks.link/micro/projeto08.zip

Material do Projeto 08