# OBJETIVO

Realizar a programação e movimentação do robô no simulador, realizando comandos de acionamentos elétricos e monitoramento de entradas digitais

Diagram

Description automatically generated A picture containing microscope

Description automatically generated

# LISTA DE MATERIAIS

Software EPSON RC+ 7.0

# PROCEDIMENTOS

Realize o download dos arquivos **Lab5.zip** e **Lab\_Garra.zip**

Feche o software **EPSON RC+** e descompacte os arquivos usando software apropriado (Winrar, 7Zip, etc), e copie o primeiro arquivo (**Lab5)** para a seguinte pasta:

**Lab5** -> C:\EpsonRC70\projects

Graphical user interface, application

Description automatically generated

Abra o software **EPSON RC+ 7.0**

Vamos adicionar o novo simulador. Selecione o botão Connection. Graphical user interface, application, chat or text message, website

Description automatically generated

Na tela que aparece, selecione a opção **Add** e em seguida selecione **Connection to new virtual controller** e selecione **OK.**

**Graphical user interface, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated**

O controlador **Virtual 1** será adicionado à lista. Renomeio para **Lab\_Garra.** Selecione **Apply** e em seguida **Close.**

**Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated**

Copie o segundo arquivo (**Lab\_Garra)** para a seguinte pasta. Substitua todos os arquivos que forem pedidos.

**Lab\_Garra** -> C:\EpsonRC70\Simulator

Graphical user interface, text, application, email

Description automatically generated

Agora, podemos abrir o projeto Lab5 dentro do **EPSON RC+.**

**Graphical user interface, text, application

Description automatically generated Graphical user interface, application

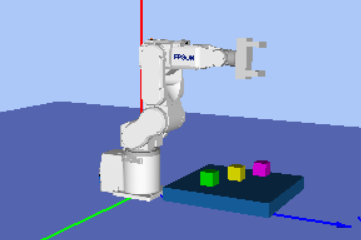
Description automatically generated**

Conecte ao simulador **Lab\_Garra** que acabamos de criar.

**Graphical user interface, table, Word

Description automatically generated**

Abra o simulador. Caso alguma mensagem seja apresentada na tela, pressione **No (Não).** Seu simulador deverá conter uma garra na ponta do robô, 3 blocos coloridos e um plano. (Talvez o bloco verde apareça já na garra)



Agora, vamos nomear as entradas e saídas do programa, como vimos na teoria. Selecione o botão **I/O Label Editor.**

**Graphical user interface, application, Teams

Description automatically generated**

Na tela seguinte, navegue até **Standard I/O -> Inputs -> Bits** e no campo 8, insira o nome **Garra\_fechada.**

**Graphical user interface, application, table, Excel

Description automatically generated**

Ainda na tela do **I/O Label Editor,** navegue até **Standard I/O -> Outputs -> Bits** e no campo 8, insira o nome **fechar\_garra**.

**Graphical user interface, application, table

Description automatically generated**

Feche a janela **I/O Label Editor,** e salve seu projeto.

Execute o programa pelo botão **F5**, ou selecionando o **Run Window.** Deixe o simulador ao lado para que possa ver o que acontece.

**Graphical user interface, application

Description automatically generated**

Perceba, que após o robô se aproximar do bloco verde, uma mensagem aparece na tela.

Graphical user interface, application

Description automatically generated

Como estamos no simulador, vamos simular o sensor de garra fechada, através do **I/O Monitor**

A picture containing logo

Description automatically generated

Abra a janela **I/O Monitor.** Perceba, que a variável **fechar\_garra** está acionada, e a variável **Garra\_fechada** não está.

Dê um duplo clique na bola branca ao lado da variável **Garra\_fechada** para que ela fique vermelha, indicando o fechamento da garra. O mesmo processo deve ser feito para a abertura da garra.

Graphical user interface, application

Description automatically generated

Execute, até que a mensagem **Programa finalizado** apareça na tela.

Graphical user interface, text, application

Description automatically generated

Leia atentamente a função **Main**, com foco aos comentários do programa para entender como foi montado. Perceba que usamos somente comandos apresentados em aula

## Exercício para entrega

Com base no programa apresentado, continue a sua programação para retirar os quadrados da pilha, e voltar eles para as posições originais.

Você consegue montar seu programa, usando os mesmos comandos que já existem, somente mudando a ordem de chamada.

Seu programa **Não** pode apresentar colisões!

Sempre após usar um comando de abertura ou fechamento de garra, use o comando **Wait**  para garantir que a garra está na posição correta antes do próximo movimento. Comente seu código para explicar o uso das funções.