

Guião de Programação e Montagem do Circuito

Na aula de TIC, os alunos iniciarão a projeção do circuito eletrónico com *Arduino* para um caixote do lixo automático. O objetivo desse circuito é auxiliar na separação de resíduos, indicando se o material descartado corresponde ao tipo do caixote do lixo que se pretende criar (plástico, papel, metal ou orgânico).

Por meio do protótipo apresentado, o circuito reconhecerá o tipo de resíduo e ativará um LED correspondente à categoria correta ou emitirá uma sinalização caso o material não seja adequado para aquele caixote do lixo.

Para construir o circuito, sugerimos começar no **Tinkercad**. Ao criar o circuito virtualmente, os alunos terão uma noção clara de possíveis ajustes e poderão identificar o que funcionará ou não antes de implementar fisicamente – figura 1.

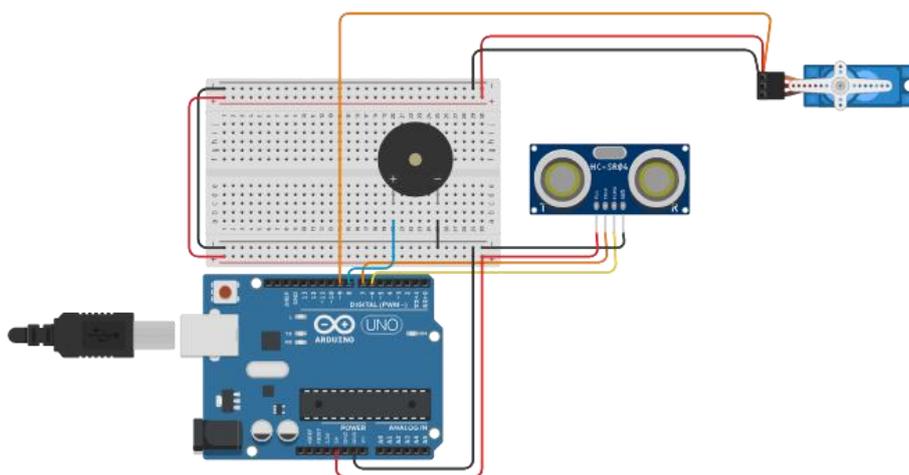


Figura 1 – circuito virtual no Tinkercad

Componentes	Quantidade
<i>Arduino</i> Uno R3	1 unidade
Sensor de distância ultrassónico	1 unidade
Posicional <i>Microservo</i>	1 unidade
Placa de ensaio pequena	1 unidade

Definir variáveis para o código

O código pode ser programado de forma simples. Mas, antes, é importante compreender o princípio de funcionamento que deve ser fornecido ao nosso protótipo. Basicamente, teremos de monitorizar o sensor de distância ultrassónico, conforme ele identifique um obstáculo, que é a própria mão do utilizador do caixote do lixo. O primeiro passo será definir uma *variável*. Entenda a variável como um espaço na memória do *Arduino* que irá permitir que se possa adicionar algum dado. E, para entender ainda melhor, pense que a esse espaço físico na memória nós daremos um nome, da mesma forma que ocorre ao termos um endereço para a nossa casa, com o nome da rua, o número da casa, e etc. Portanto, na opção variáveis, devemos criar uma variável chamada *distância*, conforme a fig. 2

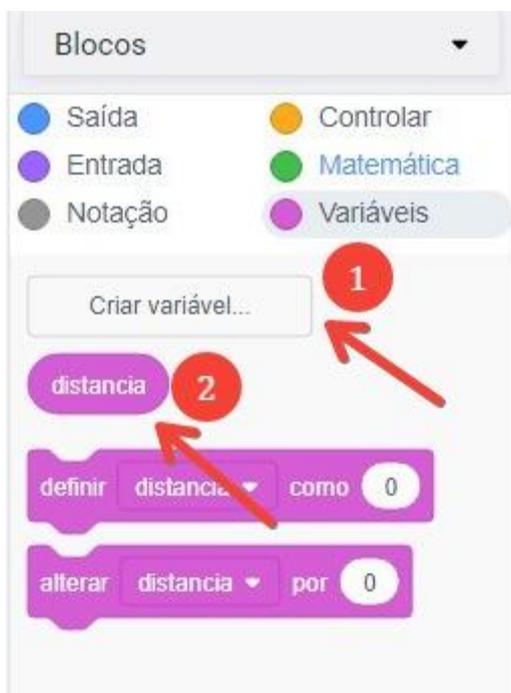


Figura 2 - Criar e manipular a variável "distancia" no Tinkercad

Depois disso, devemos configurar o nosso sensor ultrassónico: Iremos usar o bloco de definir o conteúdo da variável distância, ainda na opção de variáveis, como sendo uma entrada, que será “ler sensor de distância ultrassónico no pino acionador 7 pino eco 6 e unidade em cm”. Esta escolha de pinos foi a mesma que escolhemos para o nosso *Arduino* na montagem que disponibilizamos para cópia no *Tinkercad*. Veja a descrição dos pinos na figura 3.

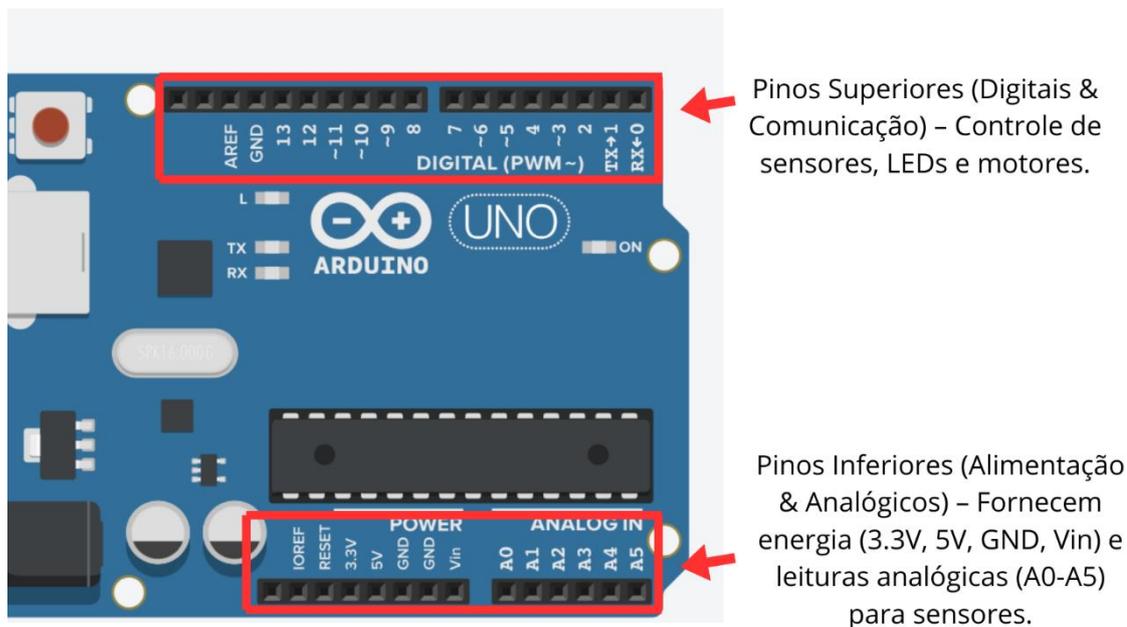


Figura 3 – Simulador Placa Arduino UNO R3 – Descrição dos Pinos Superiores e Inferiores

Programar e entender o código: Teste condicional para acionar ligar o motor

Pois bem, agora é a hora de criarmos o código com a lógica de funcionamento do nosso caixote do lixo. A lógica é simples. Iremos utilizar um sensor de distância ultrassónico, para efetuar a identificação de um obstáculo, que nada mais é do que a mão do utilizador em frente ao caixote do lixo, pronto para colocar o lixo no sítio certo. Portanto, a lógica definida será a seguinte:

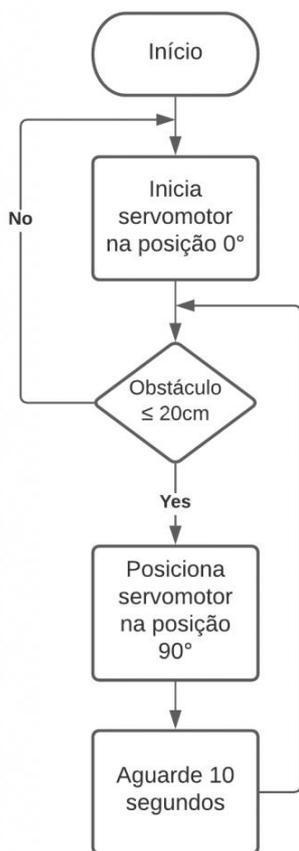


Figura 4 – Fluxograma de funcionamento

Inicialmente, o motor é definido na posição de 0°, que corresponde à posição angular que mantém a tampa do caixote do lixo fechada. O código deverá monitorizar, dentro de um ciclo, se há a identificação de um obstáculo a uma distância inferior a 20 cm do sensor de presença montado no caixote. Caso um obstáculo seja identificado, o motor será acionado para a posição de 180°, mantendo a tampa aberta, e ocorrerá um atraso de 10 segundos antes de se verificar novamente a presença do obstáculo. Apenas quando não houver obstáculo, o código sairá do ciclo de repetição e a posição do servomotor será acionada novamente para a posição inicial de 0°.

Relativamente à condição do ciclo de repetição, esta irá proporcionar um desvio na execução do nosso código, com base na análise da veracidade da condição: enquanto a leitura da distância feita pelo sensor ultrassónico for inferior a 20 cm, o motor deve manter-se na posição de 180° (tampa aberta). Durante todo o tempo, estaremos dentro do ciclo de repetição, efetuando leituras da distância e verificando se esta se mantém inferior a 20 cm. Quando a distância for superior, o

programa sairá do ciclo de repetição e fará com que o motor volte à posição de 0° (tampa fechada). O fluxograma acima é uma boa forma de visualizar este princípio de funcionamento.

O código, programado em linguagem de blocos no caixote do lixo, é apresentado abaixo. Inicialmente, para que seja possível visualizar as mudanças de estado, adicionámos alguns atrasos em pontos específicos do código, não apenas após a abertura do caixote, permitindo assim que o código suspenda a execução por um curto intervalo de tempo, tornando os movimentos mais previsíveis e perceptíveis.

Quanto ao restante funcionamento do código, além da já mencionada configuração da leitura do sensor de distância ultrassónico na variável "distância" e dos atrasos adicionados em pontos estratégicos, implementámos o ciclo de repetição, que se mantém ativo enquanto a distância medida for inferior a 20 cm, conforme já explicado. É essencial atualizar o estado da variável "distância" a cada iteração, para que o código possa ser atualizado periodicamente e possa sair do ciclo assim que a leitura do sensor já não indicar a presença de um obstáculo.

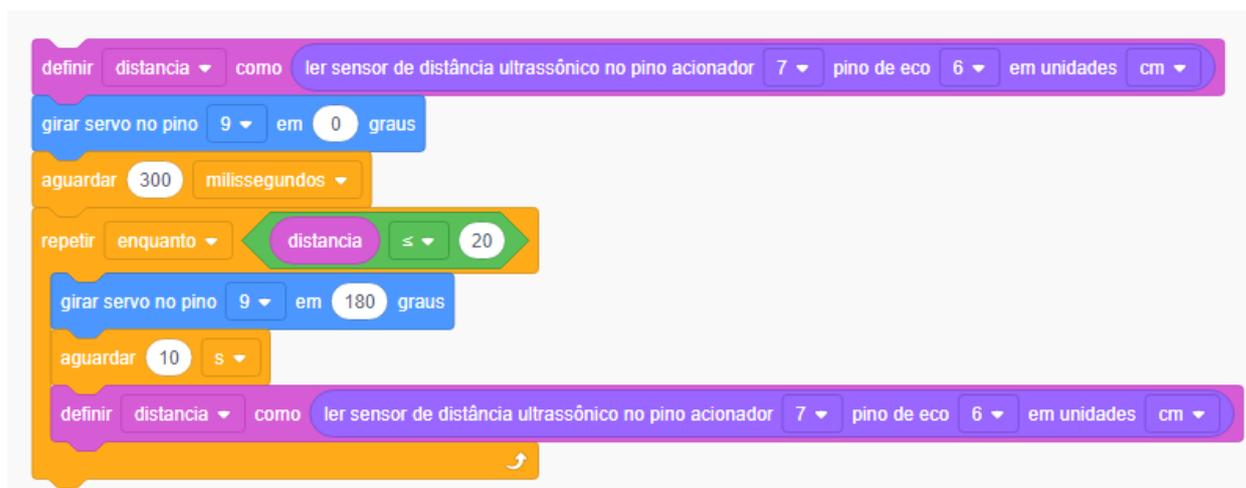


Figura 5 –Código completo por blocos no Tinkercad

Após a montagem do circuito no *Tinkercad*, o próximo passo é reunir todos os materiais necessários para a construção do circuito físico. Certifique-se de que todos os componentes, como fios, sensores, placas de circuito, resistores e o *Arduino*, estejam disponíveis e organizados.

Além disso, prepare o espaço de trabalho de forma que tenha fácil acesso a todos os materiais e um local adequado para realizar a montagem do circuito, para que garanta a segurança e eficiência durante o processo. A organização dos componentes é essencial para evitar erros e garantir que o circuito funcione corretamente na etapa de implementação física.

Preparação do Material:

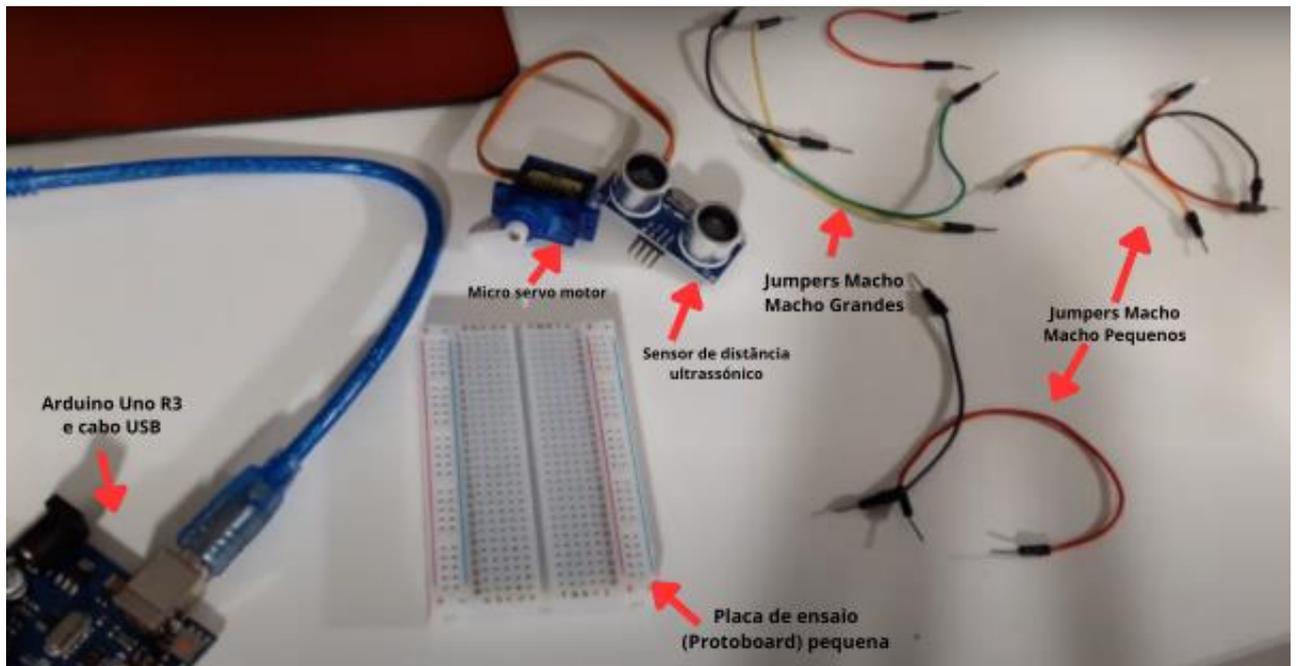


Figura 6 – Preparação dos componentes eletrónicos para a montagem

Certifique-se de ter todos os componentes necessários:

- Arduino (como o Arduino Uno),
- Sensor de distância (HC-SR04),
- Servo motor para abertura da tampa do caixote do lixo,
- Fios de conexão (*jumper wires*), *Protoboard* (placa de ensaio) para fazer as conexões,
- Caixote do lixo (já criado na aula anterior).

Para a montagem, basta seguir o esquema disponibilizado no início da página do projeto, no *site* do *Tinkercad*. Reproduza-o exatamente conforme a simulação, prestando especial atenção aos detalhes da alimentação da placa, do sensor e do motor.

Ao ligar o *Arduino* ao computador, a placa será automaticamente alimentada. Deverá então estabelecer a ligação dos terminais “+” e “-” às trilhas da *Protoboard* e, em seguida, conectá-los aos terminais de alimentação do sensor ultrassónico e do Servomotor, conforme ilustrado nas imagens seguintes:

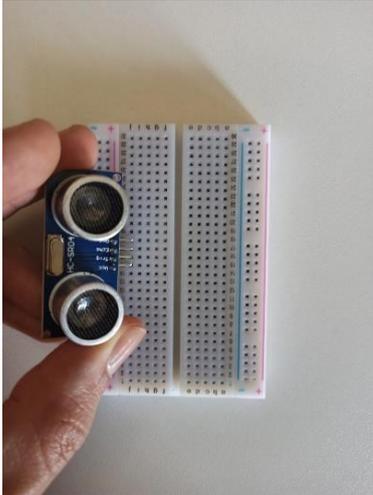


Figura 7 - Forma correta de encaixe do componente sensor

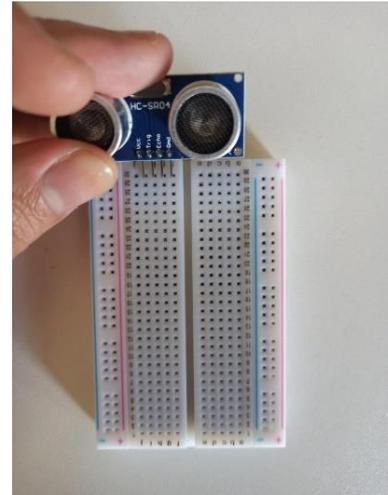


Figura 8 - Forma incorreta de encaixe do componente sensor na Protoboard

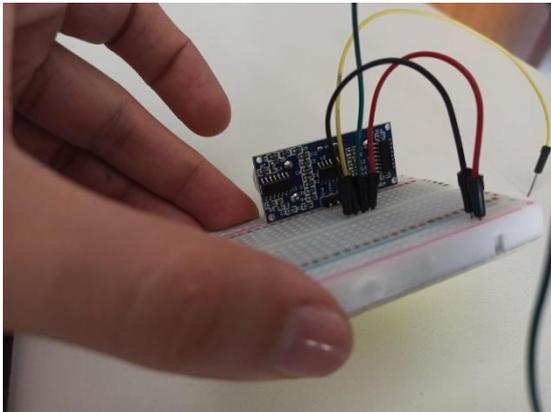


Figura 9 - Conexão dos cabos dos pinos Trigger, Echo e Alimentação (5 V e GND ou + e -) nas trilhas correspondentes da Protoboard.

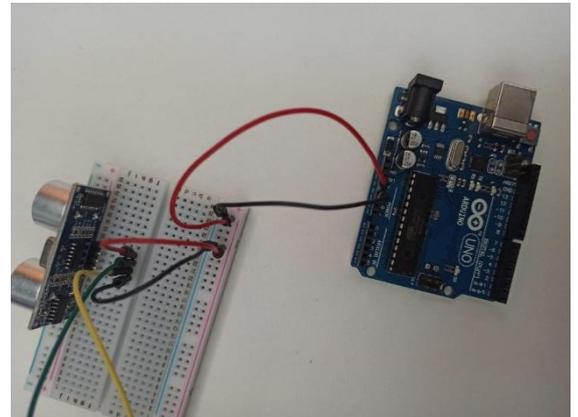


Figura 10 - Conexão da alimentação da placa Arduino para a Protoboard, que liga a energia às trilhas + e -.

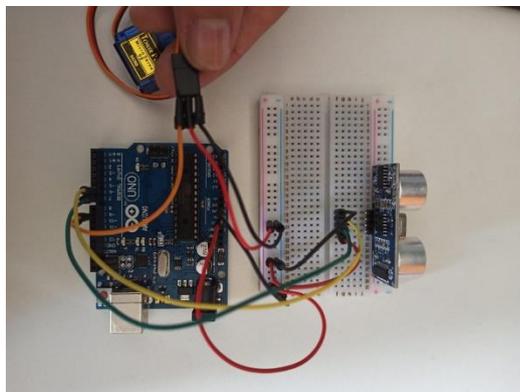


Figura 11 - Ligação do motor às trilhas de alimentação junto a conexão do pino de controlo.

Chegou a hora! Vamos carregar o código e programá-lo no Arduino!

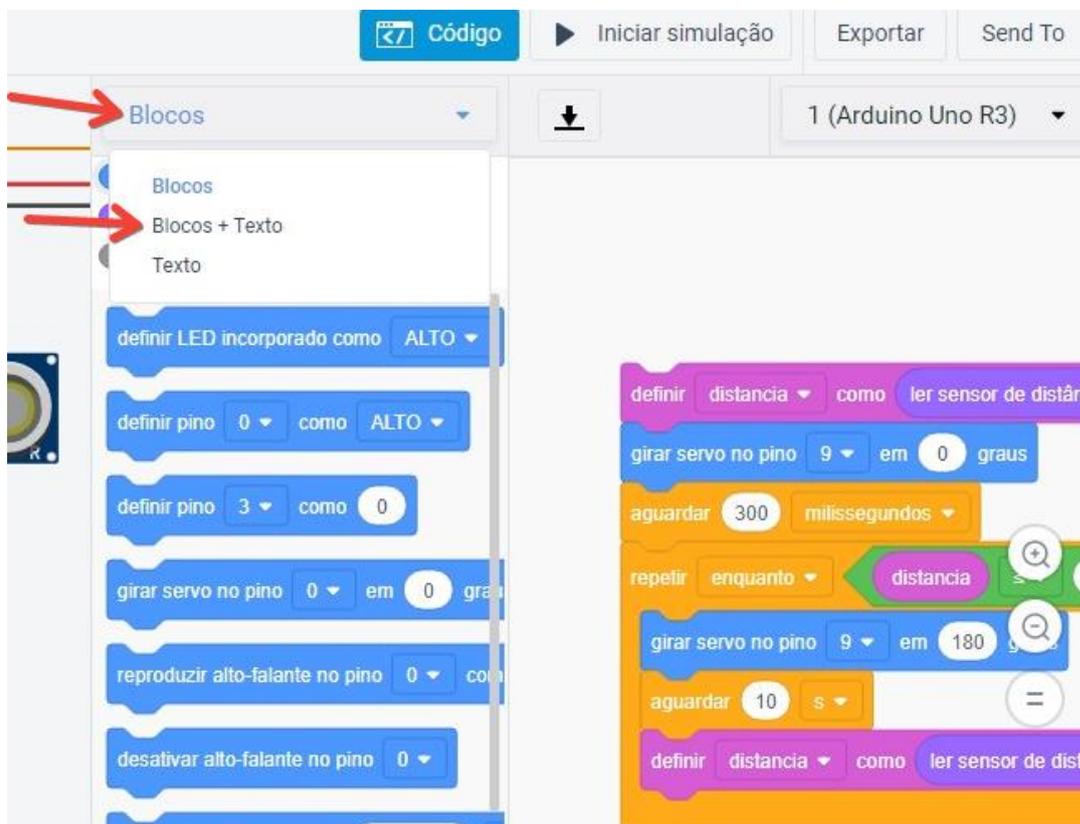
Sim, este é o momento mais esperado! Vamos ver na prática o funcionamento do protótipo que dará vida ao nosso Caixaote do Lixo Inteligente!

Aqui está o grande truque: depois de programar o código em linguagem de blocos, poderá convertê-lo para a linguagem textual necessária para carregar o *firmware* no Arduino.

Neste caso, será necessário converter a linguagem de blocos para a linguagem nativa da IDE do *Arduino*, que é baseada quase integralmente em C++.

Para isso, basta selecionar a opção **Blocos + Texto** no menu, copiar o código gerado e colá-lo na IDE do *Arduino*, num novo projeto – conforme a figura 12

Simple, não é?



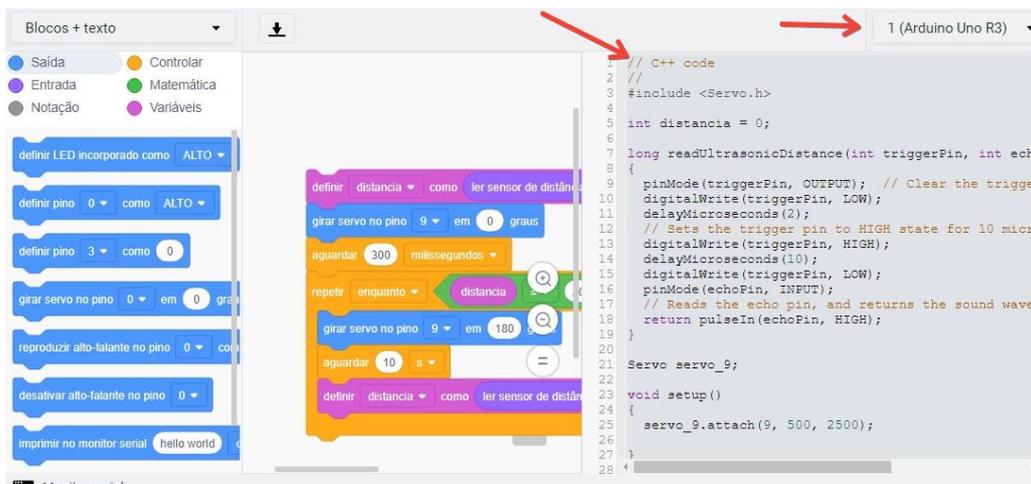


Figura 12 – Copiar código em texto do Tinkercad para Arduino

No Arduino:



Figura 13 – Código em texto que foi copiado no Tinkercad para o Arduino IDE

Uma possível melhoria para o projeto seria adicionar um *display* ao caixote do lixo (de acordo com a figura 14), integrado a sensores de nível de capacidade. Esse sistema indicaria visualmente o status do caixote do lixo: uma luz verde acenderia quando estivesse vazia, uma luz laranja ou amarela sinalizaria que está com metade da capacidade, e uma luz vermelha alertaria que está cheia.

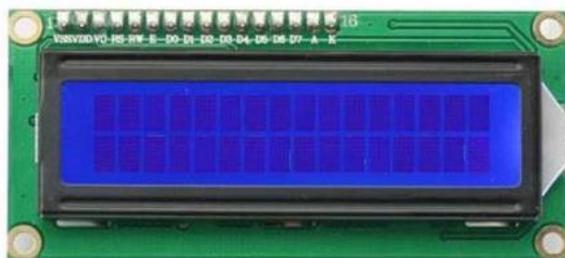


Figura 14 –Display LCD

Teste e Ajustes

Teste o funcionamento do circuito. Verifique se o sensor deteta corretamente a aproximação de objetos e se o servo motor abre e fecha a tampa conforme esperado. Se necessário, faça ajustes no código ou na posição do sensor e do motor servo para otimizar o desempenho, pois poderá haver alguns *delays* a corrigir.

Finalização

Depois de garantir que tudo está a funcionar corretamente, organize os fios e componentes de forma segura e limpa, fixando-os para evitar que se desloquem durante o uso.

Observação: Os modelos encontram-se publicados sob Licenças *Creative Commons*.



Os conteúdos abordados neste documento encontram-se sob a licença *Creative Commons*. Utilização Não Comercial. BY - Os créditos devem ser dados ao autor. NC – Não são permitidos usos comerciais. SA – As adaptações devem ser partilhadas nos mesmos termos.

AUTOR(ES)	DATA
CDI Portugal	Fevereiro/2025