



**TINKERCAD** Neste **guião** vais continuar a aprender **automação**. O objetivo será **simular** a **cancela** de um **parque de estacionamento** que se **abre** quando um **carro** se **aproxima**.

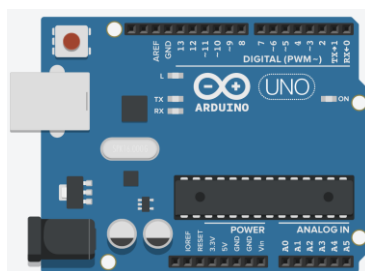
**TINKERCAD** **Todas** as **prints/recortes** pedidos ao longo deste **guião** devem ser **colados** no **documento Diário de Bordo 7ºX** que tens no teu **drive**. **Adiciona** um **título** ao **final** do teu **documento** de nome: **Guião 3 - Servomotores e Sensores**

**TINKERCAD** Acede a <https://www.tinkercad.com/> e **entra** na tua **conta**.


**TINKERCAD** Na tua **conta** **escolhe**, no **menu** do **lado esquerdo** a **opção** .

**TINKERCAD** **Cria** um **novo circuito**. 

**TINKERCAD** **Começa** por **adicionar** um **Arduíno**:



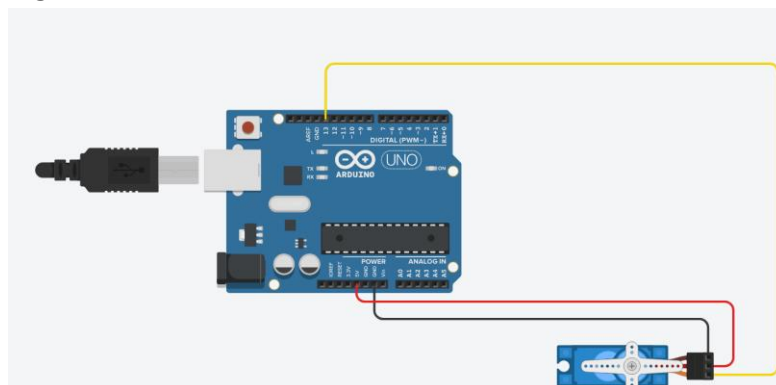
**TINKERCAD** **Adiciona** também um **Micro servo** (servo motor ou motor **passo a passo**). Este tipo de **motores** são de **alta precisão** e **conseguem mover-se** com muita **fineza** (em graus) de **acordo** com o que for **programado**.

**TINKERCAD** Gira  o **servomotor** para que **fique** na **horizontal**.



Como **podes verificar** o **pino superior** terá de **ligar** ao **negativo** e o do **meio** ao **positivo**. O **sinal** será **ligado** a uma das **portas do Arduíno** digitais **superiores** para **podermos controlar** o **motor** através dos **blocos de programação**.


**TINKERCAD** Vamos então **ligar** o **motor** ao **Arduíno**.

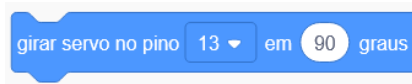


Para **conseguires fazer** estas **curvas** com os **fios** basta **clicar** com o **botão** do **rato** onde **queres** que **faça curva**.

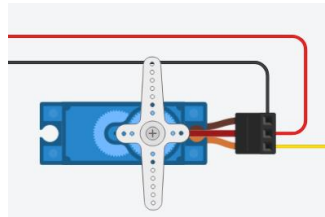


**Resumindo** ligamos o pino **negativo** do motor ao **GND** do **Arduíno**, o pino **positivo** do motor aos **5V** do **Arduíno** e finalmente o pino do **signal** do motor à **pino 13** do **Arduíno**.

Vamos **programar** este **motor** para se **mover**. Para isso **clica** em . **Adiciona** o seguinte **bloco**.

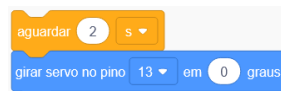


Inicia a **simulação**  e **visualiza** o **motor** a fazer uma **rotação** de **90º**.



Imagina que este **motor** estava **ligado** a uma **cancela** num **parque de estacionamento**. Já **conseguíamos abrir** a **cancela**. 1º passo conseguido!

Vamos agora, depois de **2s** **aberta** **fechar** a **cancela**. **Adiciona** os seguintes **blocos** ao **anterior**.



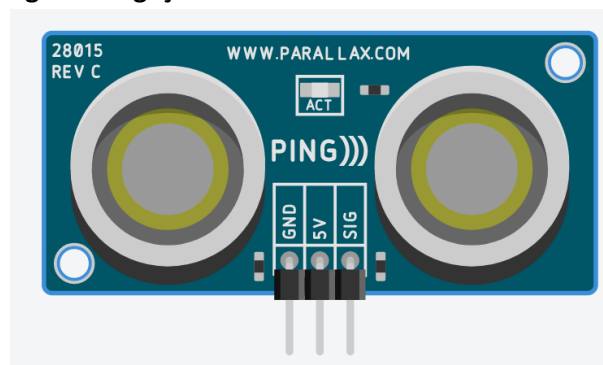
**Faz um print/recorte** que apanhe o **Arduíno** e o **motor** com a **cancela aberta** e os **blocos de programação**. **Cola-o** no teu **diário de bordo**.

Bem **2º passo conseguido**. Porém desta **forma** o **sistema** estava **sempre** a **abrir** e a **fechar** **sem lógica**. Para darmos **lógica** ao **projeto** vamos **adicionar** um **sensor**. Neste caso um **sensor de proximidade** ou **distância**.

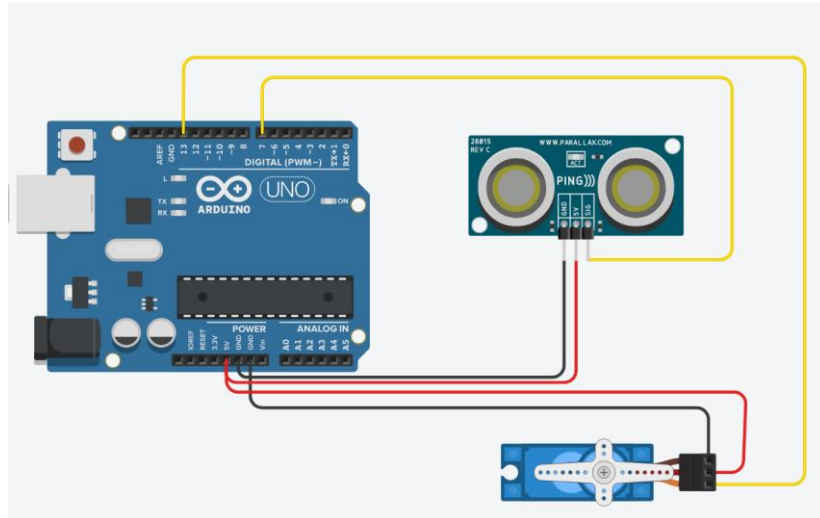


*Sabias que o Arduíno tem ao seu dispor outros sensores como de humidade, fogo, fumo, temperatura, etc. Imagina o que poderias fazer com todos estes sensores ao teu dispor!*

Como **podes** ver na **figura** as **ligações** estão **escritas** no **sensor**.



**Liga-o** como se **mostra** na **figura** seguinte.

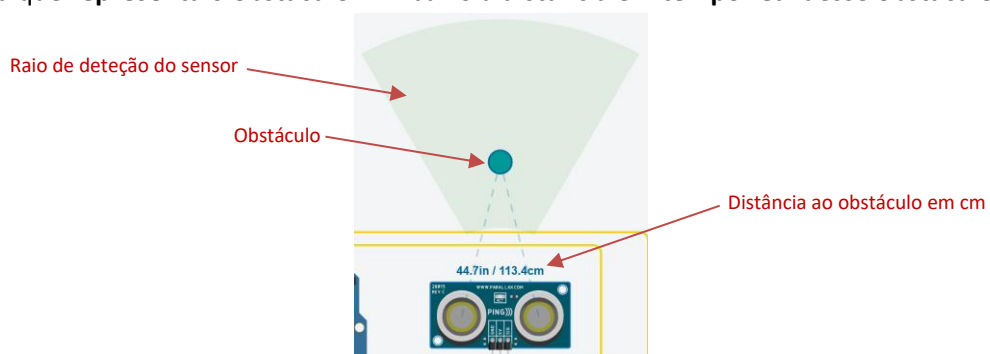


Resumindo ligamos o **pino negativo** do sensor ao **GND** do **Arduino**, o pino **positivo** do sensor aos **5V** do **Arduino** e finalmente o pino do **signal** do sensor à **pino 7** do **Arduino**.

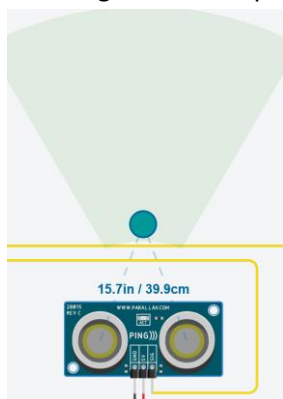


Mas como funciona este sensor? Começa por [▶ Iniciar simulação](#).

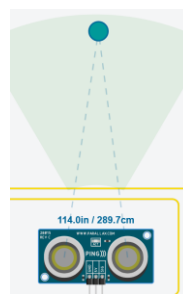
Posteriormente clica no sensor. Irás ver o **raio de deteção** do sensor a **verde claro** e uma **bola** que representa o **obstáculo**. Em baixo a **distância em tempo real** desse **obstáculo**.



Mexe agora na **bola** para mais **perto** do sensor.



ou para mais **longe**



e verás a **distância em cm**

**mudar**.



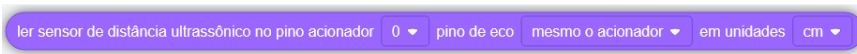
Bem, agora é só **imaginar** que a nossa **bola** representa um **carro** a **aproximar-se** do **parque de estacionamento**. Quando a **distância** for de **pelo menos 70 cm** fazemos a **cancela** **abrir** caso **contrário** ela mantém-se **fechada**.



Abre o [Código](#).

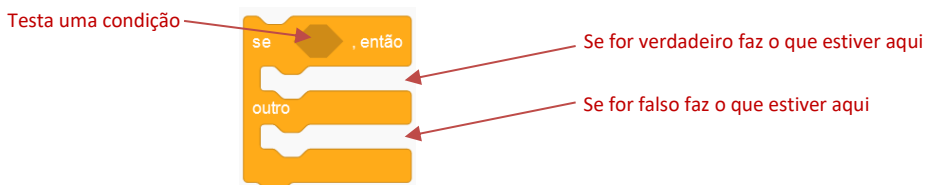


**T I N**  
**K E R**  
**C I A D** Ao **contrário** do que temos **feito** até **aqui** os **sensores** são **entradas** e não saídas. **Enquanto** que **acender** um **LED** ou **girar** um **motor** é **programar** um **pino** para **saída** num **sensor** **queremos** **saber** o que ele está a **ler**, **logo** será uma **entrada**. Acede à categoria Entrada  
De seguida **escolhe** o seguinte **bloco**



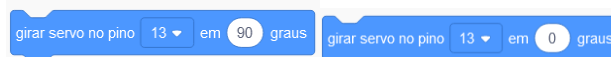
Qual será o **pino** a ler? Onde está **ligado** o **sensor**. Então **qual** é esse **pino**?

**T I N**  
**K E R**  
**C I A D** Mas onde **encaixar** este **bloco**? Num **bloco** **especial** e muito **importante** para os **programadores** o bloco **Se**.



O **bloco** **Se** testa uma **condição** e se for **verdade** faz o que está **encaixado** no **primeiro** buraco **senão** faz o “**Outro**”. Por exemplo: **Se** o **nível** de **fumo** de uma **sala** for **maior** do que **20** **partículas** então **toca** o **alarme**, **senão** **alarme** permanece **desligado**.

**T I N**  
**K E R**  
**C I A D** O nosso caso é **idêntico**. Se o **carro** **estiver** a uma **distância** **menor** que **70** **cm** então a **cancela** **abre** **senão** **mantém-se** **fechada**. Com os **blocos** **seguintes** tenta **fazer** com que a **cancela** **abra** e **feche** de **acordo** com a **condição** de **70cm**. Tens todos os **blocos** a seguir de que **necessitas** **falta** **encaixar** nos locais **corretos** e **preencher** o valor **numérico**. Qual será?



**T I N**  
**K E R**  
**C I A D** Faz um **print/recorte** que apanhe o **Arduino** e o **motor** com a **cancela** **aberta**, o **sensor** a uma **distância** do **obstáculo** a **menos** de **70** **cm** e os **blocos** de **programação**. Cola-o no teu **diário** de **bordo**. Faz outro **print** igual, mas com a **cancela** **fechada**, o **obstáculo** a **mais** de **70** **cm** de **distância** e cola-o no teu **diário** de **bordo**.