


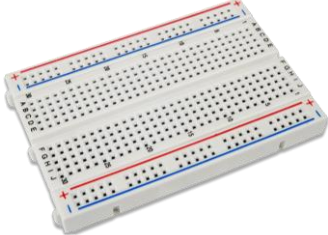








- ∞ Neste **guião** vais continuar a aprender **Arduino**. Neste guião vais controlar um **LED** através de **Bluetooth** utilizando uma **aplicação de telemóvel**.
- ∞ **Todas** as **prints/recortes** pedidos ao longo deste guião devem ser **colados** no **documento Diário de Bordo**. Adiciona um **título** ao **final** do teu **documento** de nome: **Guião 8 Bluetooth**
- ∞ Este **guião** é também **laboratório**, pelo que vais **necessitar** do seguinte **material**:

Arduino UNO 	Resistência 220Ω 
Módulo Bluetooth HC05 ou HC06 	Breadboard 
Cabos tipo Dupont 	Multímetro 
LED 	Smartphone 

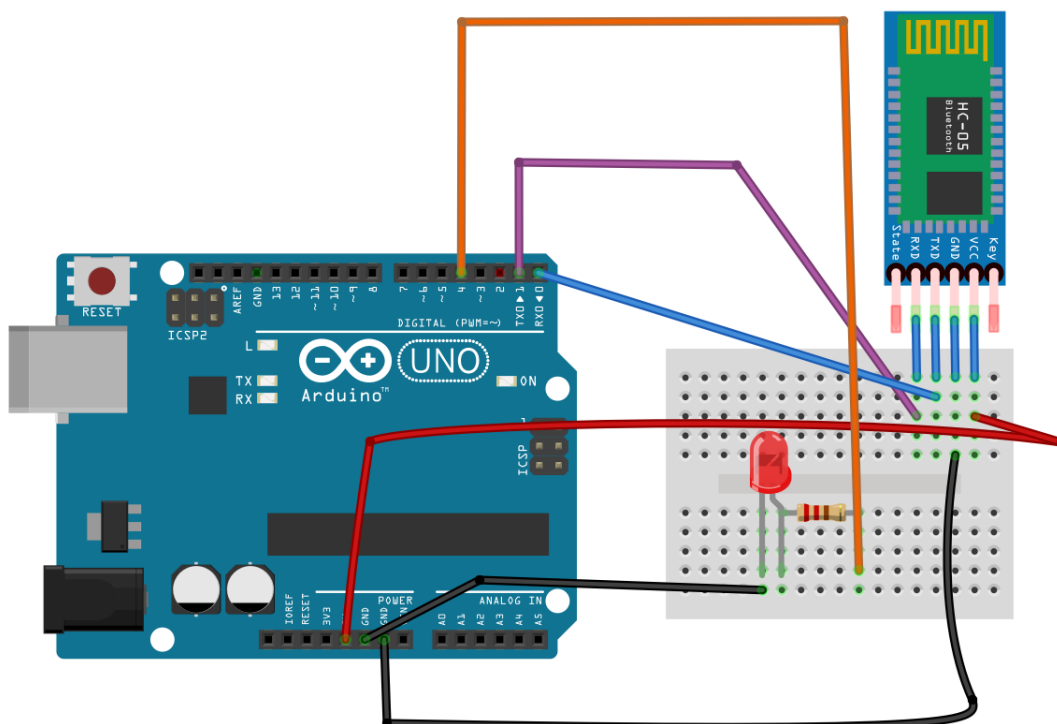
Parte 1

- ∞ Nesta **primeira parte** vamos **realizar** as **ligações elétricas** e a **programação** do **Arduino**. A **lógica** deste **programa** é muito **parecida** com o do **botão de pressão** que **ligava** o **LED**, porém neste programa será um **botão numa aplicação de telemóvel** que **ligará** e **desligará** o nosso **LED** através de **Bluetooth**.
- ∞ Neste **guião** vamos **utilizar** um **componente** que irá **permitir interligar** o nosso **Smartphone** e o **Arduino** utilizando **Bluetooth**. Este componente denomina-se por **HC05** ou **HC06** conforme o **módulo fornecido** pelo teu **professor**. Ficom os **Pinouts** de cada **modelo**:



Pinout do módulo HC05	Pinout do módulo HC06
<p>Utilizar apenas os 4 pinos centrais</p> <ul style="list-style-type: none">• VCC ligado a 5V• GND ligado aos 0V• TX ligado ao RX do Arduino• RX ligado ao TX do Arduino	<p>Utilizar todos os pinos:</p> <ul style="list-style-type: none">• VCC ligado a 5V• GND ligado aos 0V• TX ligado ao RX do Arduino• RX ligado ao TX do Arduino

∞ O esquema de ligações desta atividade não pode ser realizado no Tinkercad porque não existe o componente BT HC05 (ou HC06). Assim terás de fazer diretamente com os componentes físicos. Faz as ligações tal como se mostra a seguir: **Atenção: o esquema utiliza HC05, mas como só utilizamos os 4 pinos do meio deste componente as ligações são exatamente as mesmas para o HC06.**

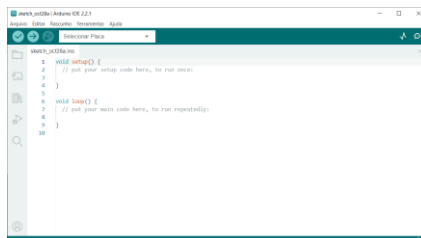


Este esquema foi realizado em **Fritzing** que permite **muito mais componentes** que o **Tinkercad**. **Porém** é mais **complexo** de trabalhar.

Foi ainda ligado um **LED vermelho** com resistência de **220Ω** ao **pino 4**.



- ∞ Como sabes, a **lógica** deste **exercício** é conseguir **ligar/desligar** o **LED** através de uma aplicação de **telemóvel** (será feita mais à frente). Os **comandos** para **ligar/desligar** o **LED** terão de **ser recebidos** através do módulo **Bluetooth HC05/06** e **interpretados** pela **programação** do nosso **Arduino**.
- ∞ Abre o IDE do **Arduino** para **procederes** à **programação** do **lado** do **Arduino**.



- ∞ Começa por definir o **pino 4** onde liga o **LED** e **declara-o** como **OUTPUT**.
Cria, também uma **variável**, de nome **dadosAPP** do tipo **char** que irá **servir** para **guardar** os **dados** recebidos por **BT** enviados pela **APP** que mais à frente irás **fazer**.
Inicializa o **monitor série**.
- ∞ Na função **loop()** adiciona o **seguinte código** e **completa** as **partes** em **falta**.

```
void loop() {  
  if(Serial.available()>0){//caso esteja a receber dados  
    dadosAPP=Serial.read();//lê o que está a receber pela porta série enviado pela APP e guarda na variável dadosAPP  
  
    Serial.println(dadosAPP);//imprime o valor só para controlar pelo monitor série  
    if(dadosAPP=='1'){  
      digitalWrite(ledPin,HIGH);  
    }  
    else if(dadosAPP=='0') {  
      digitalWrite(ledPin,LOW);  
    }  
  }  
}
```

Este **código** **permite guardar** na **variável dadosAPP** aquilo que é **lido** da **porta série**.
Posteriormente é **verificado** se o **valor** dessa **variável** é **1** ou **0** **ligando** ou **desligando** o **LED**.

Parte 2

Atenção: esta parte implica conhecimentos prévios de App Inventor. Podes adquirir estes conhecimentos com os guiões do teu professor sobre a temática.

- ∞ Nesta **segunda parte** temos de **criar** uma **APP** que **enviará ordens**, por **Bluetooth**, para o nosso **Arduino**, que **ligará** ou **desligará** o nosso **LED**. A **lógica** da nossa **APP** é muito **semelhante** à realizada no **guião lanterna**.
- ∞ Começa por **criar** um **novo projeto** de nome **BT**.

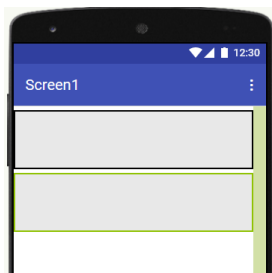
Criar um novo projeto no App Inventor

Nome do projeto:	BT
Toolkit:	Beginner ?
Theme:	Device Default ?

Cancelar OK




- ∞ Arrasta dois organizadores horizontais para a tela do telemóvel e faz com que ocupem toda a largura do ecrã e altura de **20% cada**.

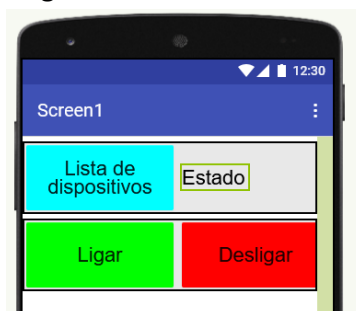


- ∞ Arrasta para o **organizador** mais abaixo **dois botões**. Troca a sua **legenda**, como mostra a **imagem seguinte** e faz com que ocupem **50% da largura** cada um. **Tamanho da fonte 25**.




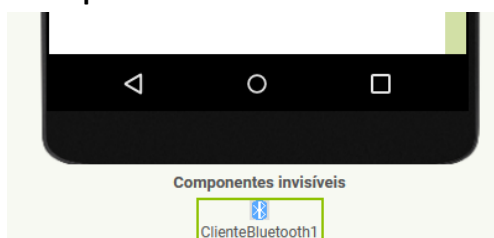
- ∞ No **organizador horizontal superior** iremos colocar a **ligação Bluetooth**. Esta ligação **implica sempre uma lista**. Será nesta **lista** que serão **mostrados os dispositivos** ao alcance do nosso **smartphone**.

Arrasta então uma  Lista do **menu da esquerda** para **organizador superior** e ainda uma **legenda**. A **lista** ocupa **50% da largura**.



- ∞ A **legenda “Estado”** apenas dará **indicação** se o **BT** está **ligado** ou **desligado** conforme se **conseguiu ligar** ou **não** ao **HC05/06** do **Arduino**.

Bem, mas para ter acesso a uma **lista de clientes BT** temos de **inserir um componente BT**. Podemos **encontrar** isso na secção **Conectividade** e arrastando o  ClienteBluetooth para **cima** do **Smartphone**. Ficará como **componente invisível**:





∞ Agora que temos tudo o que **necessitamos** para a nossa **APP**, falta **programar** o que faz **cada componente**. Abre os **Blocos de programação**.

∞ Vamos começar pela **lista de equipamentos BT**. Adiciona o **seguinte bloco**:

```
quando Lista1 .AntesDeEscolher
fazer
```

Este bloco representa o que acontece antes de escolher o **equipamento** a que vamos **ligar**. **Completa** com um **bloco** no seu **interior**, para **obter** a **lista** de **dispositivos Bluetooth** ao alcance do **Smartphone**:

```
quando Lista1 .AntesDeEscolher
fazer
  ajustar Lista1 . Elementos para ClienteBluetooth1 . EndereçosENomes
```

Assim, ao **clikarmos** no **botão lista** de **dispositivos** ser-nos-á mostrada a **lista** de **equipamentos** ao alcance do nosso **smartphone**.

∞ Adiciona os seguintes **blocos** para o que **irá acontecer** após de **escolhermos** um **dispositivo** para nos **ligarmos**:

```
quando Lista1 .DepoisDeEscolher
fazer
  se
    chamar ClienteBluetooth1 .Conectar
      endereço Lista1 . Seleção
    então
      ajustar Legenda1 . Texto para "Ligado"
    senão
      ajustar Legenda1 . Texto para "Não foi possível ligar"
```

Neste caso **depois** de **escolher** um **endereço** será **tentada** uma **ligação** com esse **endereço** (entre o smartphone e o HC05/06) e **caso** tenha **sucesso** **mostrará** na **legenda** a palavra **"Ligado"**, **caso contrário** **mostrará** **"Não foi possível ligar"**.

∞ Bem a **parte** da **ligação** entre a nossa **aplicação** e o **HC05/06** que está **ligado** ao **Arduino** está **concluída**. Agora temos de fazer a **programação** dos **botões (ligar/desligar)**. Ao clicar no botão **ligar** deve ser enviado para o **Arduino**, via **Bluetooth** uma **mensagem** a **dizer** para **ligar** o **nosso LED**. Como vimos na **programação** realizada no **IDE** do **Arduino**, para ligar o **LED** o programa está à **espera** de **receber** um **1** pela **porta série**. Recordando o código:

```
if(dadosAPP=='1'){
  | digitalWrite(ledPin,HIGH);
}
```

∞ Então teremos de **enviar** um **1** pelo **Bluetooth** do **Smartphone** (ou seja pela APP):

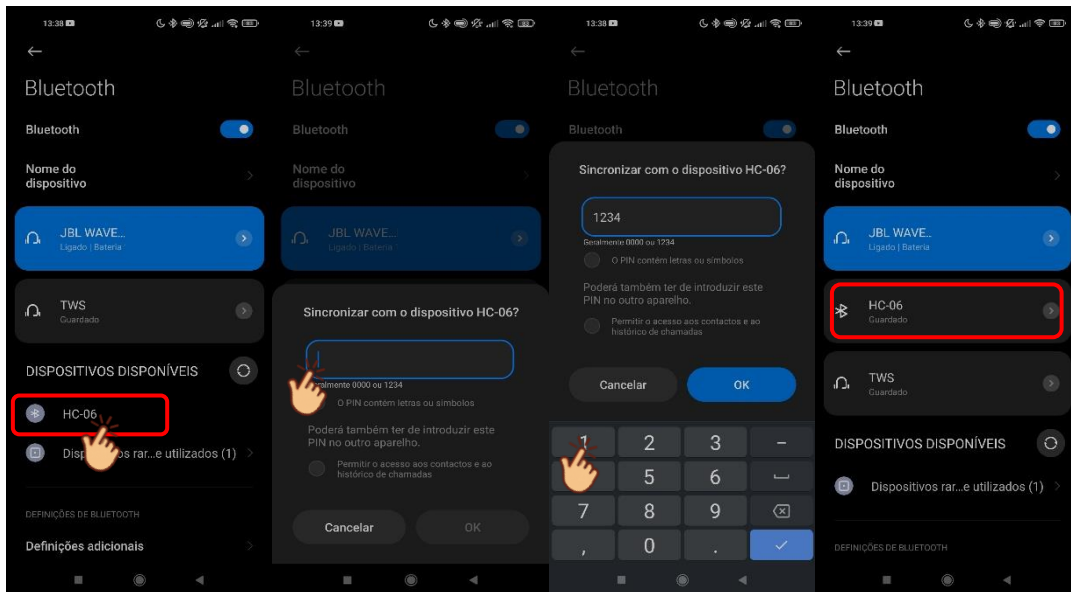
```
quando ON .Clique
fazer
  chamar ClienteBluetooth1 .EnviarTexto
    texto "1"
```

∞ Achas que **consegues** fazer o **botão OFF** (Desligar)?

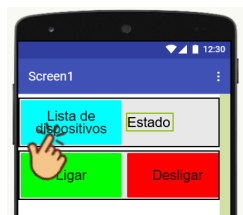
∞ **Instala** a tua **aplicação**.



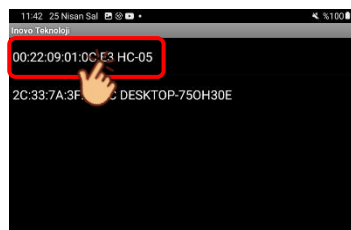
∞ Liga-te ao HC05/06. Na primeira vez tens de autorizar a ligação. Para isso acede ao Bluetooth do teu Smartphone e segue os passos:



∞ Na aplicação clica sobre o botão Lista de Dispositivos:



∞ Irá aparecer uma lista de dispositivos BT ao alcance do teu Smartphone. Deves procurar por HC05/06 e ligar-te:



Se a legenda mudar para Ligado está tudo certo. O LED azul do HC05/06 deve deixar de piscar e fixar.



∞ Envia o teu programa para o Arduino. **Atenção: tens de desligar os cabos TXe RX do Arduino ou não será possível enviar o programa. Depois de enviar o programa já podes voltar a ligar os fios.**

∞ Faz um pequeno vídeo com o telemóvel do circuito + Arduino com os LEDs a ligar/desligar conforme clicas no botão ligar/desligar da APP. Envia para o teu Drive para a pasta Arduino - Vídeos. Altera o nome do ficheiro de vídeo para 8 – Bluetooth.

∞ Tira printscreen da tua APP e envia para o teu diário de bordo.

∞ Tira também uma foto ao circuito e envia para o teu diário de bordo.



-  **Guarda** o **programa**, no IDE do Arduino com o nome **8Bluetooth**. Envia este **ficheiro** para o teu **drive** para a pasta **Arduino - Programas**
-  **Chama** o teu **professor** para **avaliar**.