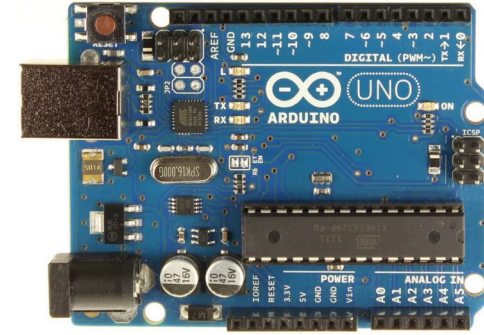


Ambiente (IDE) Arduino

E

LudoBot/ *Itty Bitty Buggy*

Parte II



Esta aula está em:

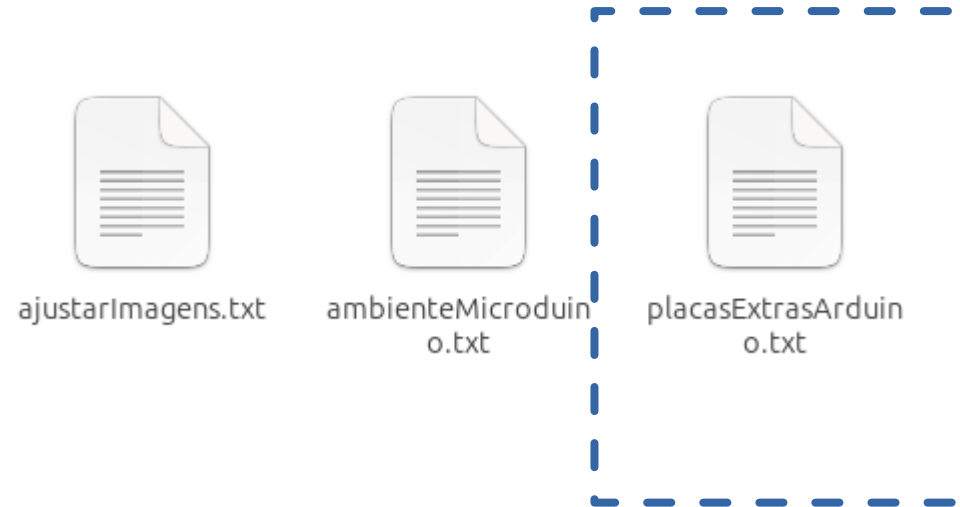
[https://tiaplicada.ufpr.br/wp-content/uploads/
2024/11/aula8novembro.zip](https://tiaplicada.ufpr.br/wp-content/uploads/2024/11/aula8novembro.zip)

Configuração

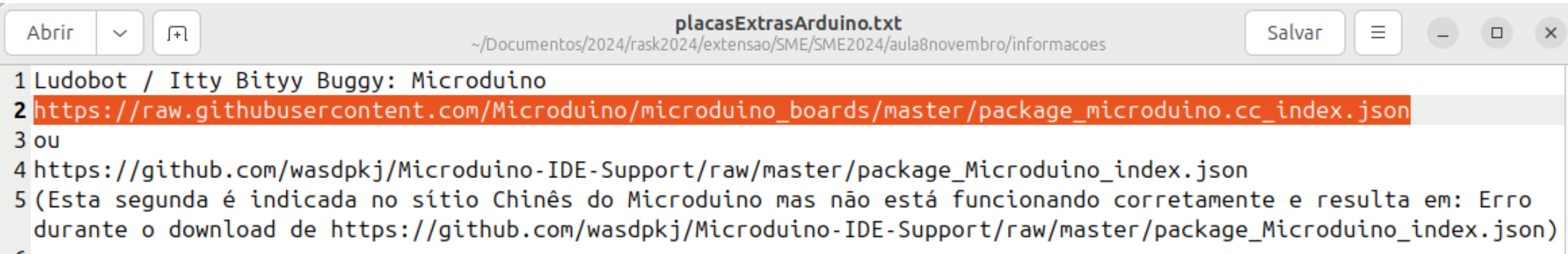
Relembrando a configuração



Relembrando a configuração



Relembrando a configuração

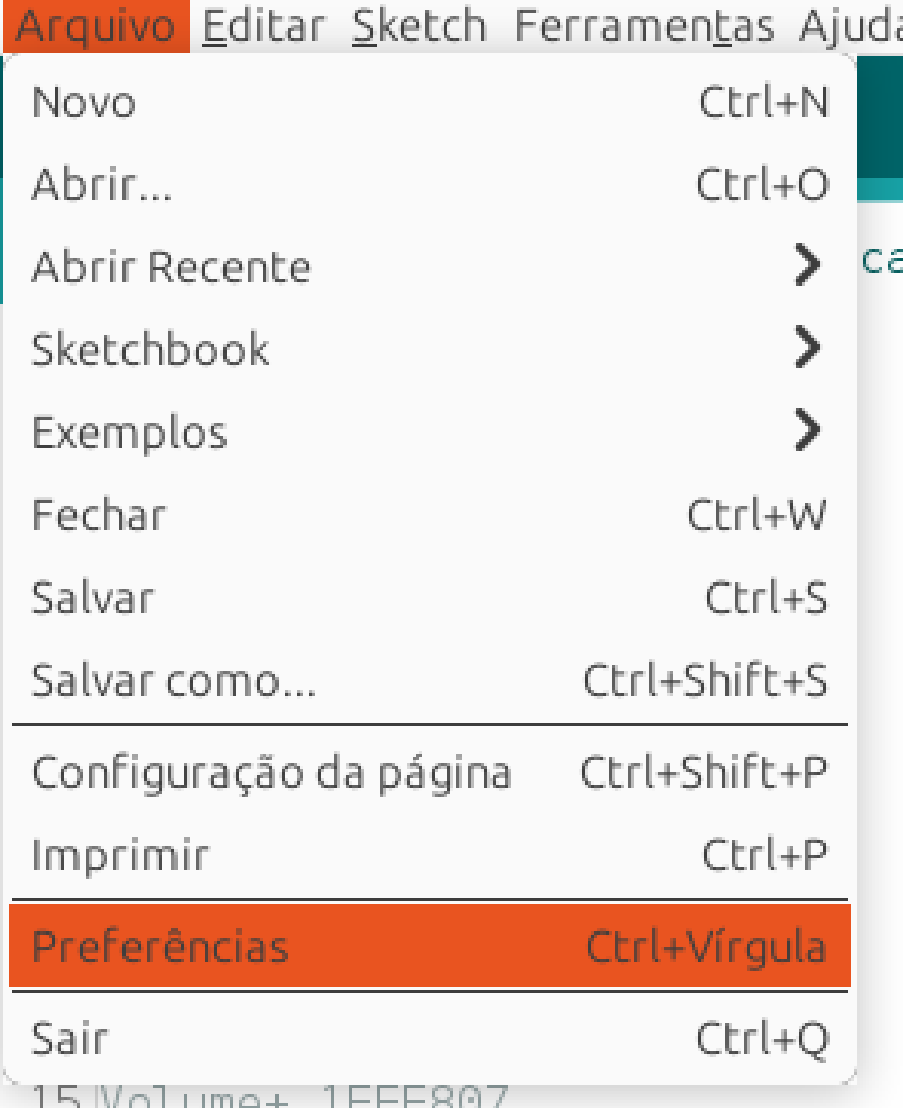


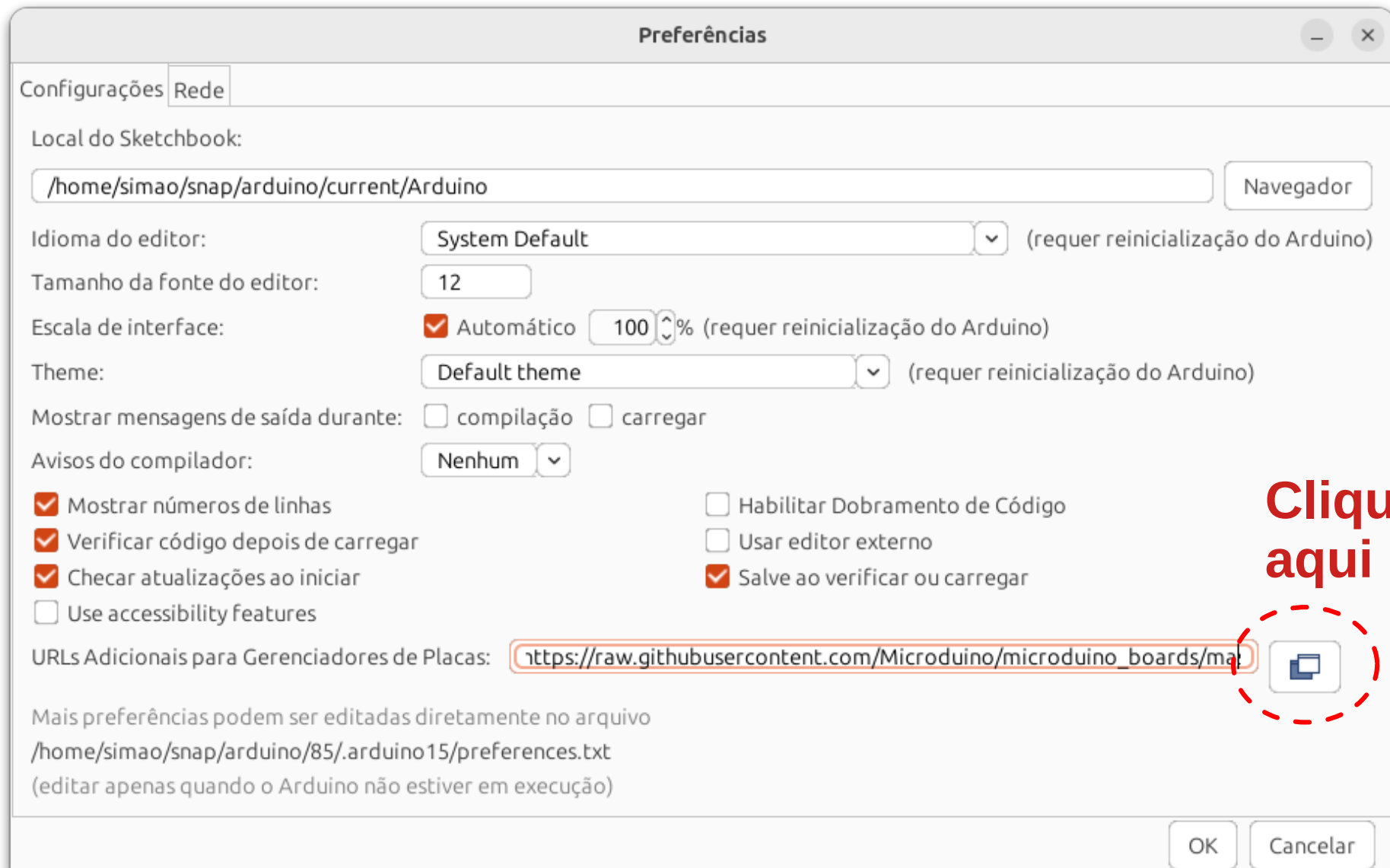
The screenshot shows a text editor window with the title bar 'placasExtrasArduino.txt'. The window contains a list of URLs for Microduino boards. The second URL is highlighted in orange. The text is as follows:

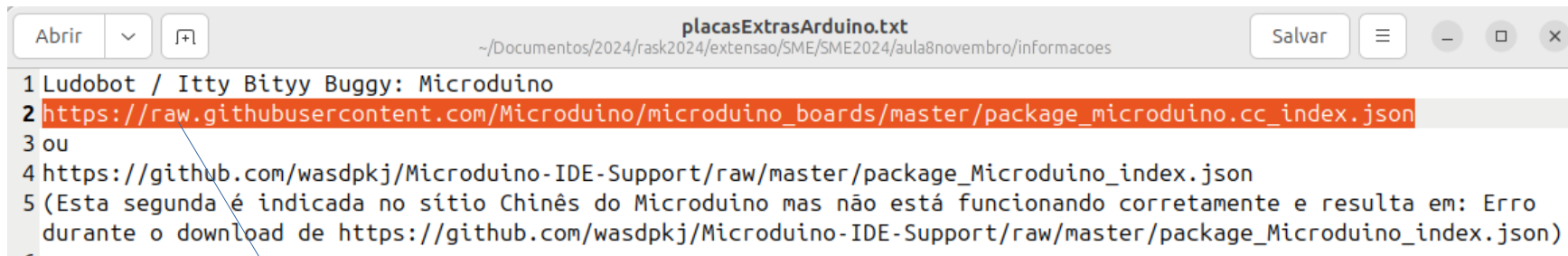
```
1 Ludobot / Itty Bityy Buggy: Microduino
2 https://raw.githubusercontent.com/Microduino/microduino_boards/master/package_microduino.cc_index.json
3 ou
4 https://github.com/wasdpkj/Microduino-IDE-Support/raw/master/package_Microduino_index.json
5 (Esta segunda é indicada no sítio Chinês do Microduino mas não está funcionando corretamente e resulta em: Erro
durante o download de https://github.com/wasdpkj/Microduino-IDE-Support/raw/master/package_Microduino_index.json)
```

Arquivo / Preferências

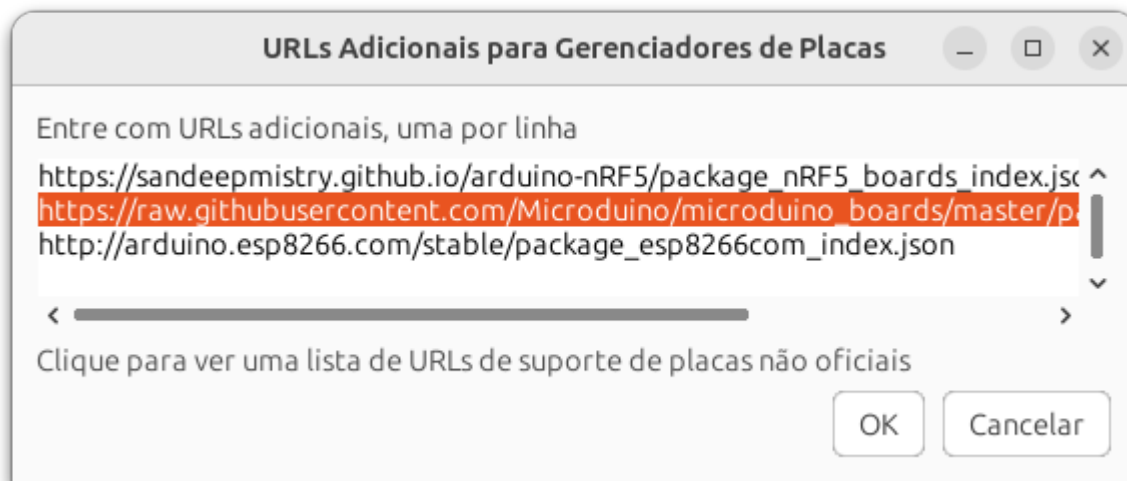
(a tela pode mudar em
função da versão)





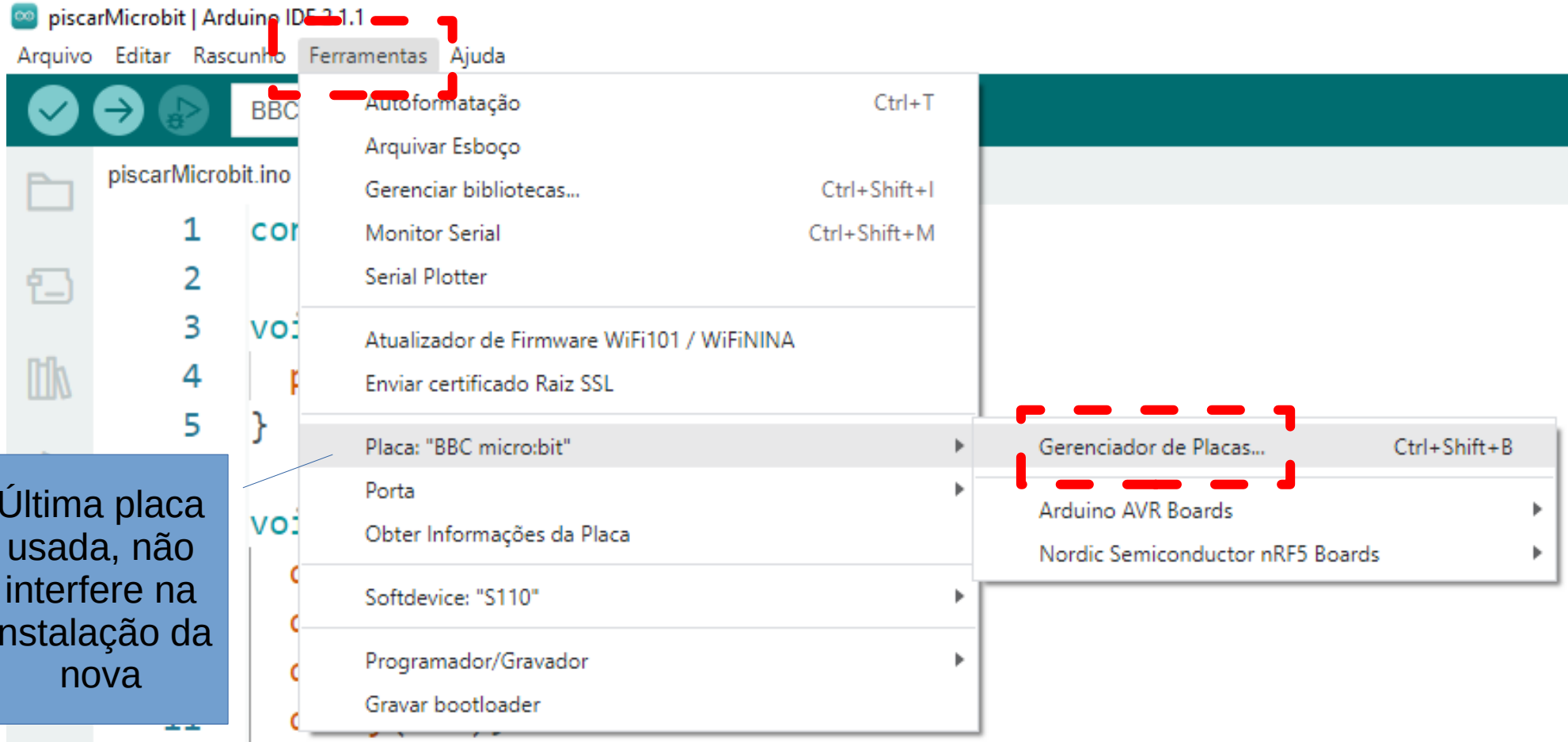


```
1 Ludobot / Itty Bitty Buggy: Microduino
2 https://raw.githubusercontent.com/Microduino/microduino_boards/master/package_microduino.cc_index.json
3 ou
4 https://github.com/wasdpkj/Microduino-IDE-Support/raw/master/package_Microduino_index.json
5 (Esta segunda é indicada no sítio Chinês do Microduino mas não está funcionando corretamente e resulta em: Erro
durante o download de https://github.com/wasdpkj/Microduino-IDE-Support/raw/master/package_Microduino_index.json)
```



Adicione o arquivo e clique em OK

Instalar os arquivos das placas





Última placa usada – não é referente à sua instalação atual

- Digite 'Microduino'
- ...aguarde a lista...
- Selecione *Microduino AVR Boards*
- Clique instalar

Saída



Downloading packages

Microduino:avr@1.0.1

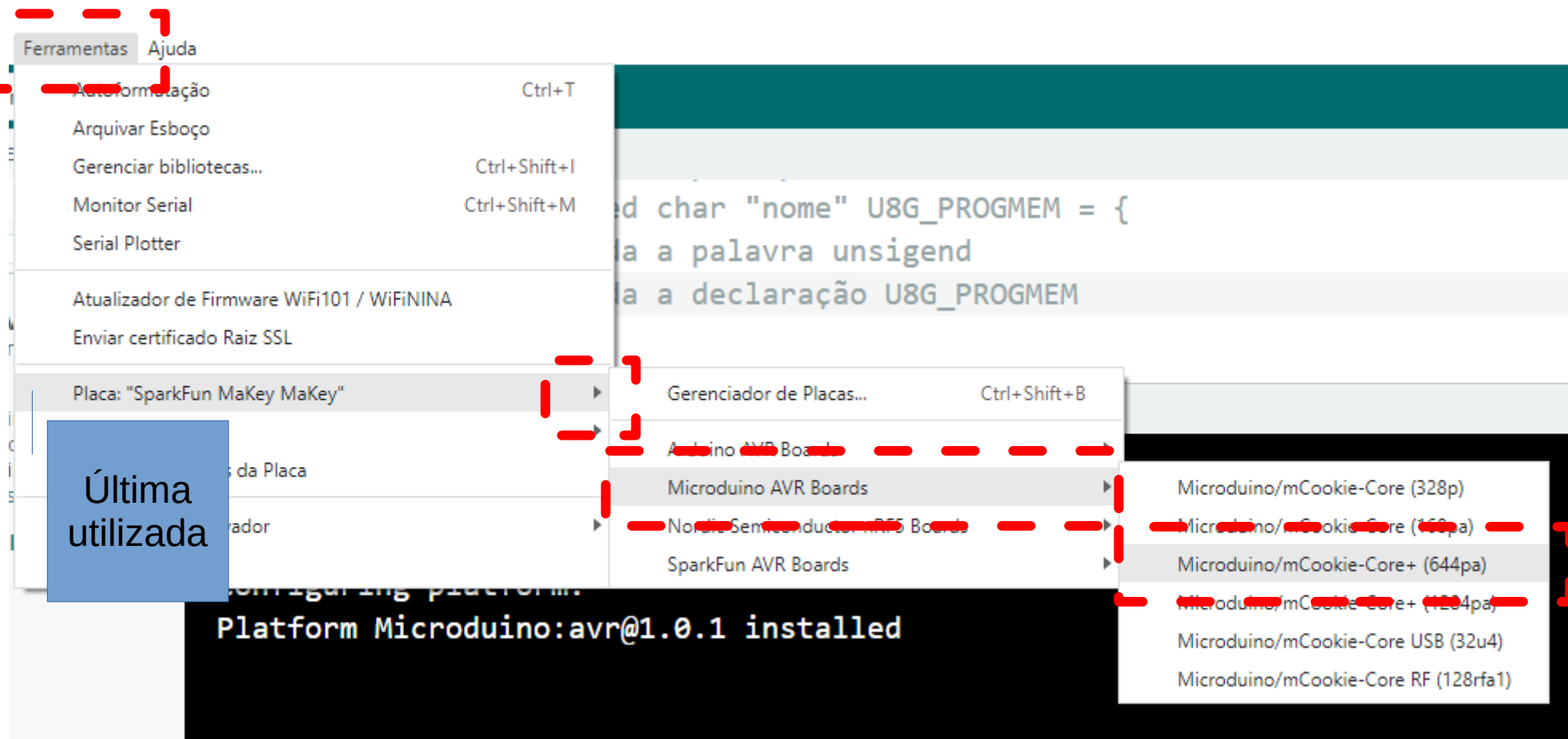
Installing platform Microduino:avr@1.0.1

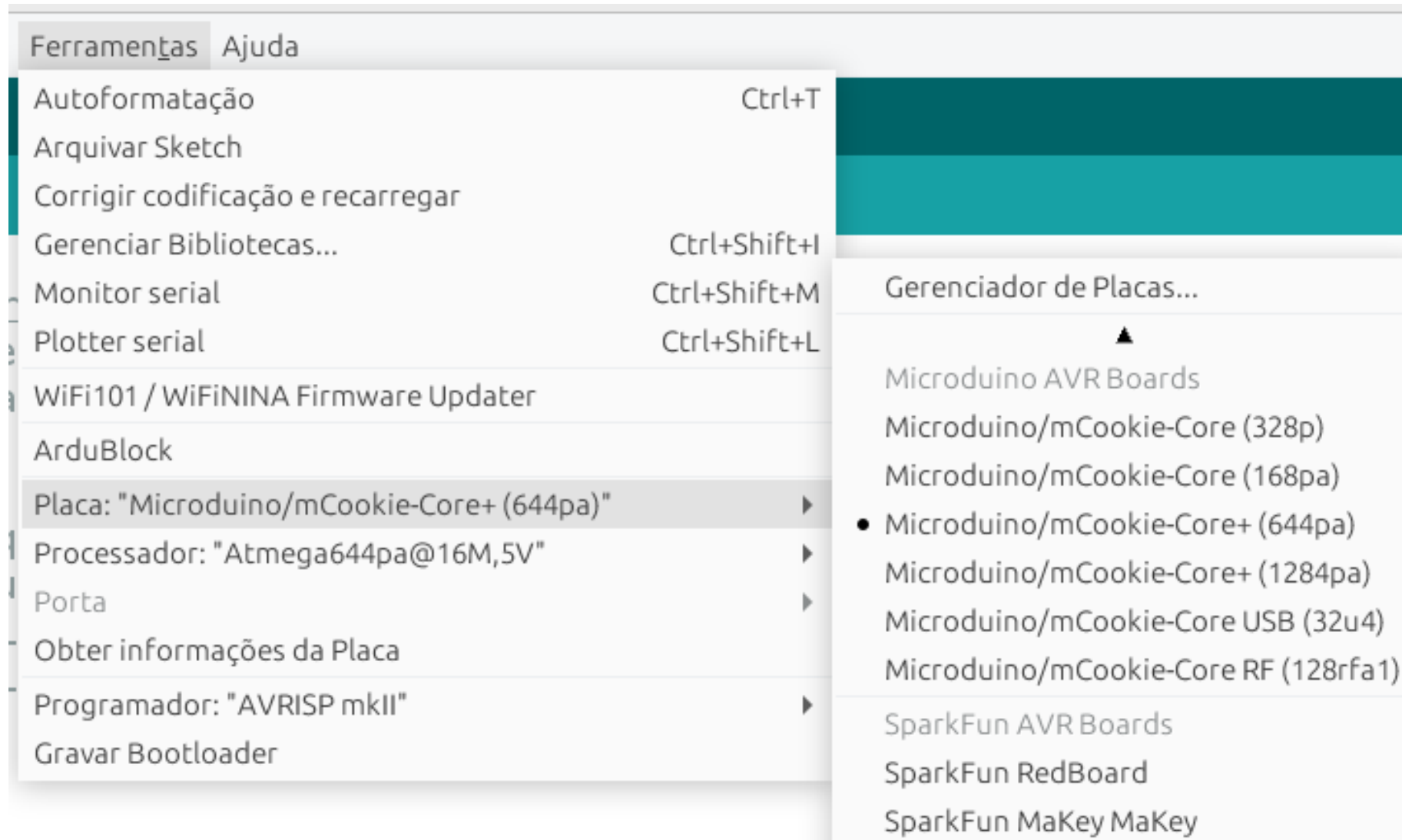
Configuring platform.

Platform Microduino:avr@1.0.1 installed

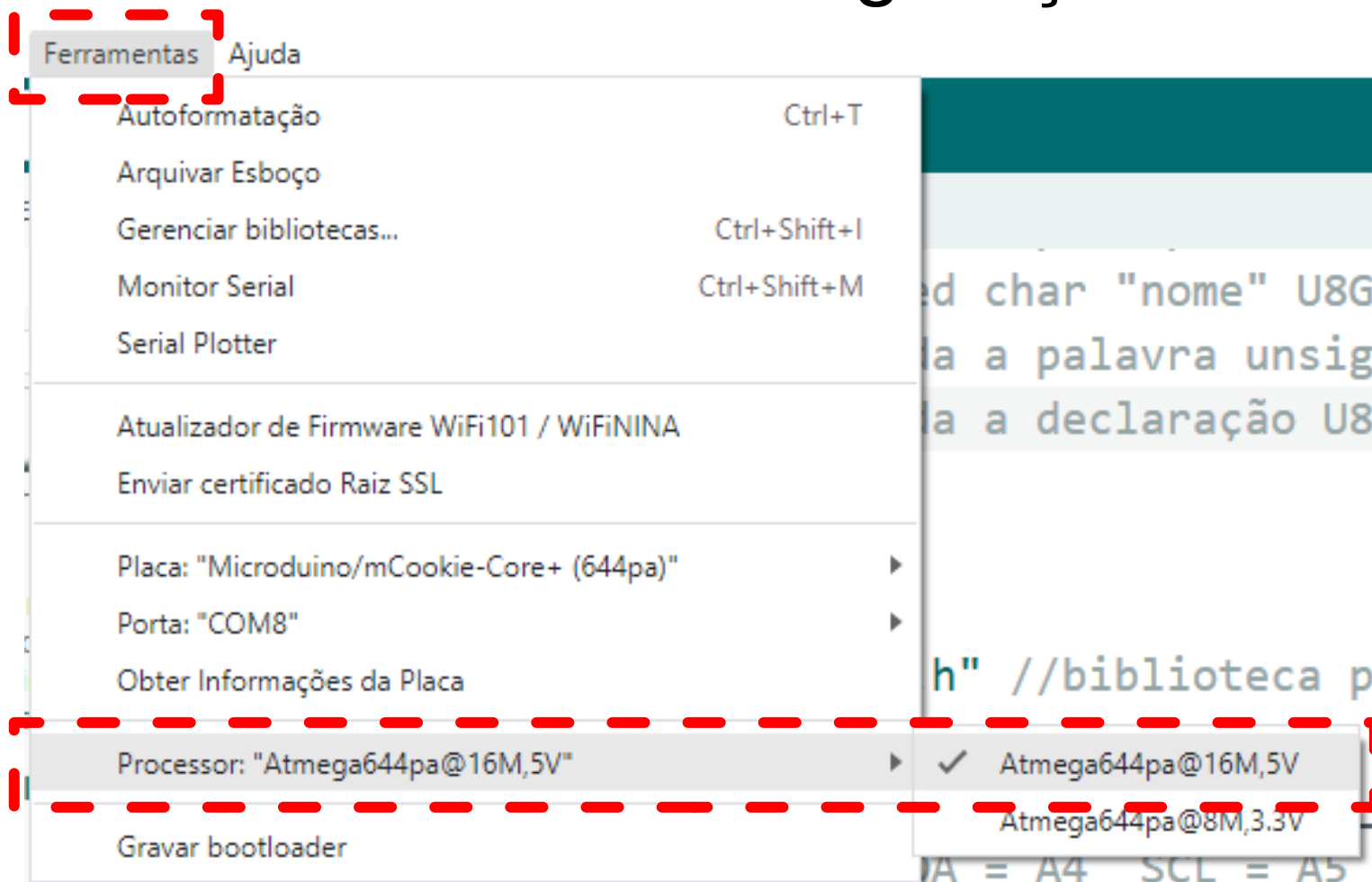
Em processamento Microduino AVR Boards:1.0.1: Platform Microduino:avr@1.0.1 installed

Selecione a placa

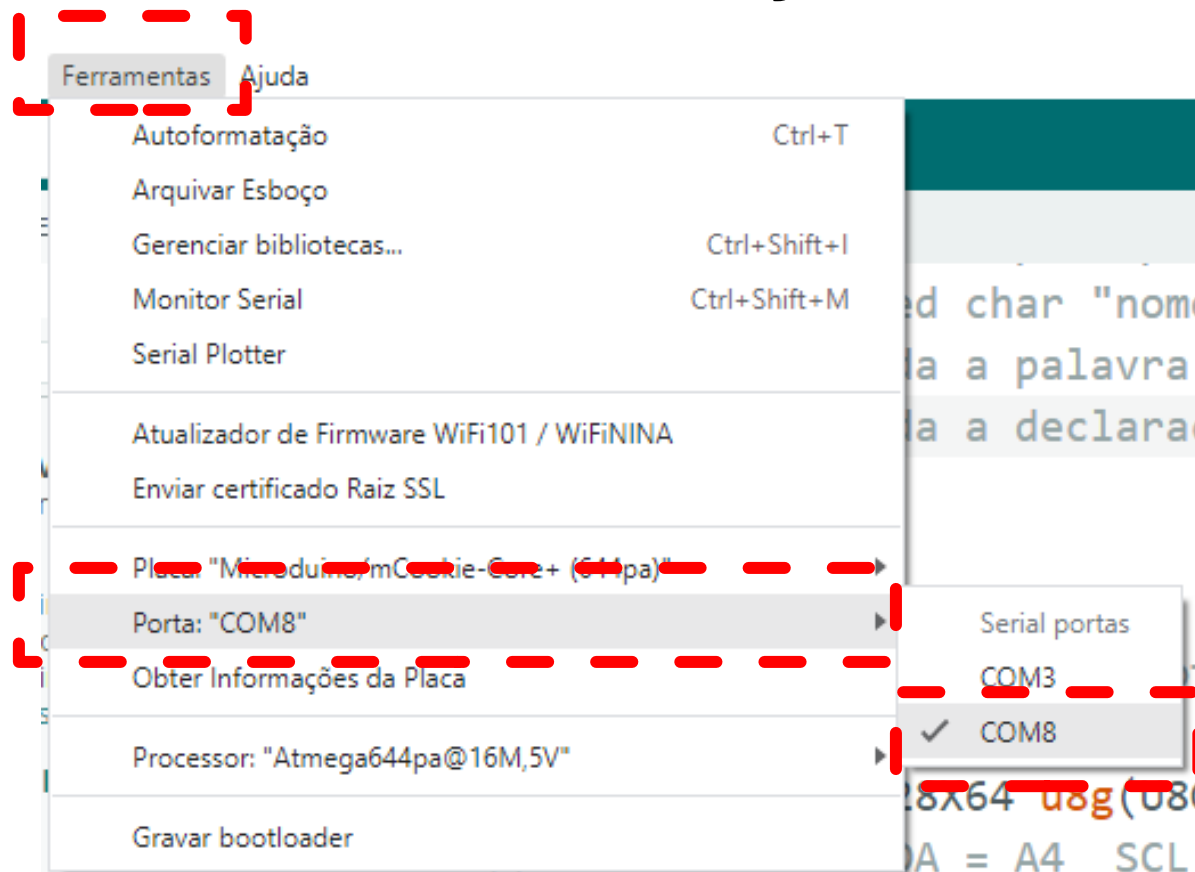




Acerte a configuração



Conecte o Ludobot e acerte a porta de comunicação



Não sabe qual porta usar ?

No Windows, pode usar o gerenciador de dispositivos



Portas (COM e LPT)

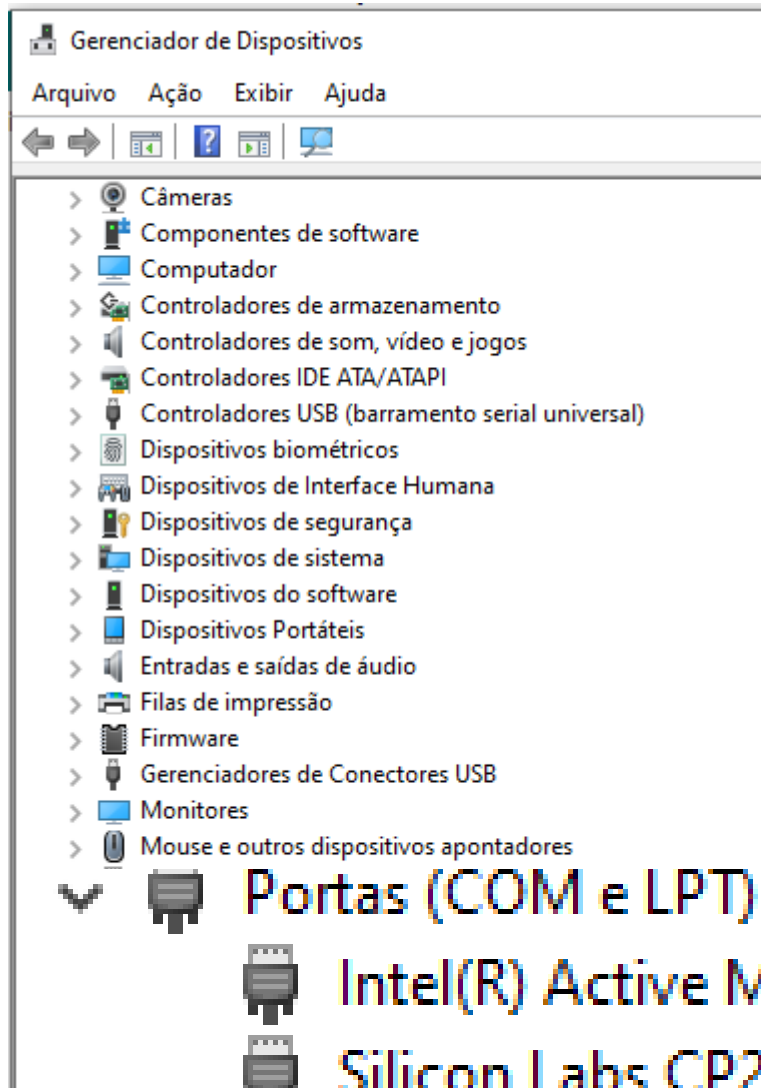


Intel(R) Active Management Technology - SOL (COM3)



Silicon Labs CP210x USB to UART Bridge (COM8)





Portas (COM e LPT)



Intel(R) Active Management Technology - SOL (COM3)



Silicon Labs CP210x USB to UART Bridge (COM8)



Imagens no *display*

Arquivo com o código



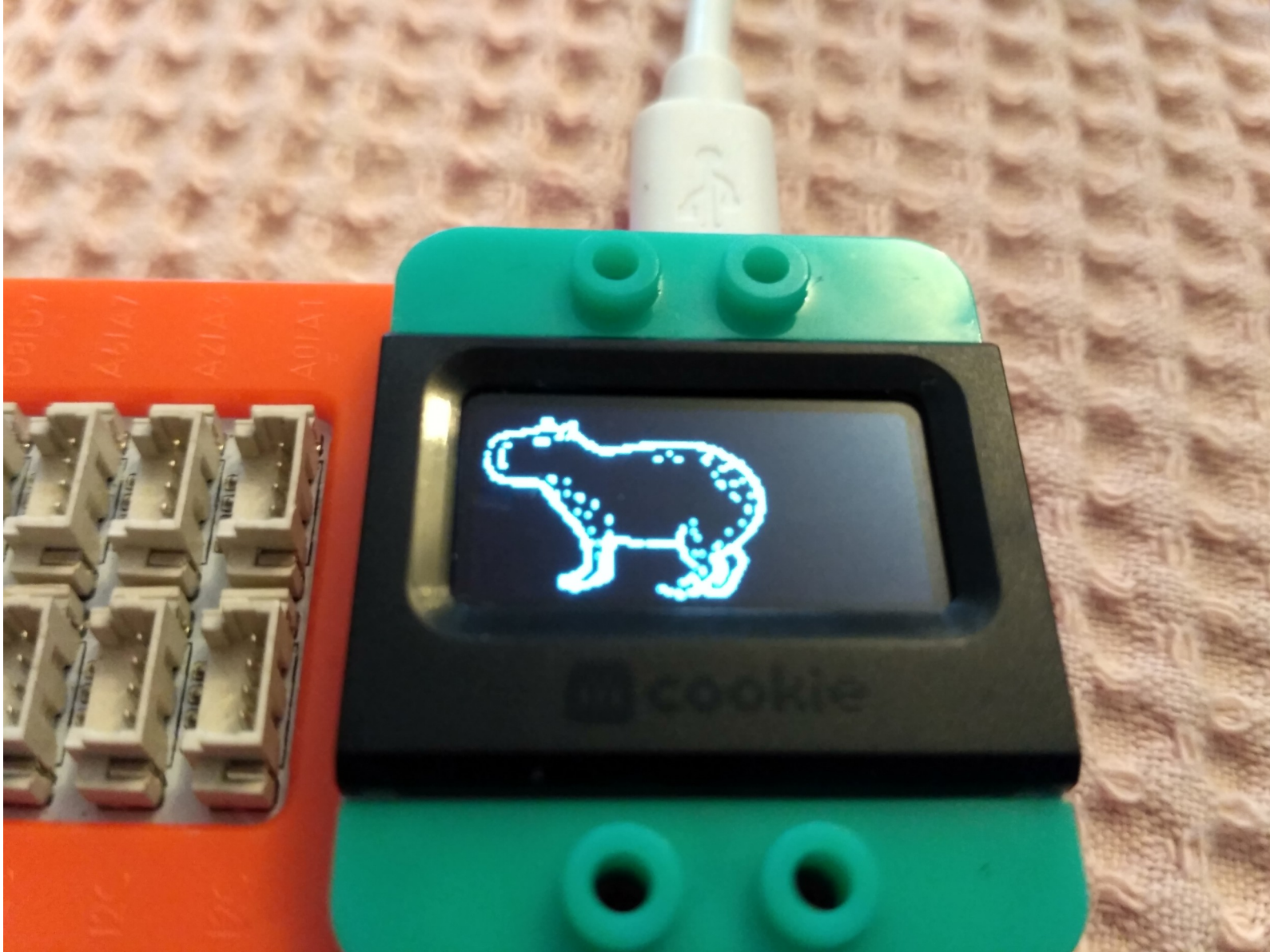


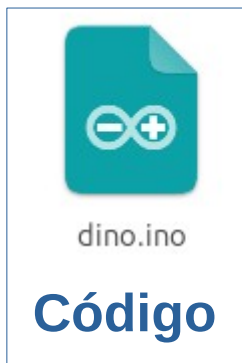
Arquivo com o código

```

1 #include "U8glib.h"
2
3 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);
4
5 #define capivara_width 89
6 #define capivara_height 64
7 static unsigned char capivara_bits[] U8G_PROGMEM = {
8     0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
9     0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
10    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
11    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
12    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
13    0x00, 0x00, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
14    0x00, 0x00, 0x38, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
15    0x00, 0x00, 0x68, 0x1E, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
16    0x00, 0x00, 0xFE, 0x13, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
17    0x00, 0xE0, 0x03, 0x19, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
18    0x00, 0x3F, 0x00, 0x39, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
19    0xC0, 0x01, 0x80, 0x75, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
20    0x20, 0x00, 0x1E, 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
21    0x10, 0x00, 0x1E, 0x80, 0x03, 0x00, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00,
22    0x10, 0x03, 0x00, 0x00, 0x07, 0xF8, 0x0F, 0x00, 0x3E, 0x00, 0x00, 0x00,
23    0x00, 0x00, 0x00, 0x00, 0xFC, 0x1F, 0x00, 0x00, 0xF0, 0x00, 0x00, 0x00,
24    0x80, 0x00, 0x00, 0x00, 0xF0, 0x00, 0x00, 0x01, 0x80, 0x01, 0x00, 0x00,
25    0x88, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x18, 0x07, 0x00, 0x00,
26    0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x30, 0x0C, 0x04, 0x0C, 0x00, 0x00,

```





dino.xbm

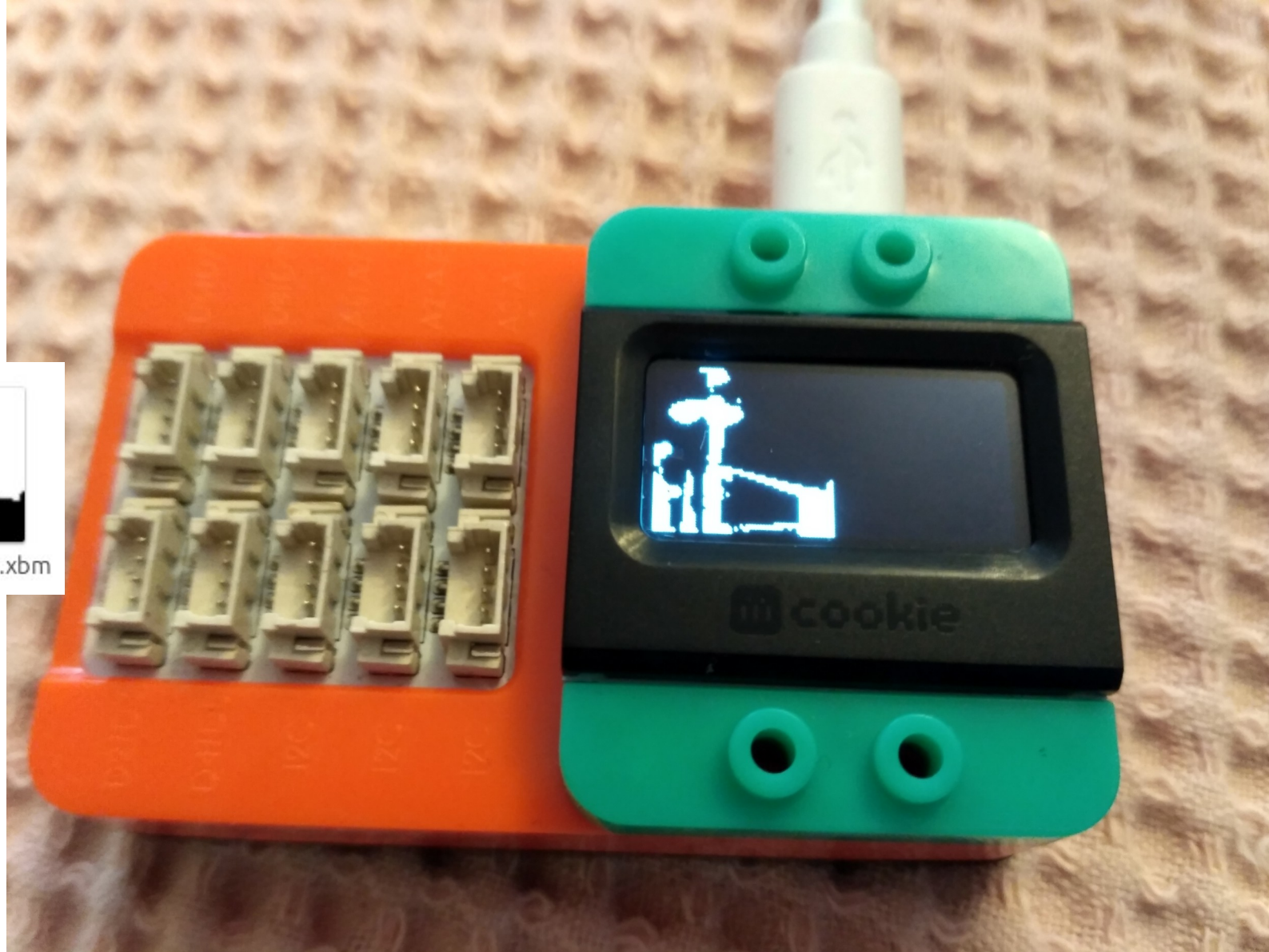




Farol.ino

farolChamecki.xbm

Código



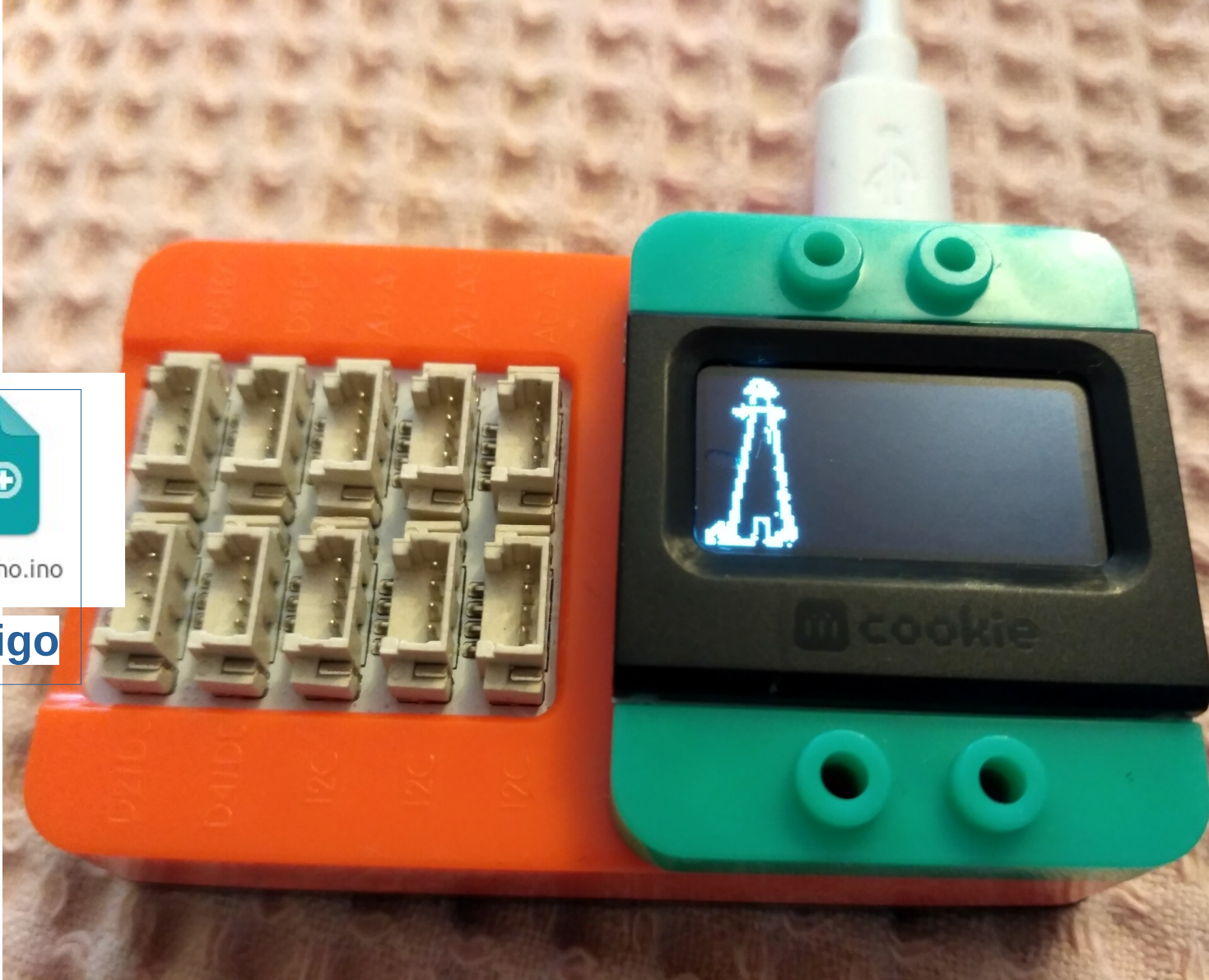


farolzinho64.xb
m



Farolzinho.ino

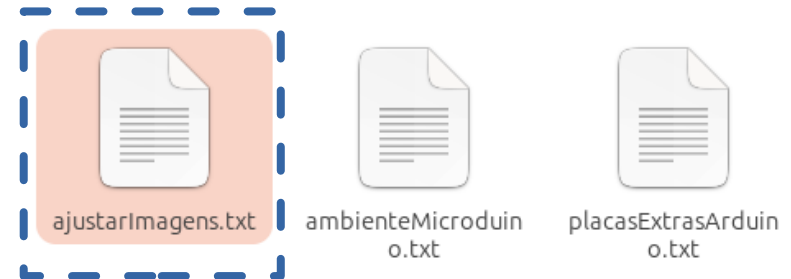
Código



Trabalhando as imagens

As imagens

- São arquivos '.XBM'. Para 'fazê-los', eu usei:
 - Para redimensionar a imagem para 128 x 64 (tamanho do *display*):
<https://www.img2go.com/pt/redimensionar-imagem>
 - Para converter a imagem para XBM, eu usei:
<https://convertio.co/pt/>



Exemplo



ufprAzulBranco.png



ufprAzulFundoBranco.png



ufprFundoCinza.png

Imagens de exemplo – antes de conversões

1 – Tamanho

Tamanho e tipo de arquivo originais



Tipo de imagem

png (PNG)

Largura

129 pixels

Altura

90 pixels

1 – Tamanho

Tamanho e tipo de arquivo originais



Tipo de imagem
png (PNG)

Largura
252 pixels

Altura
168 pixels

1 – Tamanho

Tamanho e tipo de arquivo originais



Tipo de imagem

png (PNG)

Largura

124 pixels

Altura

78 pixels

1 – Acertar tamanho

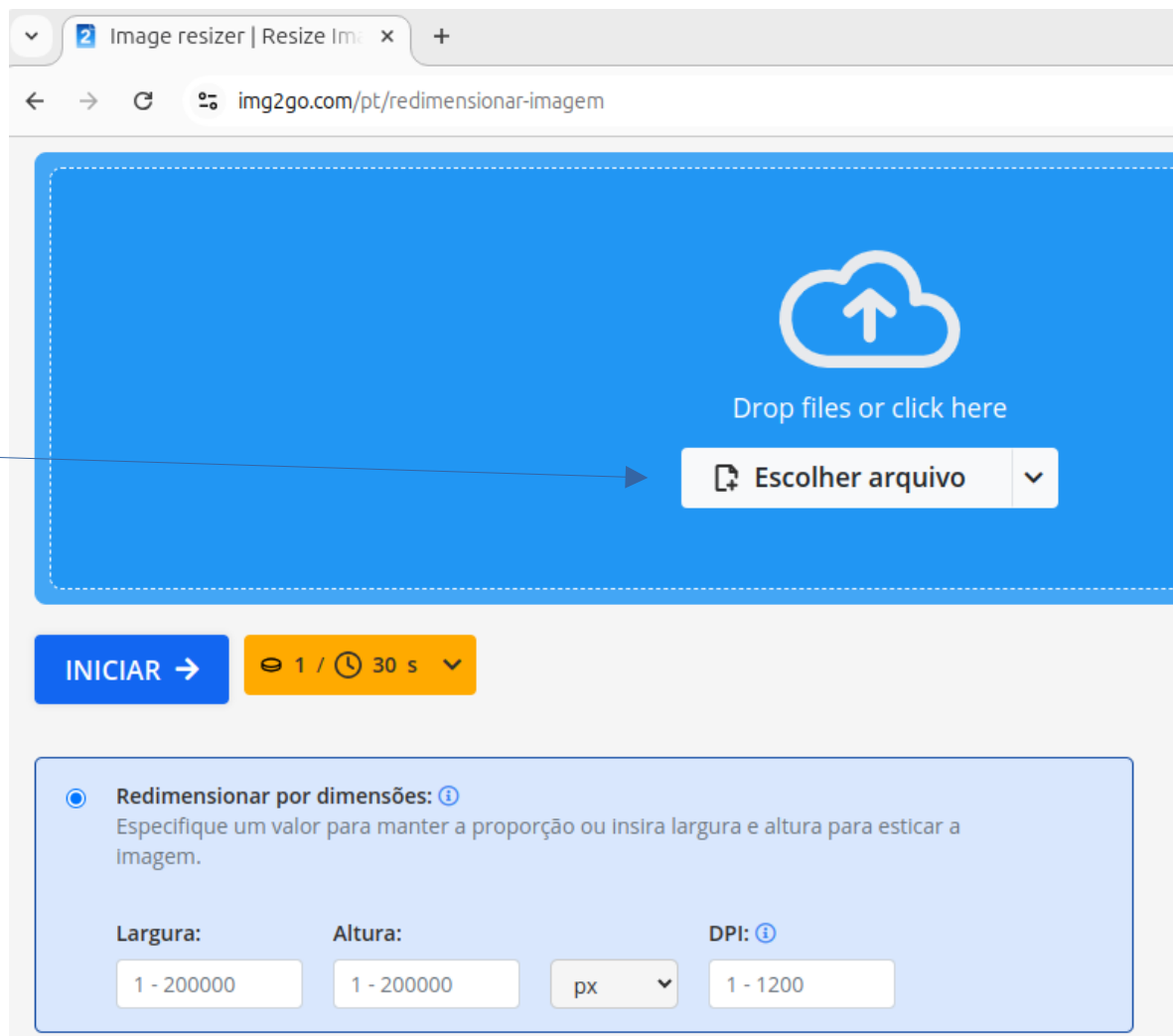


Imagem que será utilizada

Acessar:

<https://www.img2go.com/pt/redimensionar-imagem>

<https://www.img2go.com/pt/redimensionar-imagem>



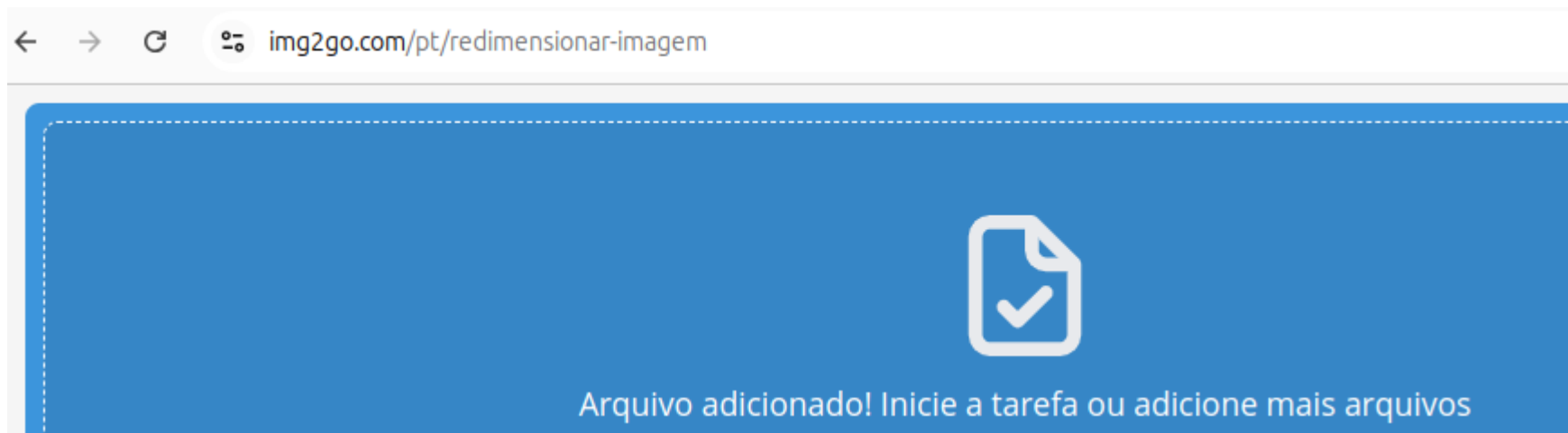
<https://www.img2go.com/pt/redimensionar-imagem>

The screenshot shows the 'Image resizer | Resize Imagem' page on the website [img2go.com/pt/redimensionar-imagem](https://www.img2go.com/pt/redimensionar-imagem). The main area is a large blue rectangle with a dashed border and a cloud icon with an upward arrow, containing the text 'Drop files or click here'. Below this is a button labeled 'Escolher arquivo' with a dropdown arrow. At the bottom left, there is a blue button 'INICIAR →' and an orange button '1 / 30 s'. Below these is a section titled 'Redimensionar por dimensões:' with instructions: 'Especifique um valor para manter a proporção ou insira largura e altura para esticar a imagem.' It includes input fields for 'Largura:' (1 - 200000), 'Altura:' (1 - 200000), a unit dropdown (px), and 'DPI:' (1 - 1200). A file selection dialog 'Abrir arquivos' is open, showing a sidebar with 'Imagens', 'Música', 'Vídeos', 'Pen...', 'Área de...', and 'Outros l...'. The main pane shows a list of files in the 'Área de Trabalho' folder:

Nome	Tamanho	Tipo	Modificado
ufprAzulFundoBranco.png	62,9 kB	Imagem	10:12
ufprFundoCinza.png	5,5 kB	Imagem	10:11
ufprAzulBranco.png	7,4 kB	Imagem	10:11

The dialog also has a search bar, an 'Abrir' button, and a checkbox 'Abrir arquivos em somente leitura'.

<https://www.img2go.com/pt/redimensionar-imagem>

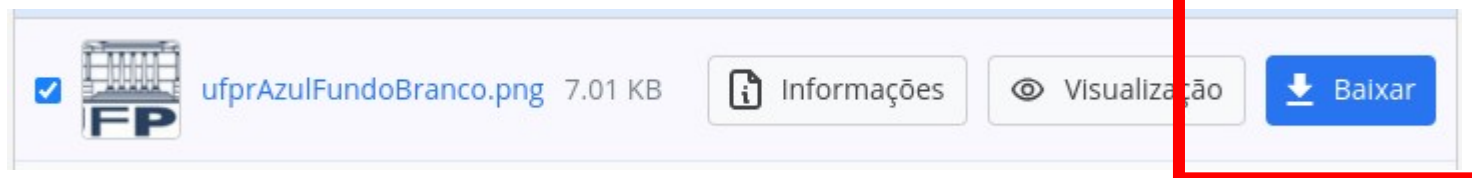
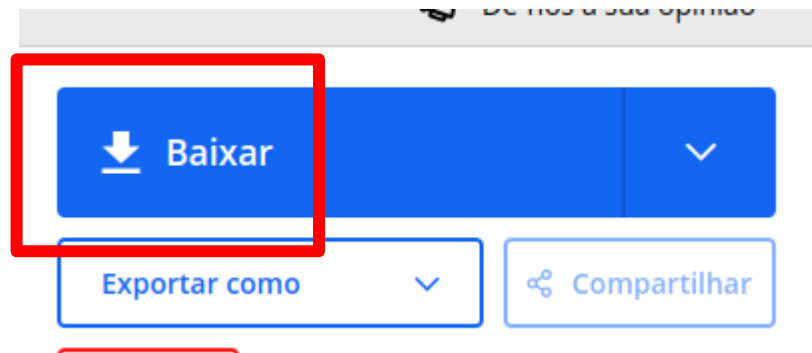
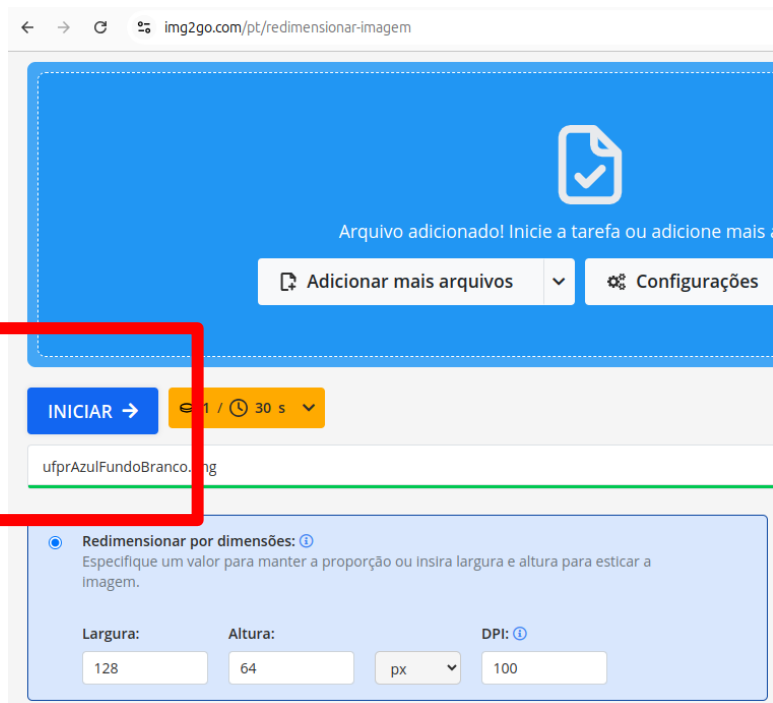


ufprAzulFundoBranco.png

☒ **Redimensionar por dimensões:** ⓘ
Especifique um valor para manter a proporção ou insira largura e altura para esticar a imagem.

Largura:	Altura:	DPI: ⓘ
<input type="text" value="128"/>	<input type="text" value="64"/>	<input type="text" value="100"/>
<input type="text" value="px"/> ▼		

<https://www.img2go.com/pt/redimensionar-imagem>



original



ajustada



ufprAzulFundoBran
co.png

ufprAzulFundoBran
co_128_64.png

Mudei o nome da segunda imagem
Dá para perceber uma queda na qualidade

2 – Converter para XBM



Imagem que será utilizada – já com o tamanho correto

ufprAzulFundoBran
co_128_64.png

Acessar:

<https://convertio.co/pt/>

https://convertio.co/pt/



Convertio — Conversor de arquivos

convertio.co/pt/

Convertio Converter ▼ OCR API Preços Help

Meus arquivos 1 Iniciar a sessão

Conversor de Ficheiros

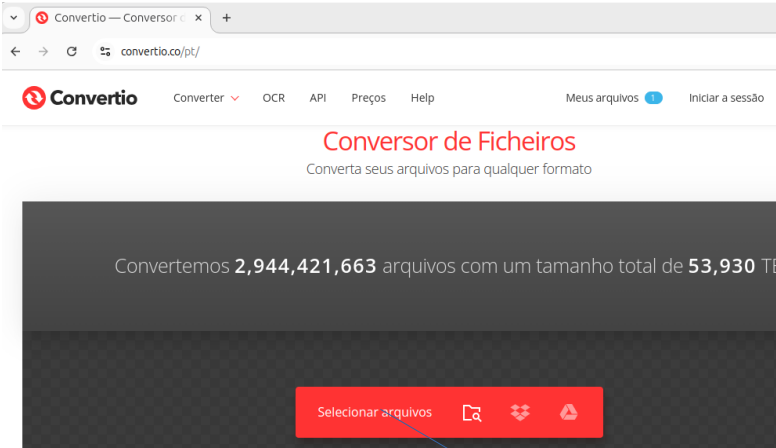
Converta seus arquivos para qualquer formato

Convertemos **2,944,421,663** arquivos com um tamanho total de **53,930 TB**

Selecionar arquivos



https://convertio.co/pt/





Converter ▾

OCR

API

Preços

Help

Meus arquivos 1

Iniciar a sessão

Registrar-se

Conversor de Ficheiros

Converte seus arquivos para qualquer formato



ufprAzulFundoBranco_128_64.png

para

... ▾

PRONTO

7.01 KB



Converte todos os em ▾

+ Adicionar mais ficheiros

Use Ctrl ou Shift para adicionar vários arquivos ao mesmo tempo



Converter



<https://convertio.co/pt/>

ufprAzulFundoBranco_128_64.png

para ... ▼

PRONTO

7.01 KB

Conv

Busca

+ Adicionar mais ficheiros

Use Ctrl ou Shift para adicionar vários arquivos ao mesmo tempo

Imagem

Documento

EBook

Fonte

Vetor

CAD

SVG

ICO

JPG

WEBP

JPEG

HDR

BMP

DDS

GIF

CUR

PSD

TIFF

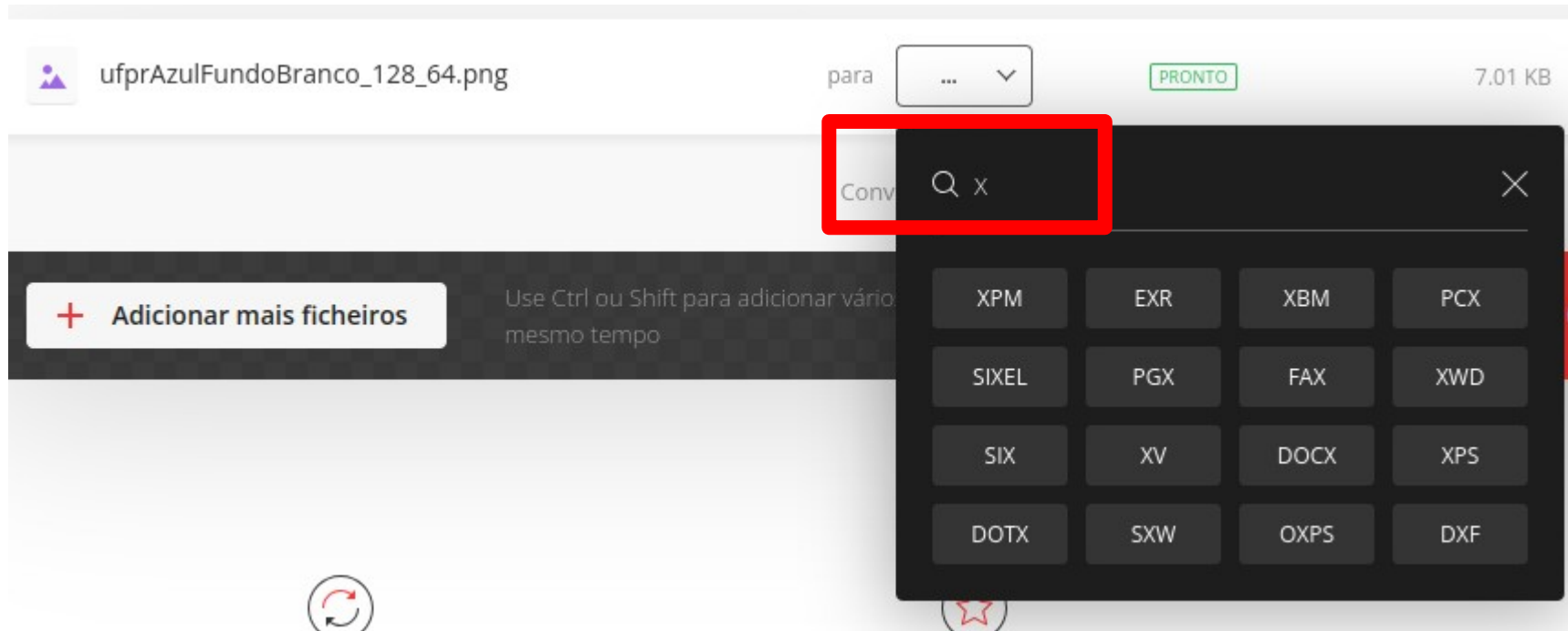
TGA

AVIF

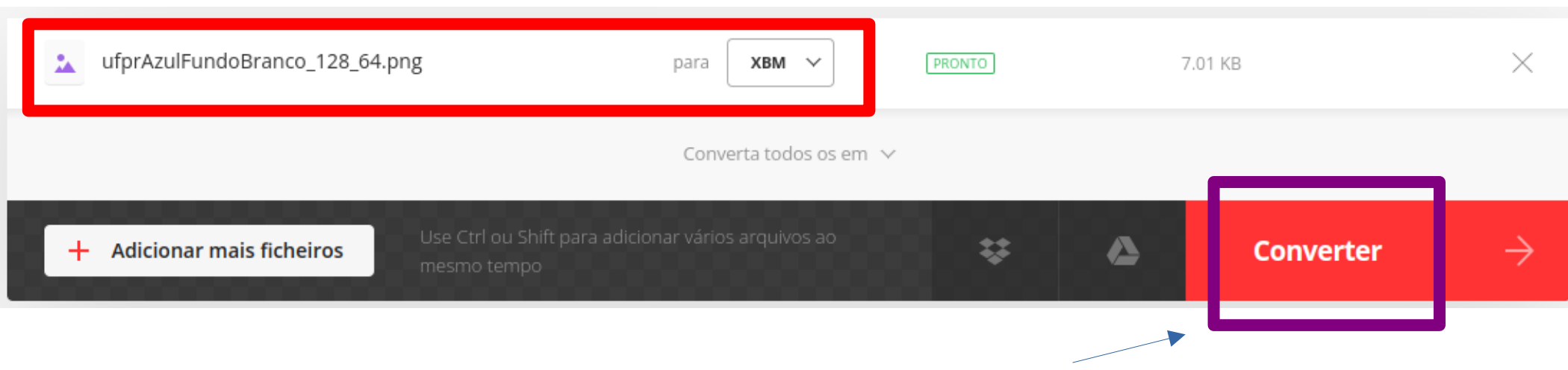
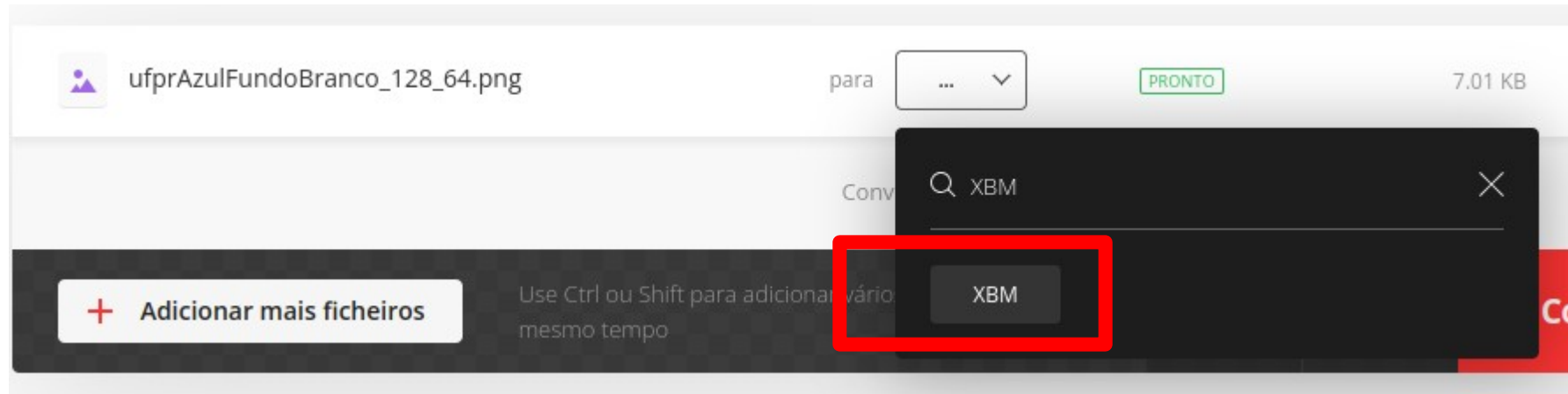
RGB

Mais de 300 formatos

<https://convertio.co/pt/>



https://convertio.co/pt/



https://convertio.co/pt/

Convertio Converter ▼ OCR API Preços Help Meus arquivos Iniciar a sessão Registrar-se

Convertendo seu arquivo
Aguarde um momento. O botão de download deve aparecer em breve...

ufprAzulFundoBranco_128_64.png **EM CONVERSÃO** Cancelar

Conversão concluída!

Baixe seu arquivo convertido

ufprAzulFundoBranco_128_64.xbm

CONCLUÍDO

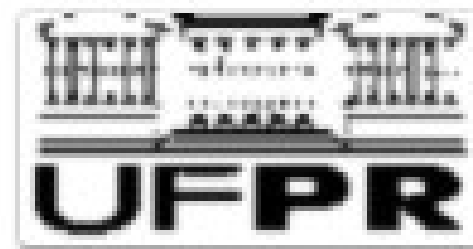
XBM / 6.46 KB

Descarregar

Resultado



ufprAzulFundoBran
co_128_64.png



ufprAzulFundoBran
co_128_64.xbm



Usar: abra o arquivo em
um editor de textos

ufprAzulFundoBran
co_128_64.xbm

A screenshot of a text editor window. The title bar shows the filename 'ufprAzulFundoBranco_128_64.xbm' and the path '~/Área de Trabalho/exemplomagem'. The window has buttons for 'Abrir', a dropdown menu, a search icon, 'Salvar', and window control buttons. The text content is as follows:

```
1 #define 3ce87b03543e493edb77a23bf8aa53fblCk8GF0StmIxsgjv_width 128
2 #define 3ce87b03543e493edb77a23bf8aa53fblCk8GF0StmIxsgjv_height 64
3 static char 3ce87b03543e493edb77a23bf8aa53fblCk8GF0StmIxsgjv_bits[] = {
4     0x00, 0x70, 0x00, 0xE0, 0x00, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x80,
5     0x07, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF,
6     0xFF, 0xFF, 0x1F, 0x40, 0x00, 0x00, 0x10, 0x00, 0x00, 0x01, 0x00, 0x00,
7     0x08, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x80, 0x00,
8     0xE0, 0x01, 0x00, 0x00, 0xF8, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0x0F, 0x0F,
9     0x00, 0x00, 0x80, 0x07, 0xE0, 0x00, 0x00, 0x00, 0xE0, 0x33, 0x00, 0x00,
10    0x00, 0x00, 0xCC, 0x07, 0x00, 0x00, 0x00, 0x07, 0x40, 0x00, 0x00, 0x00,
11    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x04, 0x00, 0x00, 0x00, 0x02,
12    0xE0, 0x00, 0x00, 0x00, 0x60, 0x30, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x06,
```

ufprAzulFundoBran
co_128_64.xbm

Usar: abra o arquivo em um editor de textos

```
ufprAzulFundoBranco_128_64.xbm
~/Área de Trabalho/exemploimagem

1 #define 3ce87b03543e493edb77a23bf8aa53fb1Ck8GF0StmIxsgjv_width 128
2 #define 3ce87b03543e493edb77a23bf8aa53fb1Ck8GF0StmIxsgjv_height 64
3 static char 3ce87b03543e493edb77a23bf8aa53fb1Ck8GF0StmIxsgjv_bits[] = {
```

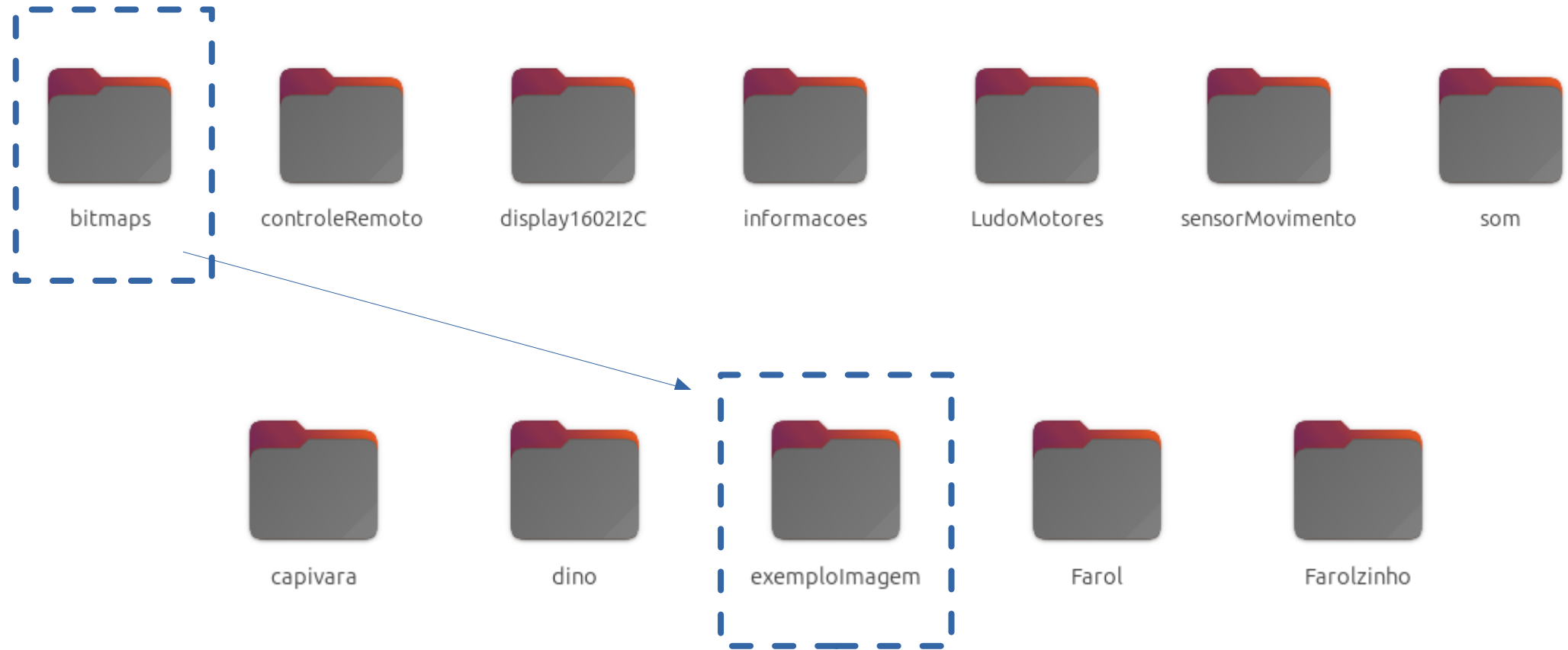
Este 'nome', 3ce87b03543e493edb77a23bf8aa53fb1Ck8GF0StmIxsgjv, foi criado pelo conversor para identificar o arquivo. Mas, podemos mudá-lo para facilitar para nós (não é necessário; é possível mudar):

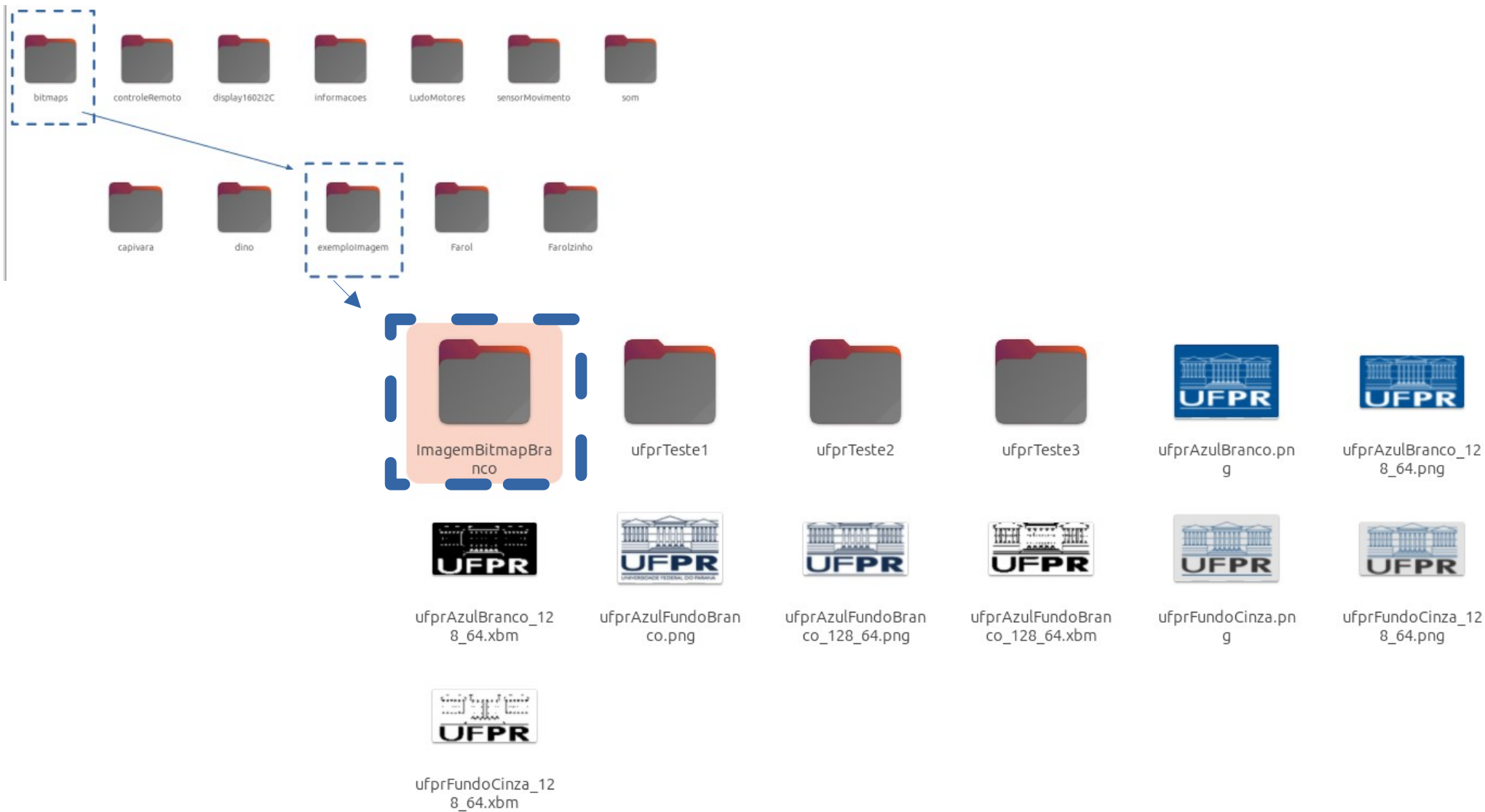
```
*ufprAzulFundoBranco_128_64.xbm
~/Área de Trabalho/exemploimagem

1 #define ufprTeste1_width 128
2 #define ufprTeste1_height 64
3 static char ufprTeste1_bits[] = {
```

Note que é precisamos trocar o nome em: *width*, *height* e *bits*

Para usar o arquivo





Abra o exemplo 'em branco'

ImagemBitmapBranco

```
1 /*
2 Troque os nomes das variáveis
3   de:  NOME_Para_Trocar
4   para: o nome que vc quiser
5   São _6_ posições em que aparece este nome de variável
6   Se você só colar o conteúdo (bits) sem trocar os nomes vai funcionar
7 Salve o projeto com outro nome
8 */
9
10
11 #include "U8glib.h"
12
13 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);
14
15 #define NOME_Para_Trocar_width 128           //troque o nome
16 #define NOME_Para_Trocar_height 64          //troque o nome
17 static unsigned char NOME_Para_Trocar_bits[] U8G_PROGMEM = { //troque o nome
18   //cole aqui o conteúdo da imagem a ser exibida
19 };
20 };
21
22 void setup() {
23   u8g.setColorIndex(1);
24 }
25
26
27 void loop() {
28   u8g.firstPage();
29   do {
30     u8g.drawXBMP( 0, 0, NOME_Para_Trocar_width, NOME_Para_Trocar_height, NOME_Para_Trocar_bits); //troque os nomes
31   }
32   while(u8g.nextPage());
33 }
```



ImagemBitmapBranco.ino

Código


```

1 #define uftrocar_width 128
2 #define uftrocar_height 64
3 static char uftrocar_bits[] = {
4     0x00, 0x70, 0x00, 0x20, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x00,
5     0x07, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x01, 0x00, 0x00,
6     0x0F, 0x0F, 0x1F, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x01, 0x00, 0x00,
7     0x00, 0x03, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00,
8     0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F,
9     0x00, 0x00, 0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
10    0x00, 0x00, 0x0C, 0x07, 0x00, 0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00,
11    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x04, 0x00, 0x00, 0x00, 0x02,
12    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0C, 0x00,
13    0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x0C,
14    0x07, 0x0F, 0x01, 0x0C, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x03, 0x0C,
15    0x1C, 0x00, 0x07, 0x02, 0x0C, 0x01, 0x00, 0x00, 0x00, 0x0C, 0x07, 0x00, 0x07,
16    0x00, 0x03, 0x10, 0x03, 0x14, 0x00, 0x07, 0x03, 0x0C, 0x00, 0x00, 0x00,
17    0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x02,
18    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
19    0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
20    0x00, 0x01, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
21    0x04, 0x00, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
22    0x00, 0x00, 0x00, 0x00, 0x04, 0x00, 0x04, 0x00, 0x00, 0x00, 0x01, 0x10, 0x00,
23    0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x00,
24    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
25    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
26    0x22, 0x01, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x10, 0x00,
27    0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00,
28    0x00, 0x01, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
29    0x04, 0x00, 0x04, 0x00, 0x00, 0x00, 0x01, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00,
30    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x01, 0x10, 0x00,
31    0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
32    0x00, 0x01, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
33    0x04, 0x00, 0x04, 0x00, 0x00, 0x00, 0x03, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00,
34    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
35    0x1C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
36    0x00, 0x0F, 0x0E, 0x0F, 0x10, 0x00, 0x00, 0x04, 0x22, 0x13, 0x01, 0x0C,
37    0x07, 0x00, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x04,
38    0x00, 0x13, 0x01, 0x0C, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
39    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
40    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
41    0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x07, 0x0C,
42    0x03, 0x01, 0x00, 0x0C, 0x0F, 0x0F, 0x0F, 0x07, 0x00, 0x00, 0x00, 0x00, 0x00,
43    0x00, 0x00, 0x0F, 0x0C, 0x03, 0x01, 0x00, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,
44    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x01, 0x00, 0x00,
45    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
46    0x34, 0x1A, 0x01, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
47    0x20, 0x0C, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00,
48    0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x00,
49    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
50    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
51    0x0F, 0x0D, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F,
52    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
53    0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F,
54    0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x07, 0x00, 0x0F, 0x0F, 0x0F, 0x0F,
55    0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F, 0x0F,
56    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
57    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
58    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
59    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
60    0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
61    0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F,
62    0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
63    0x01, 0x07, 0x0F, 0x00, 0x0F, 0x0F, 0x0F, 0x03, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00,
64    0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x03,
65    0x0F, 0x0F, 0x0F, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
66    0x0F, 0x03, 0x0F, 0x07, 0x0F, 0x01, 0x0F, 0x03, 0x00, 0x03, 0x03, 0x00, 0x00, 0x00,
67    0x01, 0x07, 0x00, 0x00, 0x0F, 0x03, 0x0C, 0x07, 0x0F, 0x01, 0x0F, 0x03, 0x00,
68    0x00, 0x03, 0x00, 0x00, 0x00, 0x01, 0x07, 0x00, 0x00, 0x0F, 0x03, 0x0C, 0x07, 0x0F,
69    0x0F, 0x01, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x0F, 0x0F, 0x0F,
70    0x0F, 0x03, 0x0F, 0x07, 0x0F, 0x01, 0x0F, 0x01, 0x00, 0x03, 0x00, 0x00, 0x00, 0x00,
71    0x01, 0x0F, 0x0F, 0x1F, 0x0F, 0x0F, 0x0F, 0x07, 0x0F, 0x0F, 0x0F, 0x0F, 0x01,
72    0x00, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03,
73    0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
74    0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
75    0x01, 0x07, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x0F, 0x07, 0x00, 0x00,
76    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
77    0x0F, 0x0D, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
78    0x0F, 0x03, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0F,
79    0x00, 0x07, 0x00, 0x00, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
80    0x0C, 0x0F, 0x00, 0x0F, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
81    0x0F, 0x0C, 0x0F, 0x00, 0x00, 0x0F, 0x0F, 0x0F, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
82    0x0F, 0x03, 0x00, 0x00, 0x0F, 0x01, 0x0F, 0x01, 0x00, 0x0F, 0x0F, 0x0F, 0x0F,
83    0x00, 0x07, 0x00, 0x00, 0x0F, 0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
84    0x00, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
85    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
86    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
87    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
88    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
89    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

```

ImagemBitmapBranco

```

1 /*
2 Troque os nomes das variáveis
3 de: NOME_Para_Trocar
4 para: o nome que vc quiser
5 São 6 posições em que aparece este nome de variável
6 Se você só colar o conteúdo (bits) sem trocar os nomes vai funcionar
7 Salve o projeto com outro nome
8 */
9
10
11 #include "U8glib.h"
12
13 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);
14
15 #define NOME_Para_Trocar_width 128 //troque o nome
16 #define NOME_Para_Trocar_height 64 //troque o nome
17 static unsigned char NOME_Para_Trocar_bits[] U8G_PROGMEM = { //troque o nome
18     //cole aqui o conteúdo da imagem a ser exibida
19
20 };
21
22 void setup() {
23     u8g.setColorIndex(1);
24 }
25
26
27 void loop() {
28     u8g.firstPage();
29     do {
30         u8g.drawXBMP( 0, 0, NOME_Para_Trocar_width, NOME_Para_Trocar_height, NOME_Para_Trocar_bits); //troque os nomes
31     }
32     while(u8g.nextPage());
33 }

```

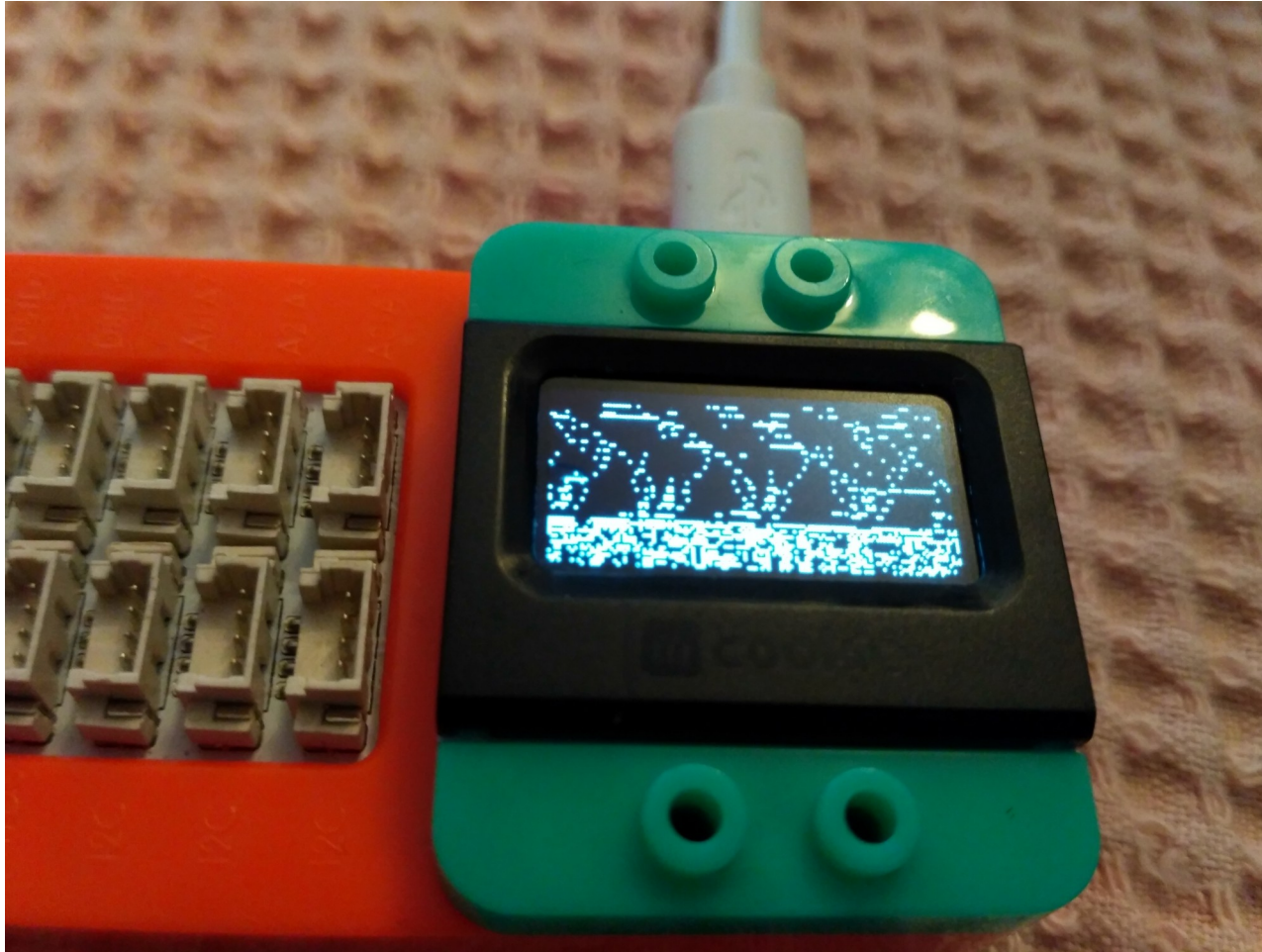
Copie o conteúdo da imagem para o arquivo em branco (ainda sem ajustar nada)

Verifique se o tamanho está correto

```
13 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);  
14  
15 #define NOME_Para_Trocar_width 128 //troque o nome  
16 #define NOME_Para_Trocar_height 64 //troque o nome  
17 static unsigned char NOME_Para_Trocar_bits[] U8G_PROGMEM = { //troque o nome  
18 //cole aqui o conteúdo da imagem a ser exibida  
19 0x00, 0x70, 0x00, 0xE0, 0x00, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x80,  
20 0x07, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF,
```

```
1 #define ufprTeste1_width 128  
2 #define ufprTeste1_height 64  
3 static char ufprTeste1_bits[] = {  
4 0x00, 0x70, 0x00, 0xE0, 0x00, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x80,  
5 0x07, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF,  
6 0xFF, 0xFF, 0x1F, 0x40, 0x00, 0x00, 0x10, 0x00, 0x00, 0x01, 0x00, 0x00,  
7 0x08, 0xE0, 0xFF, 0xFF, 0xFF, 0xFF, 0x07, 0x00, 0x00, 0x00, 0x80, 0x00,
```

Verifique se o tamanho está correto – senão o *display* 'se perde'



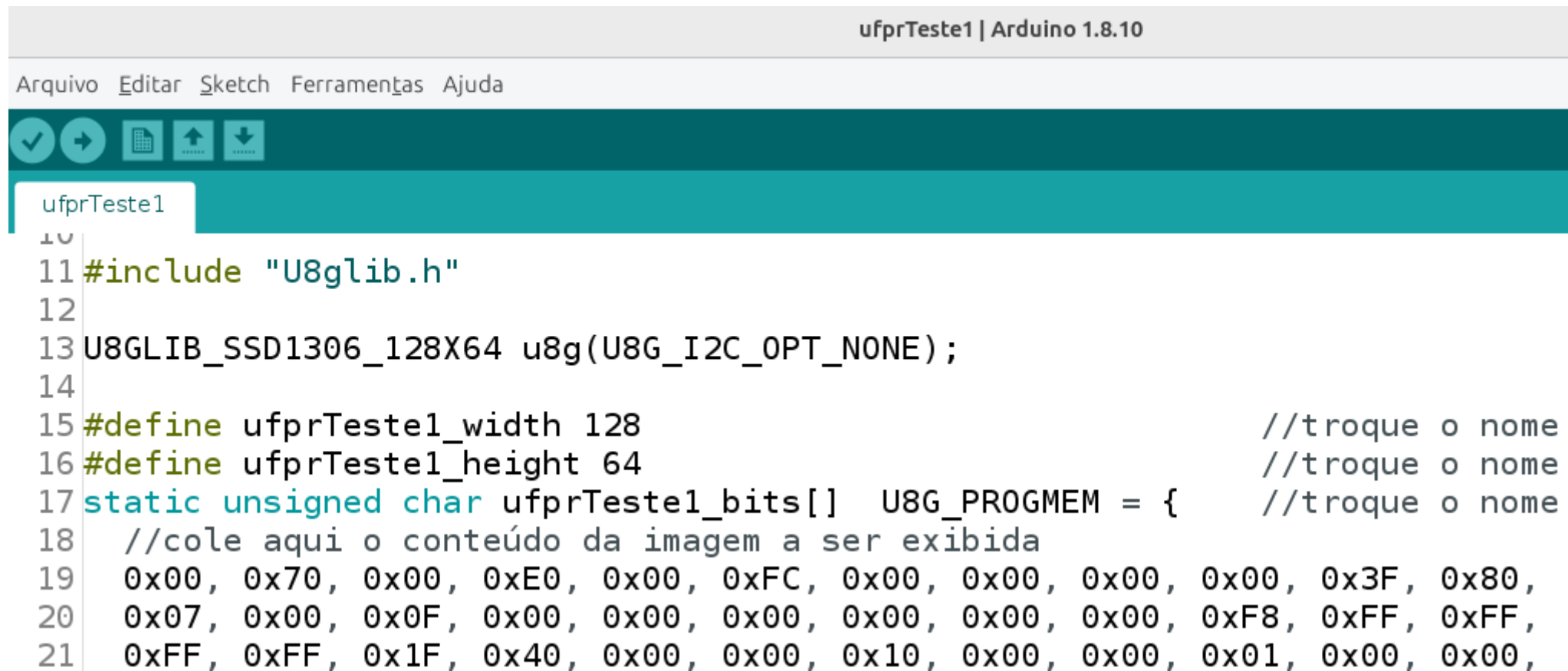
Se quiser, ajuste os nomes das variáveis

```
15 #define ufprTestel_width 128
16 #define ufprTestel_height 64
17 static unsigned char ufprTestel_bits[] U8G_PROGMEM = {
```

```
114 do {
115     u8g.drawXBMP( 0, 0, ufprTestel_width, ufprTestel_height, ufprTestel_bits);
116 }
```

** o 'nome' aparece em _6_ vezes no código*

Salve o projeto com outro nome e envie ao Ludobot



The screenshot shows the Arduino IDE interface. At the top, the title bar reads "ufprTeste1 | Arduino 1.8.10". Below it is a menu bar with "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". A toolbar contains icons for a checkmark, a right arrow, a document, an upload arrow, and a download arrow. The main editor area shows a sketch named "ufprTeste1" with the following code:

```
10
11 #include "U8glib.h"
12
13 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);
14
15 #define ufprTeste1_width 128 //troque o nome
16 #define ufprTeste1_height 64 //troque o nome
17 static unsigned char ufprTeste1_bits[] U8G_PROGMEM = { //troque o nome
18     //cole aqui o conteúdo da imagem a ser exibida
19     0x00, 0x70, 0x00, 0xE0, 0x00, 0xFC, 0x00, 0x00, 0x00, 0x00, 0x3F, 0x80,
20     0x07, 0x00, 0x0F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xFF, 0xFF,
21     0xFF, 0xFF, 0x1F, 0x40, 0x00, 0x00, 0x10, 0x00, 0x00, 0x01, 0x00, 0x00,
```


Comparativo



ufprAzulFundoBran
co.png



ufprAzulFundoBran
co_128_64.png



ufprAzulFundoBran
co_128_64.xbm



Comparativo



ufprFundoCinza.png



ufprFundoCinza_128_64.png



ufprFundoCinza_128_64.xbm



Comparativo



ufprAzulBranco.png



ufprAzulBranco_128_64.png



ufprAzulBranco_128_64.xbm



Exemplo – sensor de aceleração / movimento



bitmaps



controleRemoto



informacoes



LudoMotores



outros



sensorMovimento



som

Use
este
arquivo

As bibliotecas necessárias
foram deixadas na mesma
pasta, assim facilita a
compilação em vários
ambientes diferentes

Microduino_sensor
Motion.cppMicroduino_sensor
Motion.hsensorMovimento.i
no

Código



O sensor é ligado em uma porta do tipo I2C

São três portas I2C, pode usar qualquer uma delas

```
switch (motion.getSensorType()) {  
  case SENSOR_TYPE_SENSORMOTION:  
    Serial.println("O sensor eh do tipo Sensor Motion");  
    break;  
  case SENSOR_TYPE_MMA7660:  
    Serial.println("O sensor eh do tipo MMA7660");  
    break;  
  case SENSOR_TYPE_UNKNOWN:  
    Serial.println("ERRO - sensor desconhecido");  
    break;  
}
```

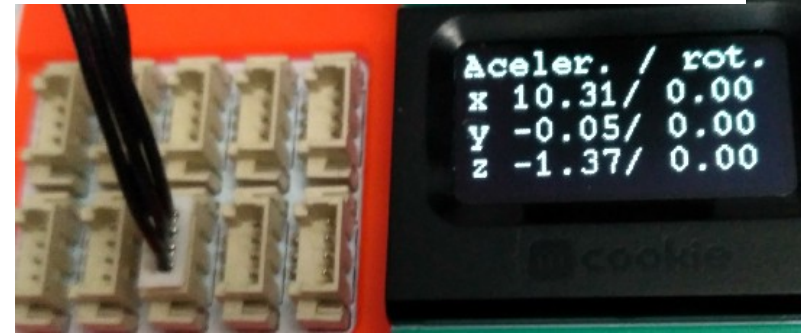
A biblioteca só funciona com os modelos 'SENSORMOTION' ou 'MMA7660'

A biblioteca retorna 6 valores:

```
motion.getMotionRaw6(&ax, &ay, &az, &rx, &ry, &rz);
```

3 de aceleração e 3 de rotação

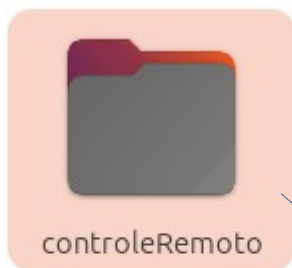
```
float ax, ay, az; //aceleração  
float rx, ry, rz; //rotação
```



Exemplo – controle remoto 1



bitmaps



controleRemoto



informacoes



LudoMotores



outros



sensorMovimento



som



leControleRemoto



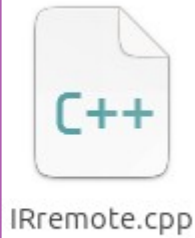
leControleRemoto
Texto



leControleRemoto
TextoComRepetica
o



**Códigos
emitidos
pelo
controle
remoto
testado**



Bibliotecas

Código





leControleRemoto

IRremote.cpp

IRremote.h

IRremoteInt.h

```

1 /*
2  * IRremote: IRecvDemo - demonstrates receiving IR codes with IRecv
3  * An IR detector/demodulator must be connected to the input RECV_PIN.
4  * Version 0.1 July, 2009
5  * Copyright 2009 Ken Shirriff
6  * http://arcfn.com
7  */
8
9 #include "IRremote.h"
10
11 int RECV_PIN = 6;
12
13 IRrecv irrecv(RECV_PIN);
14
15 decode_results results;
16
17 void setup()
18 {
19   Serial.begin(9600);
20   irrecv.enableIRIn(); // Start the receiver
21 }
22
23 void loop() {
24   if (irrecv.decode(&results)) {
25     Serial.println(results.value, HEX);
26     irrecv.resume(); // Receive the next value
27   }
28 }

```

O sensor
foi
conectado
na porta
D6

/dev/ttyUSB0

```

FFFFFFFF
1FEF807
FFFFFFFF
1FEF807
FFFFFFFF
1FE708F
FFFFFFFF
1FE708F
FFFFFFFF
1FEE01F
1FEE01F
1FEE01F
1FEA05F
1FEA05F
FFFFFFFF
1FEA05F
1FE48B7
1FE807F
1FE40BF
FFFFFFFF
1FEC03F
1FE20DF
1FEA05F
1FE609F
1FEE01F
1FE10EF
FFFFFFFF
1FE906F
FFFFFFFF

```

À medida em que as teclas são pressionadas, o código de cada tecla (em hexadecimal) é exibido no monitor serial.

O FFFFFFFF indica que a última tecla utilizada foi mantida pressionada.



Conecte o sensor na porta digital 6, envie o código ao Ludobot e ligue o monitor serial para ver os códigos das teclas

Exemplo – controle remoto 2



Bibliotecas

Código



leControleRemoto
Texto.ino

```
35 void loop() {  
36     if (irrecv.decode(&results)) {  
37         switch (results.value){  
38             case 0x1FE48B7:  
39                 Serial.println("Tecla Liga/ desliga");  
40                 break;  
41             case 0x1FE807F:  
42                 Serial.println("Tecla A");  
43                 break;  
44             case 0x1FE40BF:  
45                 Serial.println("Tecla B");  
46                 break;  
47             case 0x1FEC03F:  
48                 Serial.println("Tecla C");  
49                 break;
```

Agora, ao invés de exibir os códigos, serão exibidas mensagens para cada tecla.

Mesmo que a tecla seja mantida pressionada e código não será considerado, pois o 'FFFFFFFF' não está sendo testado.

Arquivo Editar Sketch Ferramen

/dev/ttyUSB0



leControleRemotoTexto

IRn

```
1 /*
2 Repetição FFFFFFFF
3 Liga/Desliga 1FE4
4 A 1FE807F
5 B 1FE40BF
6 C 1FEC03F
7 D 1FE20DF
8 E 1FE609F
9 Seta_para_Cima 1F
10 Seta_paraEsquerda
11 Seta_paraDireita
```

```
Tecla Liga/ desliga
Tecla Seta para cima
Tecla OK
Tecla Seta para baixo
Tecla Seta para direita
Tecla E
Tecla C
Tecla Desligar o som
Tecla Voltar
```

Exemplo – controle remoto 3



IRremote.cpp



IRremote.h



IRremoteInt.h

Bibliotecas

Código



leControleRemoto
TextoComRepetica
o.ino


```
36 void loop() {  
37   if (irrecv.decode(&results)) {  
38     if(results.value != 0xFFFFFFFF){  
39       tecla = results.value;  
40     }  
41     switch (tecla){  
42       case 0x1FE48B7:  
43         Serial.println("Tecla Liga/ desliga");  
44         break;  
45       case 0x1FE807F:  
46         Serial.println("Tecla A");  
47         break;  
48       case 0x1FE40BF:  
49         Serial.println("Tecla B");  
--
```

Agora, ocorre o teste de 'FFFFFFFF' : Se o valor não for (!=) o 'FFFFFFFF', será lido o valor da tecla e em seguida ocorre a exibição da mensagem correspondente. Se o valor for 'FFFFFFFF' o programa não faz nada, e, como a última tecla lida está na variável, ele repete a mensagem (enquanto a tecla ficar pressionada).

Arquivo Editar Sketch Ferramentas Ajuda



/dev/ttyUSB0

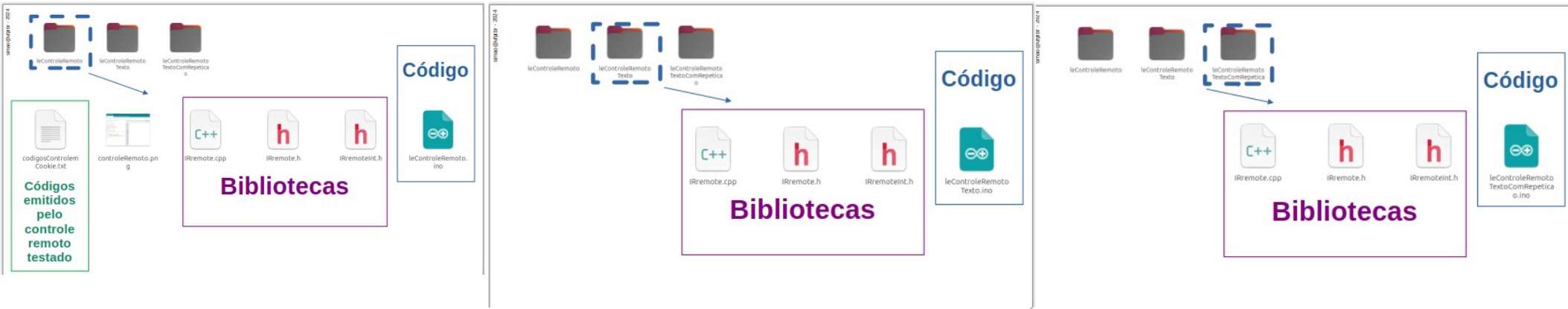
leControleRemotoTextoComRepeticao

```
36 void loop() {
37     if (irrecv.decode(&Tecla OK
38         if(results.value Tecla OK
39         tecla = results Tecla OK
40     }
41     switch (tecla){ Tecla OK
42         case 0x1FE48B7: Tecla OK
43             Serial.print Tecla OK
44             break; Tecla OK
45         case 0x1FE807F: Tecla OK
46             Serial.print Tecla OK
47             break;
48         case 0x1FE40BF:
49             Serial.print Tecla OK
50     }
```

Observe que, nos três exemplos fornecidos, as bibliotecas necessárias foram deixadas na pasta, junto com o código.

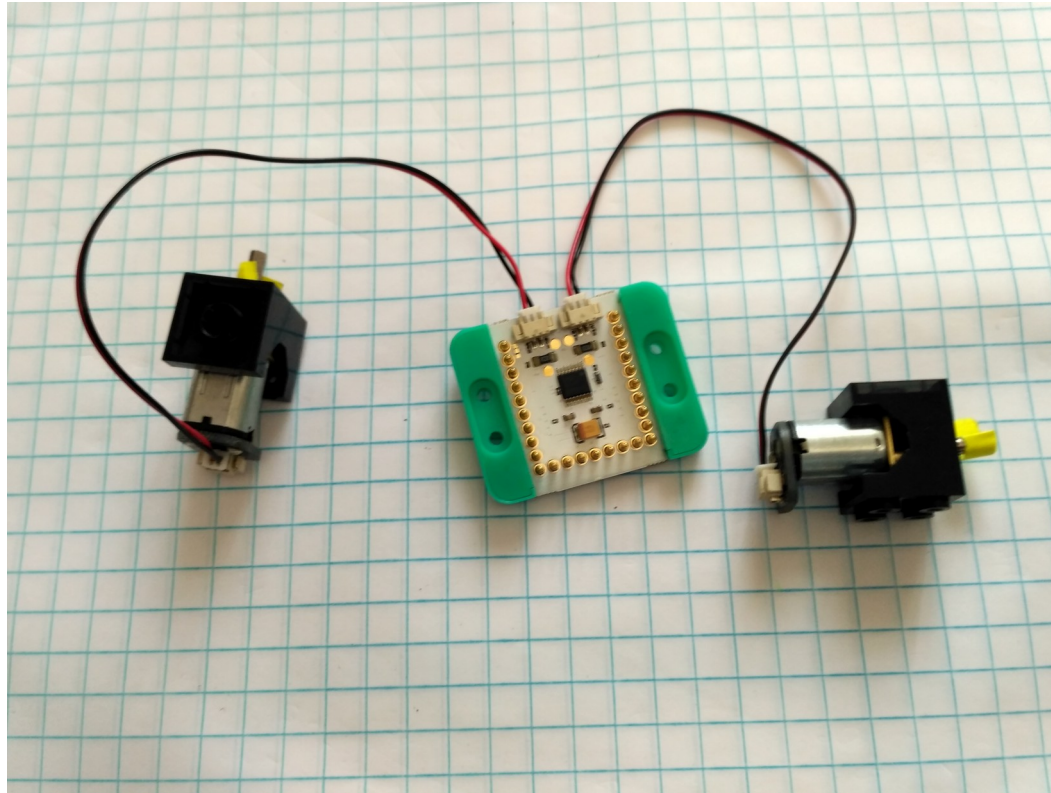
Isto facilita levar e testar o código em qualquer equipamento; porém, repete a cópia das bibliotecas em várias pastas, o que é ruim em caso de atualização (e desperdiça espaço...)

O ideal é instalar as bibliotecas necessárias na pasta de bibliotecas (*libraries*) do Arduino.



Controlando motores

Exemplo – controle de motores usando o *mCookie*





bitmaps



controleRemoto



informacoes



LudoMotores



outros



sensorMovimento



som



baseBuggy



carroControle



mCookie



servoMotor



wally

Código



mCookie.ino

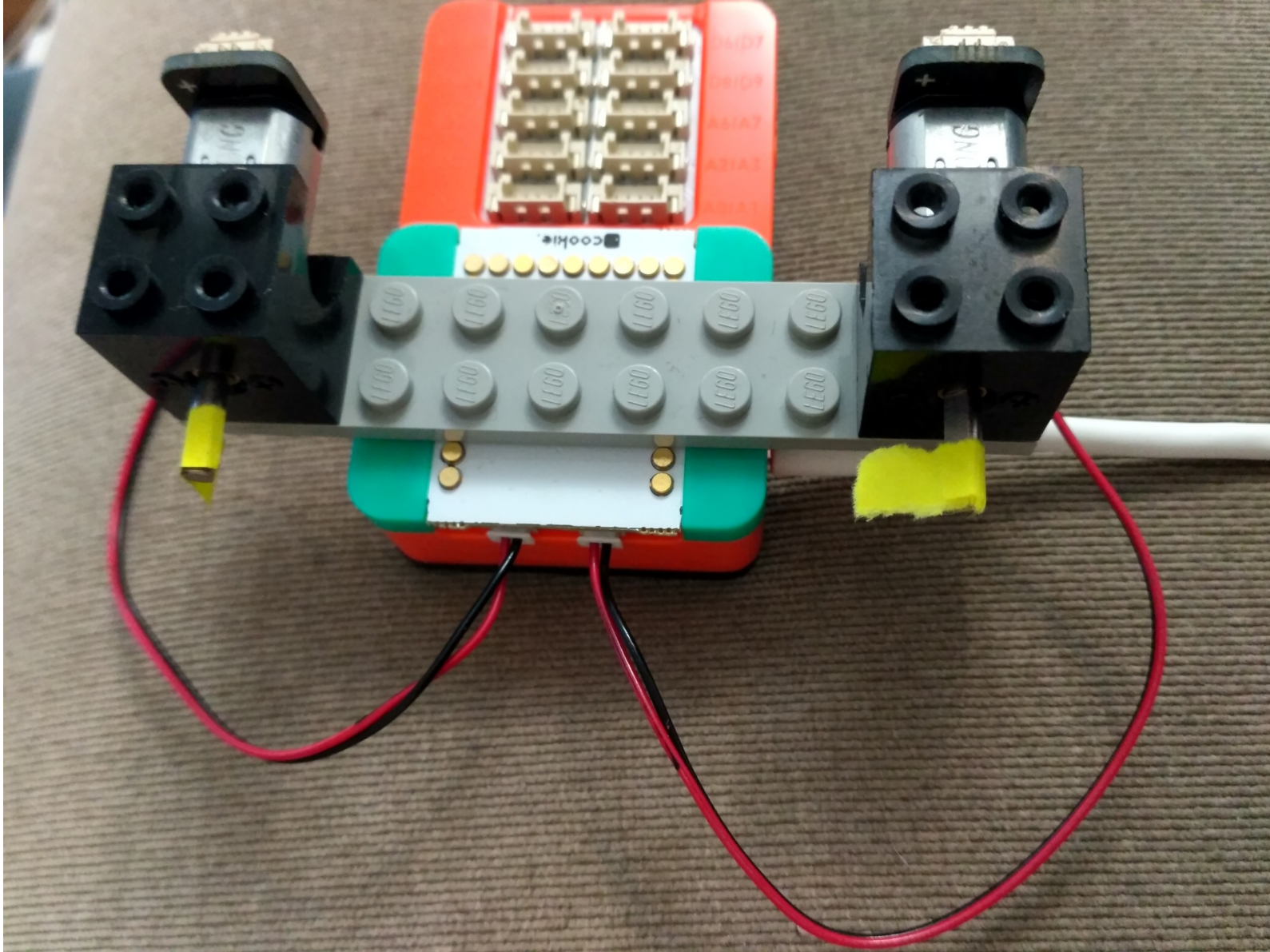


Microduino_Motor.
cpp



Microduino_Motor.
h

Bibliotecas



Biblioteca de controle do módulo mCookie

```
8 #include "Microduino_Motor.h"
9
10 Motor MotorLeft(MOTOR0_PINA, MOTOR0_PINB);
11 Motor MotorRight(MOTOR1_PINA, MOTOR1_PINB);
```

Constantes definidas na
biblioteca

Velocidade positiva = sentido de movimento depende da montagem

```
25 MotorLeft.setSpeed(100);  
26 //Defina a velocidade do motorMotorRight para 100  
27 MotorRight.setSpeed(100);  
  
38 //Defina a velocidade MotorLeft do motor para -100 = gira para trás  
39 MotorLeft.setSpeed(-100);
```

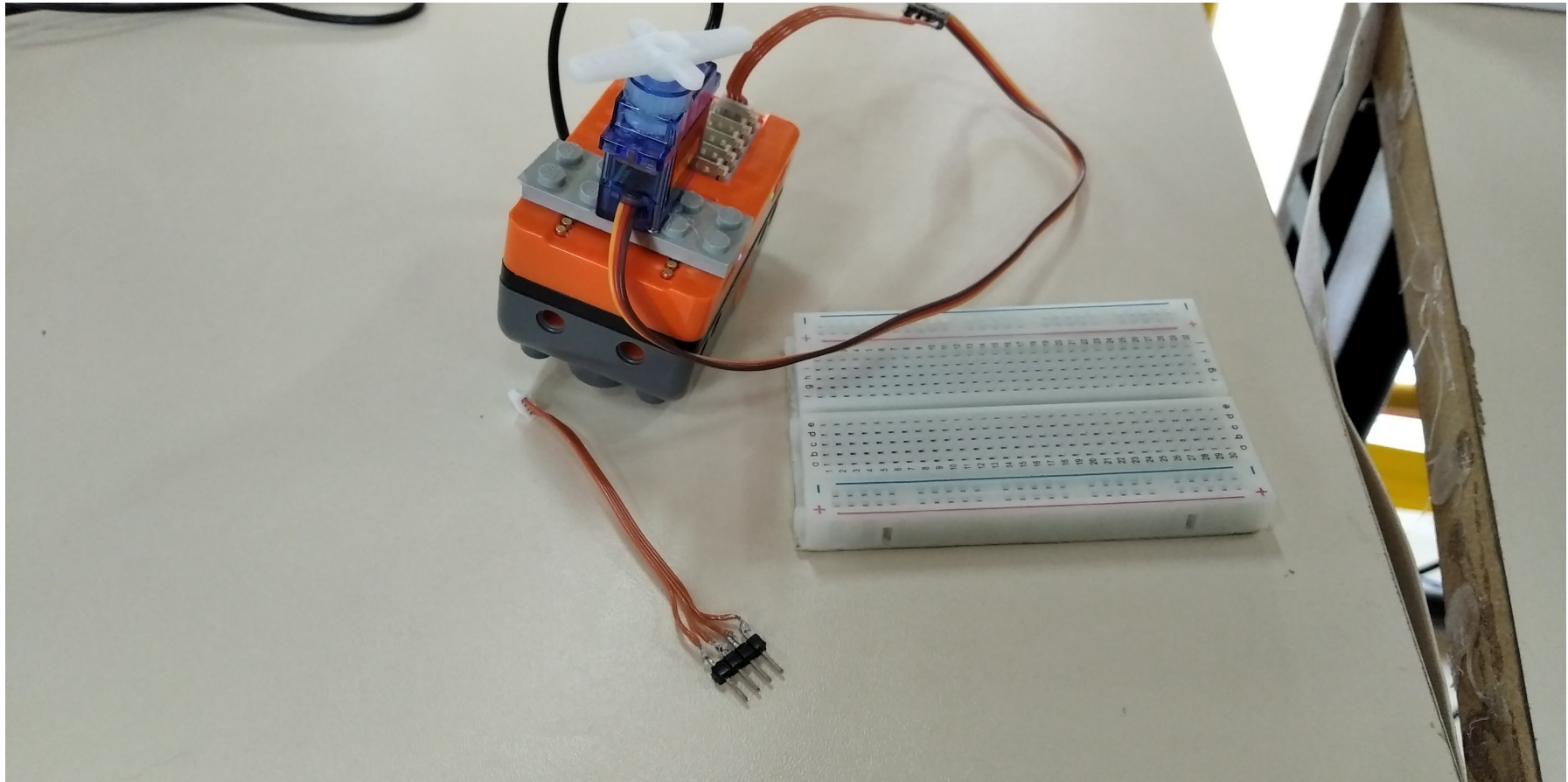
Velocidade negativa = sentido de movimento inverso

Parar motor

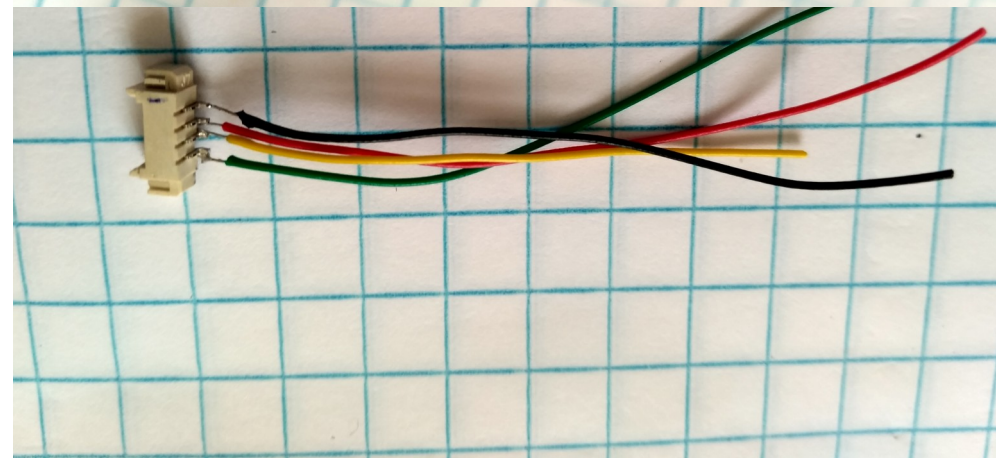
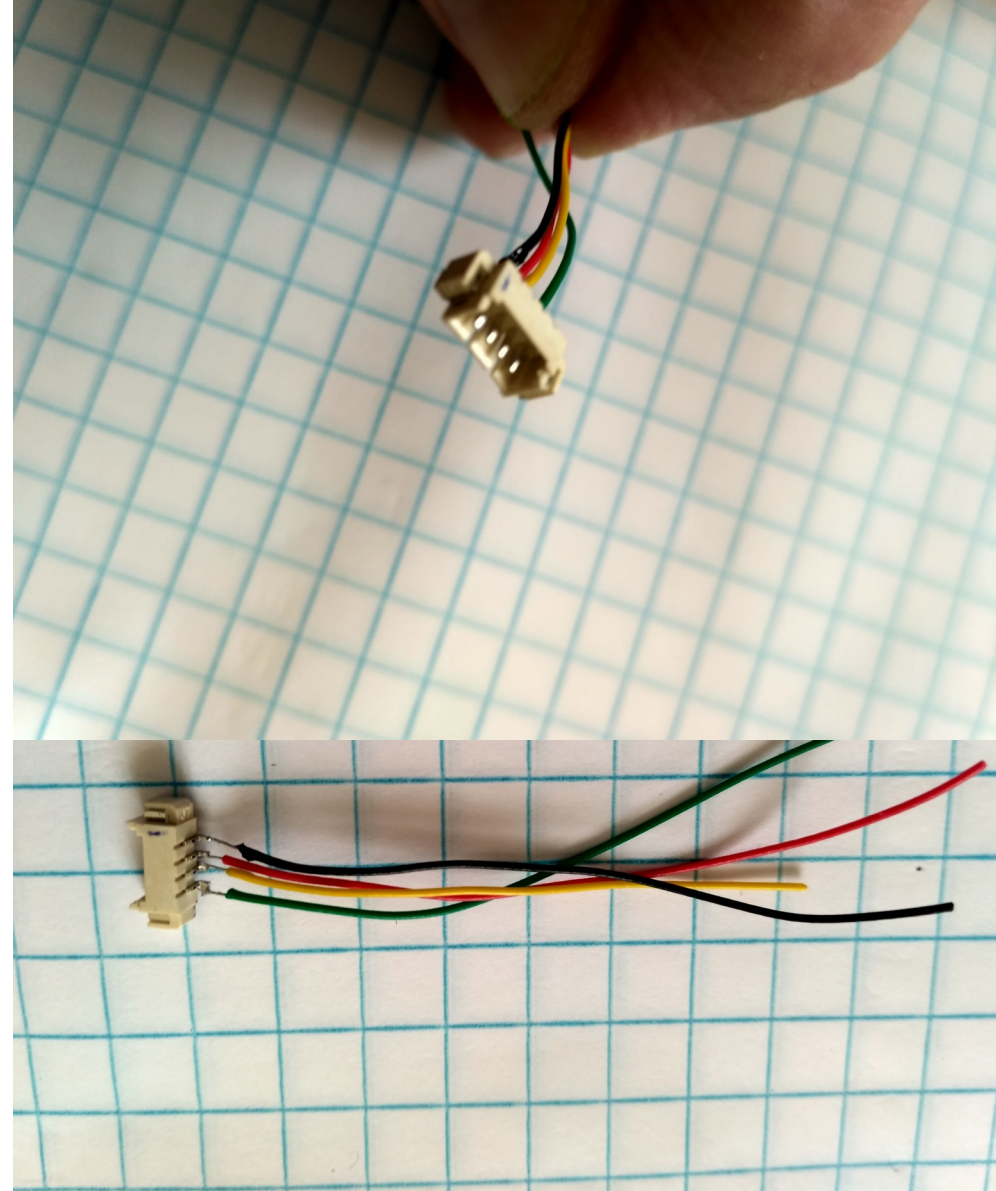
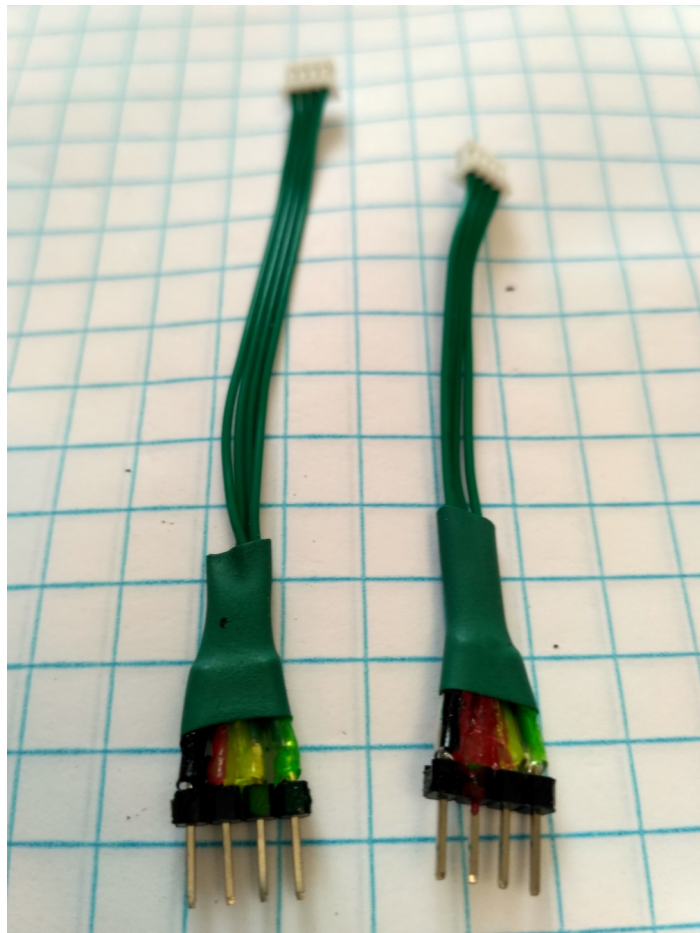
```
32 MotorLeft.Brake();  
33 //Parar motor direito
```

```
45 //Defina o motor MotorLeft para o estado de liberação, ou seja, a velocidade é 0  
46 MotorLeft.setSpeed(FREE);
```

Exemplo – controle de servo motor



Você vai precisar de cabos para conexão do servo motor ao Ludobot





bitmaps



controleRemoto



informacoes



LudoMotores



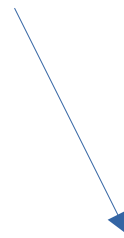
outros



sensorMovimento



som



baseBuggy



carroControle



mCookie



servoMotor



wally



Servo.cpp



Servo.h

Bibliotecas

Código



servoMotor.ino

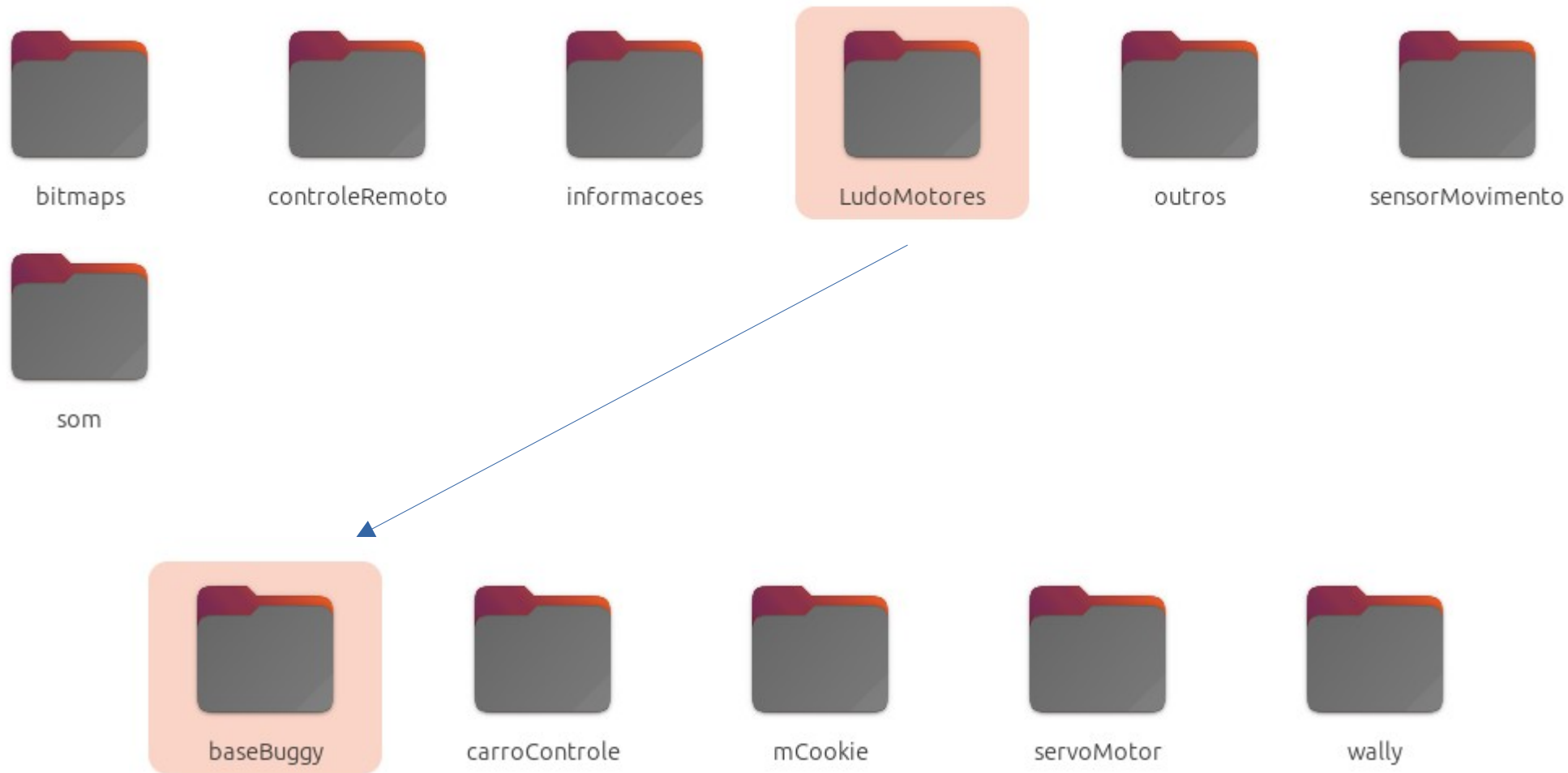

```
7 #define servoPin 6 //Se trocar o pino, ajuste aqui
```

Pino de conexão
do controle do
servo

```
9 int pos = 0;
10
11 void setup()
12 {
13     myservo.attach(servoPin);
14 }
15
16 void loop()
17 {
18     for (pos = 0; pos < 180; pos += 1)
19     {
20         myservo.write(pos);
21         delay(15);
22     }
23     for (pos = 180; pos >= 1; pos -= 1)
24     {
25         myservo.write(pos);
26         delay(15);
27     }
28 }
```




Exemplo –
controle da base *buggy*



Código



baseBuggy.ino



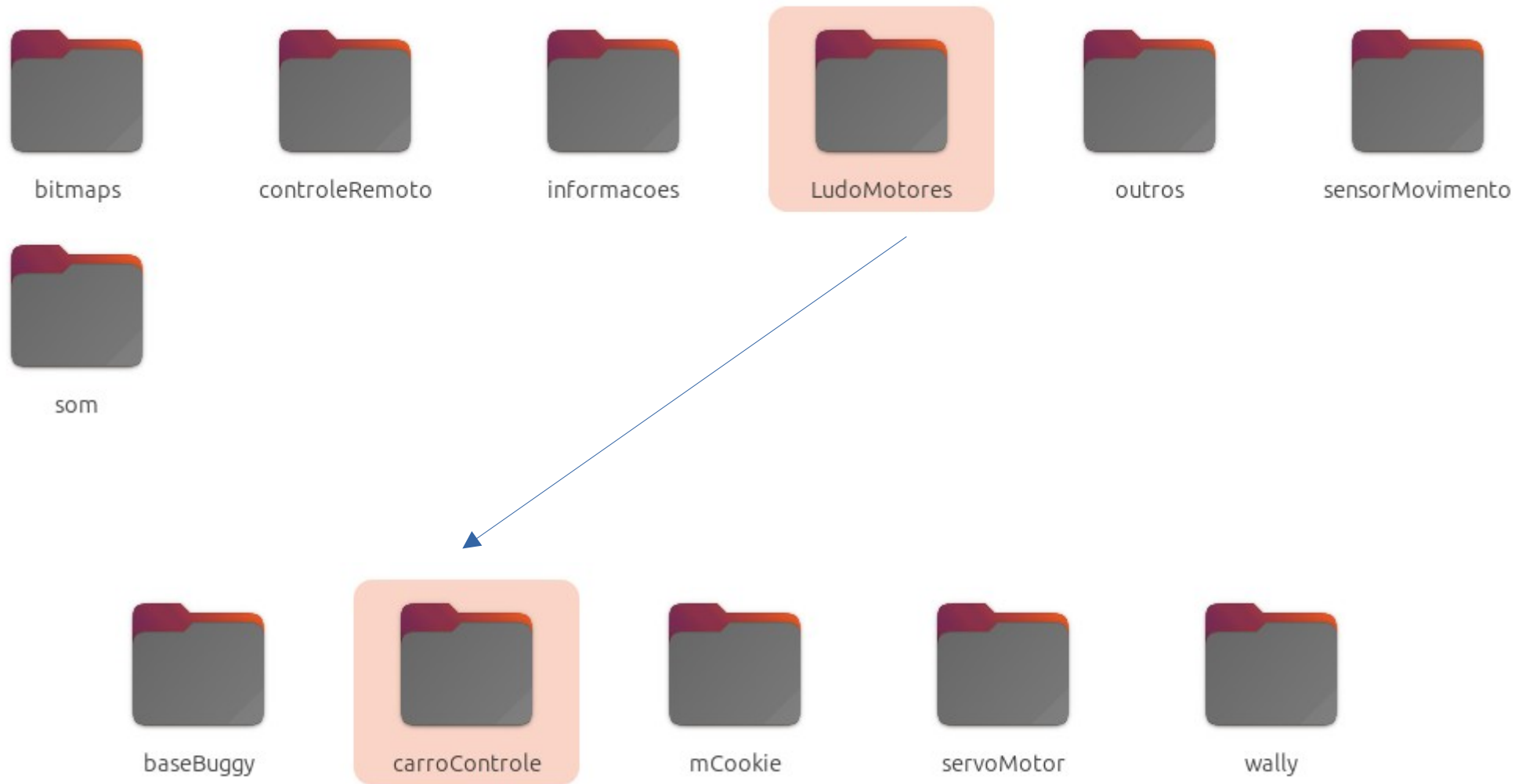
BittyBuggy.cpp



BittyBuggy.h

Bibliotecas

Exemplo – controle da base ***buggy*** por meio do controle remoto



Código



BittyBuggy.cpp



BittyBuggy.h



carroControle.ino



IRremote.cpp



IRremote.h



IRremoteInt.h

**Bibliotecas
da base
*buggy***

**Bibliotecas do
controle remoto**

```
11 #include "BittyBuggy.h"
12
13 #define mySerial Serial1
14 BittyBuggy car(&mySerial);
```

O objeto 'car' (pode trocar o nome) terá acesso a todos os controles do *bitty buggy* por meio da *interface* serial

BittyBuggy.h

```
52 #define COLOR_NONE 0X000000
53 #define COLOR_BLACK 0X000000
54 #define COLOR_WHITE 0XFFFFFF
55 #define COLOR_RED 0XFF0000
56 #define COLOR_ORANGE 0XEA8B17
57 #define COLOR_YELLOW 0XFFFF00
58 #define COLOR_GREEN 0X00FF00
59 #define COLOR_CYAN 0X00FFFF
60 #define COLOR_BLUE 0X0000FF
61 #define COLOR_PURPLE 0X4B0082
62 #define COLOR_WARM 0X9B6400
63 #define COLOR_COLD 0X648264
```

```
41 #define MINSPEED 45
42 #define MAXSPEED 255
```

```
65 #define BLACK 0
66 #define RED 1
67 #define GREEN 2
68 #define BLUE 3
```


BittyBuggy.h

```
93 class BittyBuggy
94 {
95     public:
```

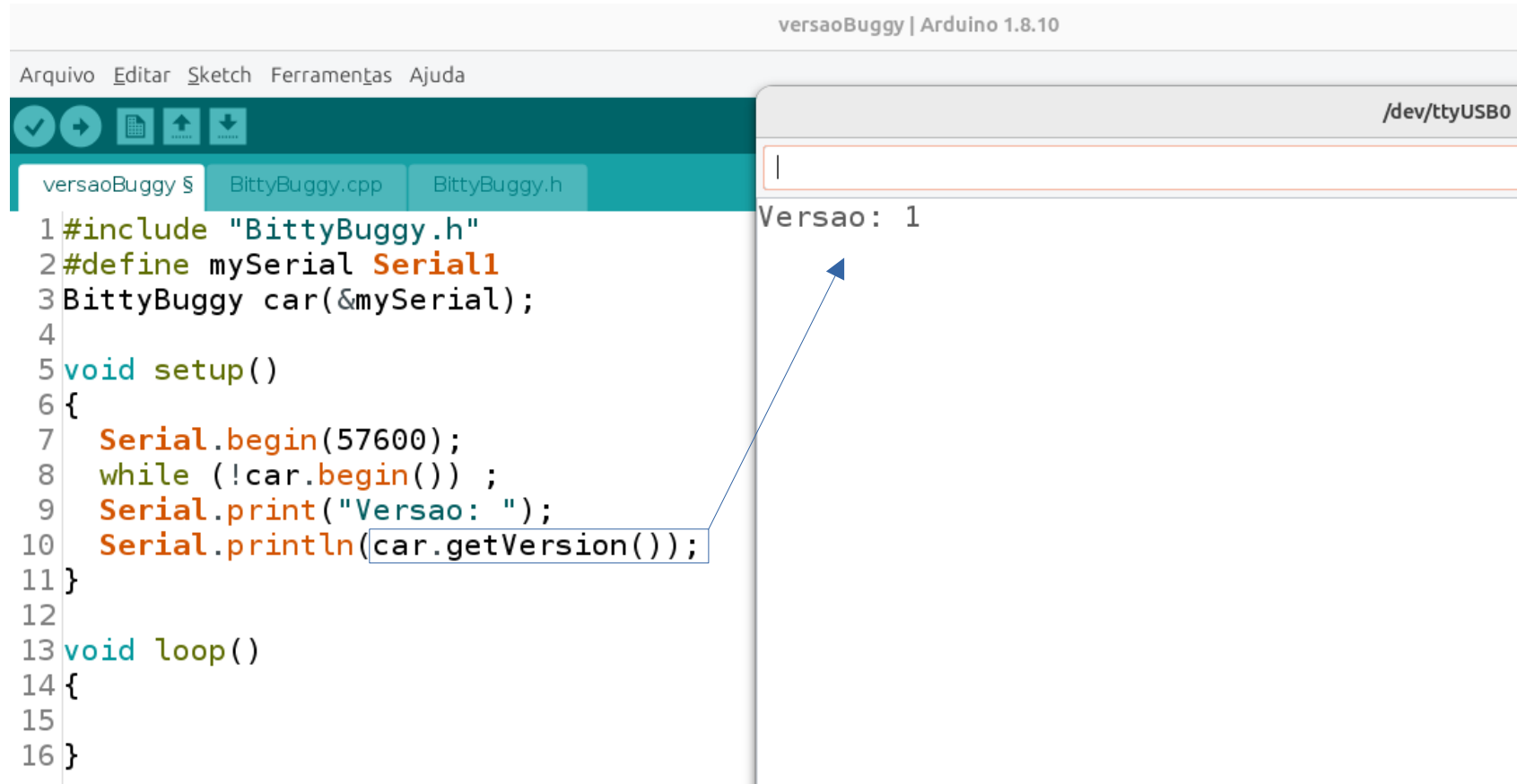
Só podemos
acessar o que
for público

```
112 void setColorLEDA(uint8_t ar, uint8_t ag, uint8_t ab);
113 void setColorLEDB(uint8_t br, uint8_t bg, uint8_t bb);
114 void setColorLEDA(uint32_t _rgb);
115 void setColorLEDB(uint32_t _rgb);
116 void setAllLED(uint32_t _rgb);
117 void setAllLED(uint8_t ar, uint8_t ag, uint8_t ab);
118 void setSpeed(int16_t left, int16_t right);
119 void setSpeedA(int16_t left);
120 void setSpeedB(int16_t right);
121 void tone(int16_t _fre, int16_t _time);
122 void tone(int16_t _fre);
123 void noTone();
124 uint8_t getVersion();
125 void getLineAlone(uint8_t *_array, uint8_t _colorA = RED, uint8_t _colorB = RED);
126 void getLine(uint8_t *_array, uint8_t _color = RED);
127 uint8_t getLineA(uint8_t _color = RED);
128 uint8_t getLineB(uint8_t _color = RED);
129 void getLineLowpassFilter(uint8_t *_array, uint8_t _color, float _num = 0.5);
130 void getLineAverageFilter(uint8_t *_array, uint8_t _color, uint8_t len = 12);
```

BittyBuggy.h

```
112 void setColorLEDA(uint8_t ar, uint8_t ag, uint8_t ab);
113 void setColorLEDB(uint8_t br, uint8_t bg, uint8_t bb);
114 void setColorLEDA(uint32_t _rgb);
115 void setColorLEDB(uint32_t _rgb);
116 void setAllLED(uint32_t _rgb);
117 void setAllLED(uint8_t ar, uint8_t ag, uint8_t ab);
118 void setSpeed(int16_t left, int16_t right);
119 void setSpeedA(int16_t left);
120 void setSpeedB(int16_t right);
121 void tone(int16_t _fre, int16_t _time);
122 void tone(int16_t _fre);
123 void noTone();
124 uint8_t getVersion();
125 void getLineAlone(uint8_t *_array, uint8_t _colorA = RED, uint8_t _colorB = RED);
126 void getLine(uint8_t *_array, uint8_t _color = RED);
127 uint8_t getLineA(uint8_t _color = RED);
128 uint8_t getLineB(uint8_t _color = RED);
129 void getLineLowpassFilter(uint8_t *_array, uint8_t _color, float _num = 0.5);
130 void getLineAverageFilter(uint8_t *_array, uint8_t _color, uint8_t len = 12);
```

BittyBuggy.h – exemplo: getVersion()



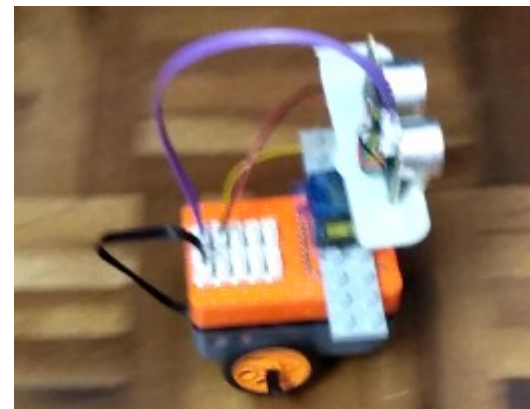
The screenshot displays the Arduino IDE interface. The top menu bar includes 'Arquivo', 'Editar', 'Sketch', 'Ferramentas', and 'Ajuda'. The toolbar contains icons for checking, running, saving, and uploading. The file explorer shows 'versaoBuggy S', 'BittyBuggy.cpp', and 'BittyBuggy.h'. The main editor window displays the code for 'BittyBuggy.h'.

```
1 #include "BittyBuggy.h"
2 #define mySerial Serial1
3 BittyBuggy car(&mySerial);
4
5 void setup()
6 {
7     Serial.begin(57600);
8     while (!car.begin());
9     Serial.print("Versao: ");
10    Serial.println(car.getVersion());
11 }
12
13 void loop()
14 {
15
16 }
```

The serial monitor on the right shows the output 'Versao: 1'. A blue arrow points from the `car.getVersion()` call in the code to the output in the serial monitor.



Exemplo – Wall-e







BittyBuggy.cpp



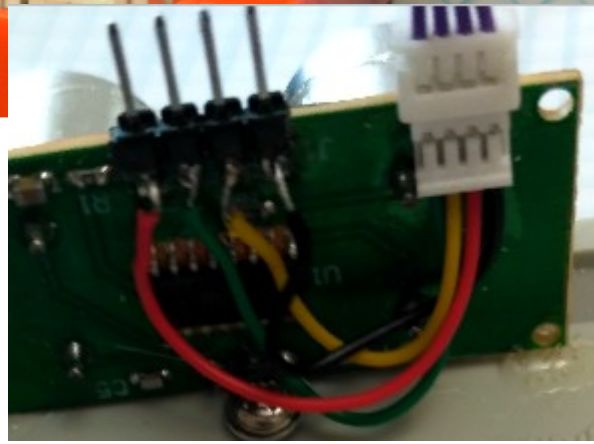
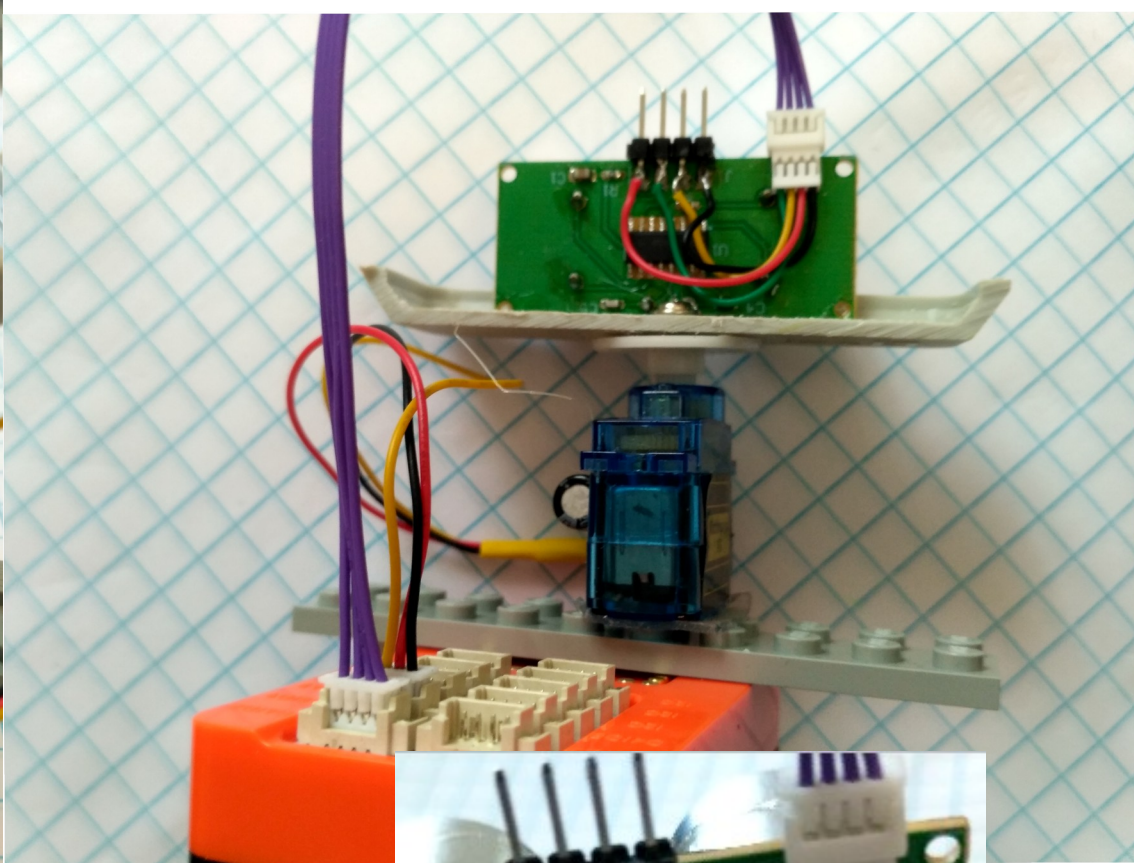
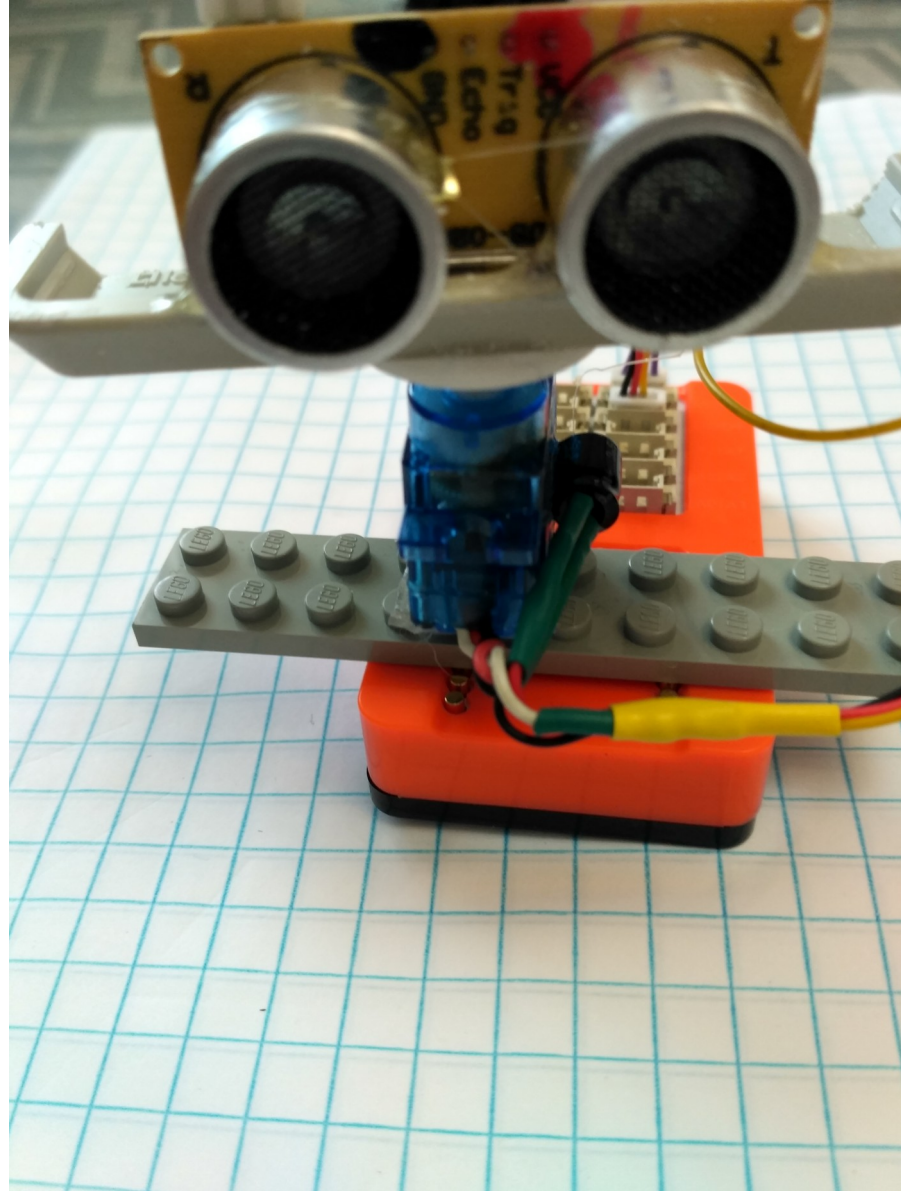
BittyBuggy.h

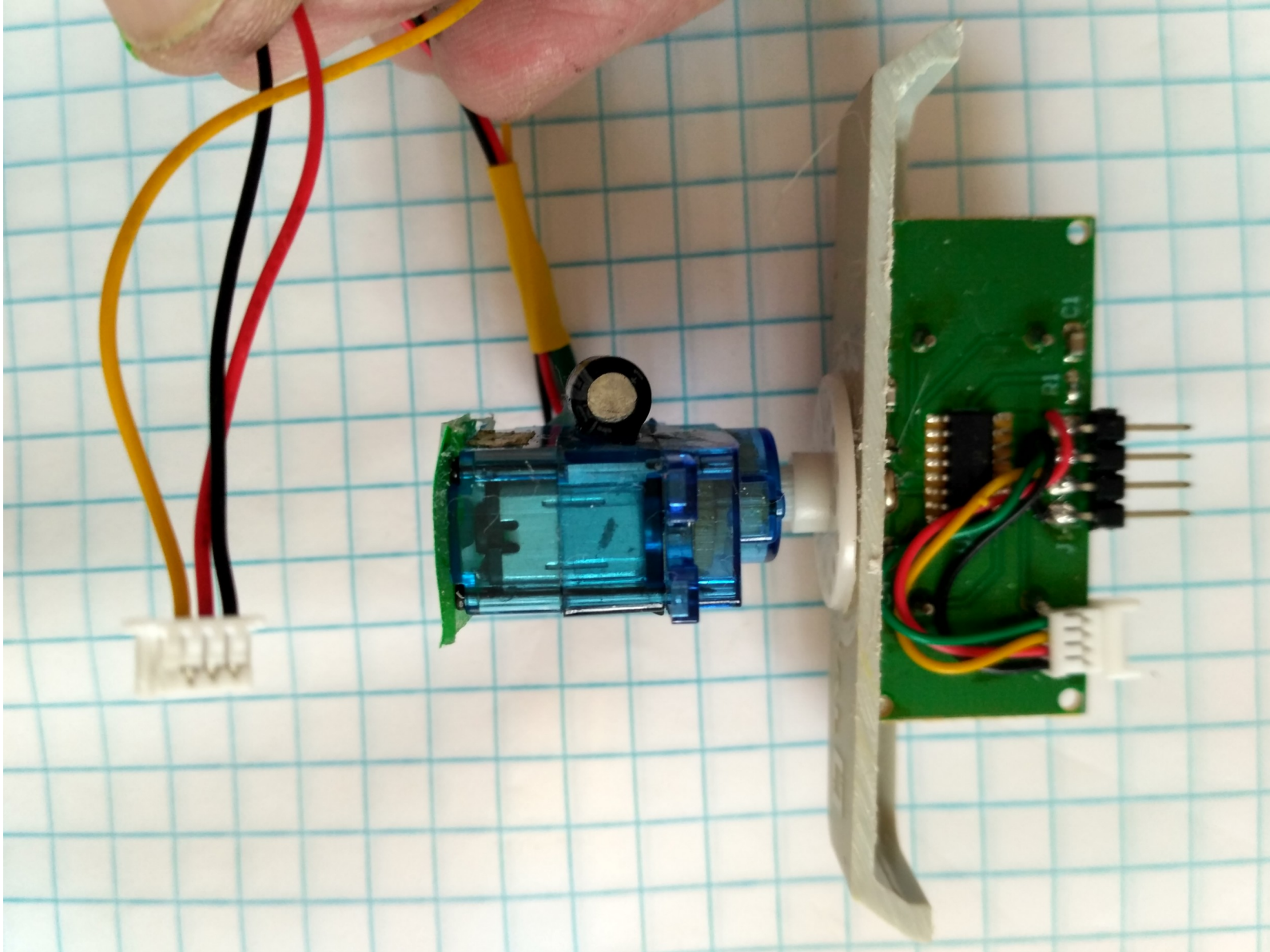
Bibliotecas

Código



wally.ino






```

13 //
14 //      -                      -      -> parte aberta
15 //      |                      |
16 //      |  o      o      o      o  |      -> onde estão os pinos
17 //      |-----|
18 //      GND VCC  D8  D9      (o D9 não foi usado no meu exemplo)
19 //
20 //
21 //      -                      -      -> parte aberta
22 //      |                      |
23 //      |  o      o      o      o  |      -> onde estão os pinos
24 //      |-----|
25 //      GND VCC  D6  D7

```

Ligações

```
27 const int PinoServoMotor = 8; // D8/D9, usei o D8 - No conector: GND-5V-D8 (D9 sem uso)
28 //ligação do servo motor:
29 //GND (em geral marrom ou preto) no GND
30 //VCC (em geral vermelho) no 5V
31 //Controle (em geral laranja ou branco) no D8
```

```
33 const int PinoTrigger = 7; // D6/D7 - No conector: GND-5V-D6-D7
34 const int PinoEcho = 6;
35 //ligação do sensor de ultrassom:
36 //GND no GND
37 //VCC no 5V
38 //Trig/trigger no D7
39 //Echo no D6
```

```
72 void loop()
73 {
74     //em frente
75     car.setSpeed(80, 80);
76     meuServo.write(90);
77     distancia = calculaDistancia();
78     if (distancia < 6){
79         //retorna um pouquinho
80         car.setSpeed(-50, -50);
81         delay(1000);
82         //para
83         car.setSpeed(BRAKE, BRAKE);
84         meuServo.write(45);
85         int esquerda = calculaDistancia();
86         delay(50);
87         meuServo.write(135);
88         int direita = calculaDistancia();
89         if ( esquerda < direita ){
90             car.setSpeed(50, -100);
91             delay(300);
92         }
93         else {
94             car.setSpeed(-100, 50);
95             delay(300);
96         }
97     }
98     delay(100);
99 }
```

O carrinho se movimenta para frente (linha 75). A distância é lida (linha 77); se for menor do que 6 (linha 78) torna a velocidade negativa (linha 80), o que faz o carrinho retroceder, por 1 segundo (linha 81) e depois ele para (linha 83).

Então, gira o servo para a posição de 45° (o que está à esquerda, pois o padrão é 90°, que é em frente). Le a distância de um obstáculo no lado esquerdo (linha 85); depois gira para a direita até 135° (linha 87) e lê a distancia novamente.

Finalmente, compara a distância de um obstáculo da esquerda e direita, virando para o lado em que ela for maior: para isso, a velocidade de um dos motores é ajustada como positiva e a do outro como negativa, o que faz com que um deles gire para frente e o o outro para trás.

Exemplo – Som pelo *buzzer* da base ***buggy***



bitmaps



controleRemoto



informacoes



LudoMotores



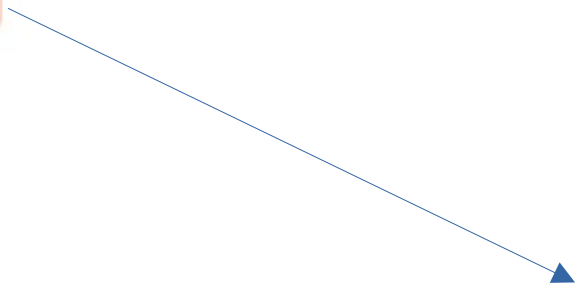
outros



sensorMovimento



som



lightTheremin



somBuggy



somBuggyParabens



BittyBuggy.cpp



BittyBuggy.h

Bibliotecas

Código



somBuggy.ino

Exemplo – Som pelo *buzzer* da base ***buggy***

Parabéns pra você



bitmaps



controleRemoto



informacoes



LudoMotores



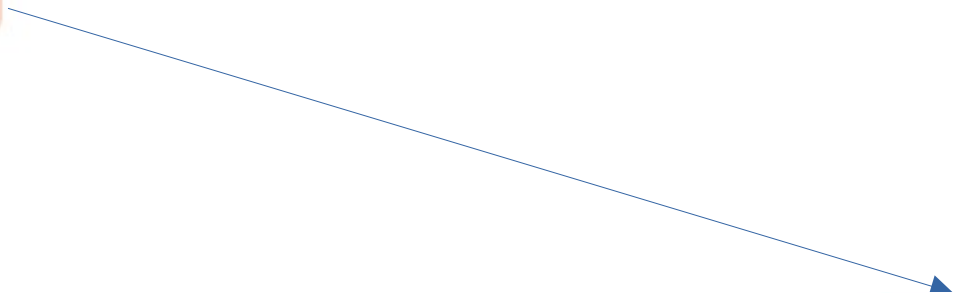
outros



sensorMovimento



som



lightTheremin



somBuggy



somBuggyParabens



BittyBuggy.cpp



BittyBuggy.h

Bibliotecas

Código



somBuggyParabens
.ino

Exemplo – Som pelo *buzzer* da base ***buggy***

Theremin controlado pela luz

pt.wikipedia.org/wiki/Teremim

O **teremim** ou theremin é um dos primeiros instrumentos musicais completamente eletrônicos, controlado sem qualquer contato físico pelo músico.

Seu nome vem da versão ocidental do nome do seu inventor, o russo **Léon Theremin**, que patenteou seu dispositivo em 1928. O instrumento é controlado através de duas antenas de metal, que percebem a posição das mãos do músico e controlam osciladores de frequência com uma das mãos, e com a outra a amplitude (volume), de forma que não seja preciso tocar no instrumento. Os sinais elétricos do teremim são amplificados e enviados para um altifalante.



bitmaps



controleRemoto



informacoes



LudoMotores



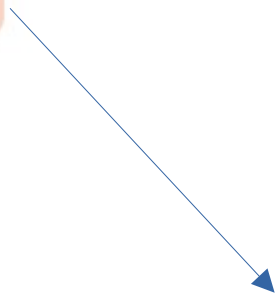
outros



sensorMovimento



som



lightTheremin



somBuggy



somBuggyParabens



BittyBuggy.cpp



BittyBuggy.h

Bibliotecas

Código



lightTheremin.ino



userDef.h

Arquivo de definições



O LDR está ligado
na porta analógica
A0

Mais exemplos



bitmaps



controleRemoto



informacoes



LudoMotores



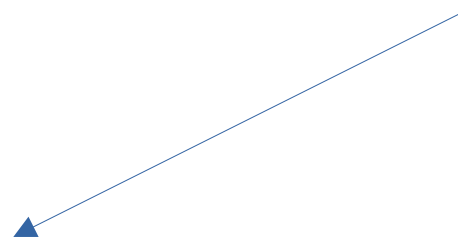
outros



sensorMovimento



som



Exemplos de fontes

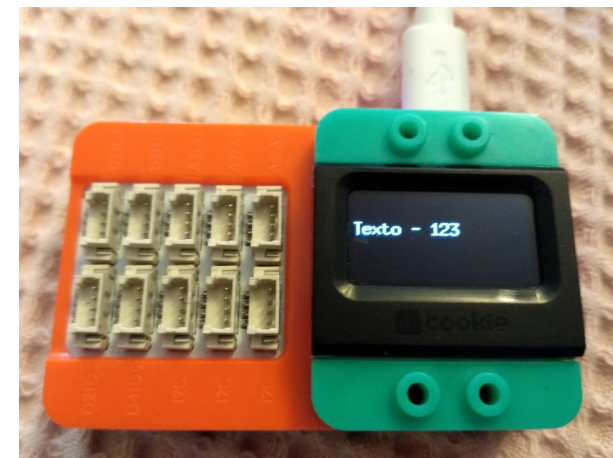
Uso do *display* LCD1602
com *interface* I2C



display1602I2C



fontesDisplay



Exemplos de fontes de caracteres



Código

```
4 #define setFont_0 u8g_font_timB14
5 #define setFont_1 u8g_font_9x15
6 #define setFont_2 u8g_font_fixed_v0r
7 #define setFont_3 u8g_font_timB18
8 #define setFont_4 u8g_font_timB14
9 #define setFont_5 u8g_font_timB10
10 #define setFont_6 u8g_font_5x7
11 #define setFont_7 u8g_font_6x10
12 #define setFont_8 u8g_font_7x13
13 #define setFont_9 u8g_font_9x15
14 #define setFont_10 u8g_font_unifont
15 #define setFont_11 u8g_font_fixed_v0r
16 #define setFont_12 u8g_font_chikitar
17
18 #define setFont_num :u8g_font_freedoomr25n
19
20 U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);
21
22 void draw(void) {
23     u8g.setFont(setFont 5); //setFont 8, etc...
```

Mude o número e compile para ver outra fonte no *display*

Display LCD-1602

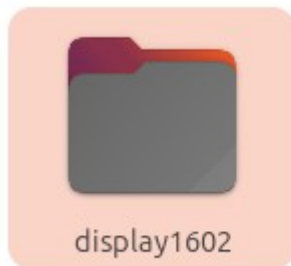
(16 colunas, 2 linhas)



display1602I2C



fontesDisplay



display1602



escreverdisplay1602

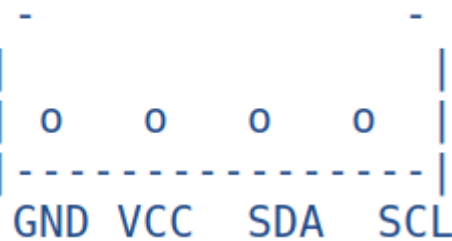
Código



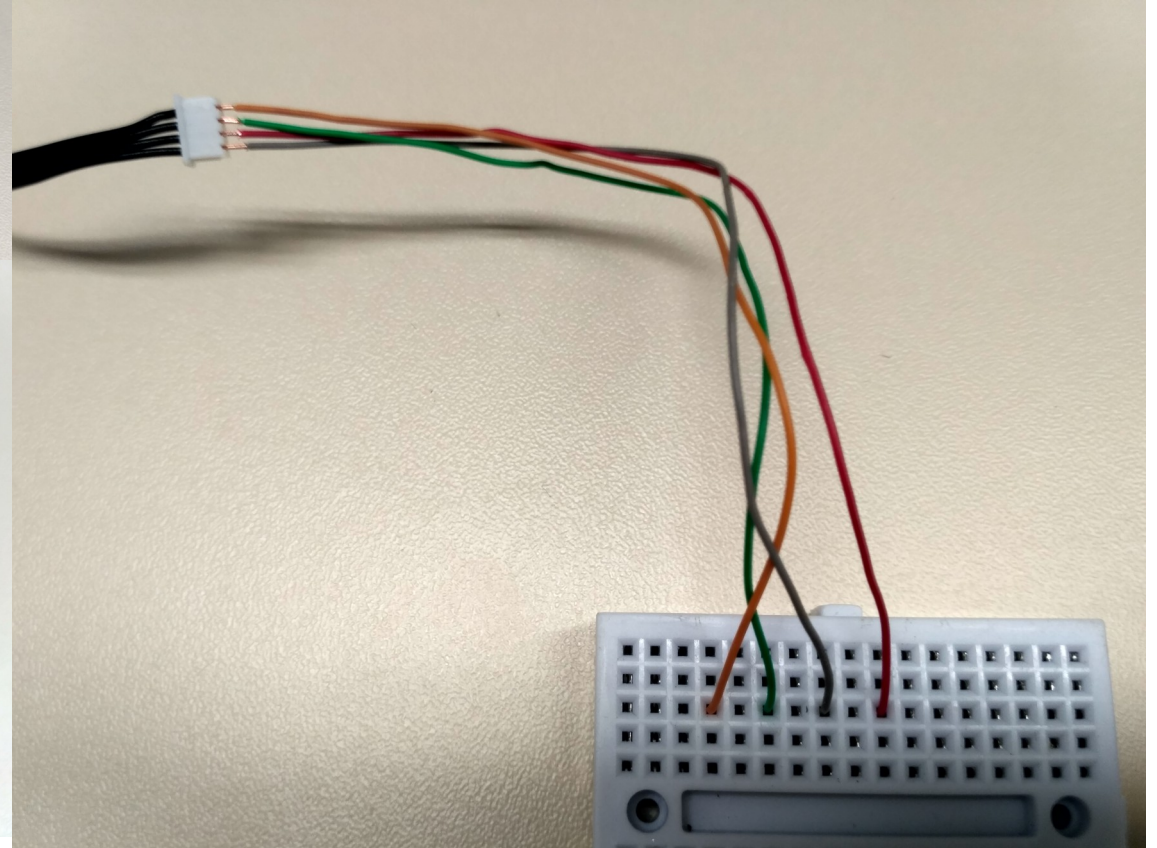
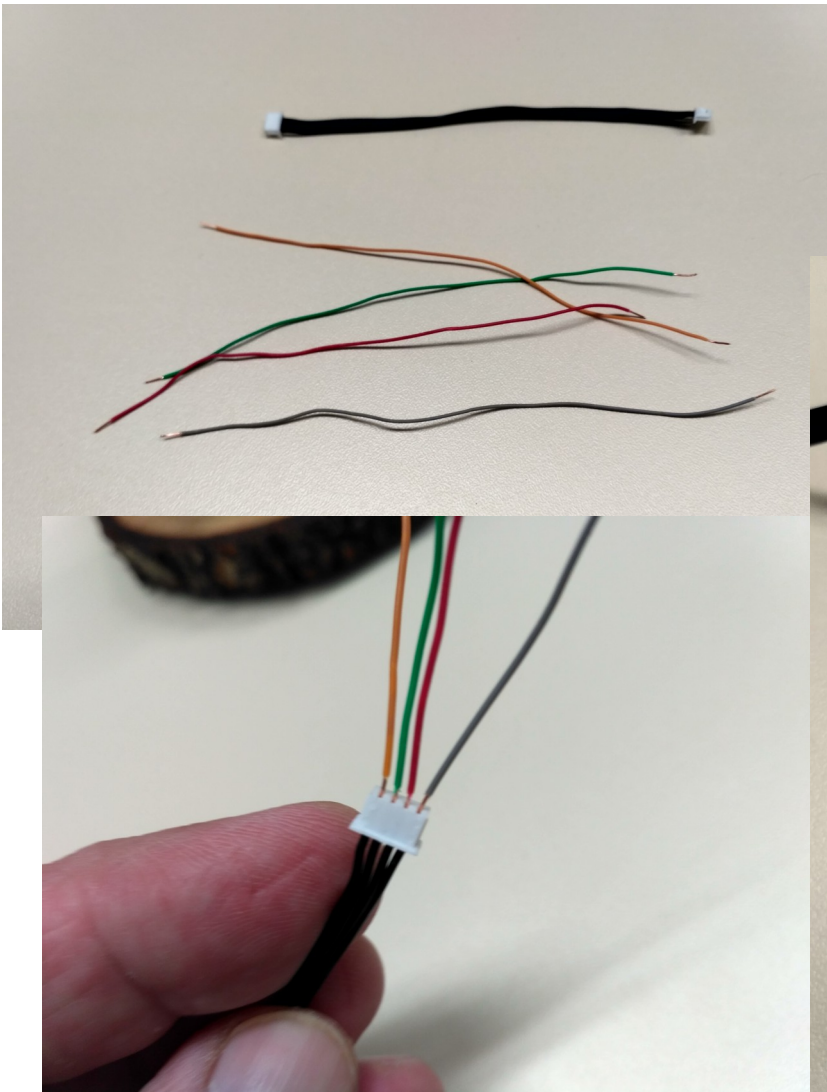
display1602.ino

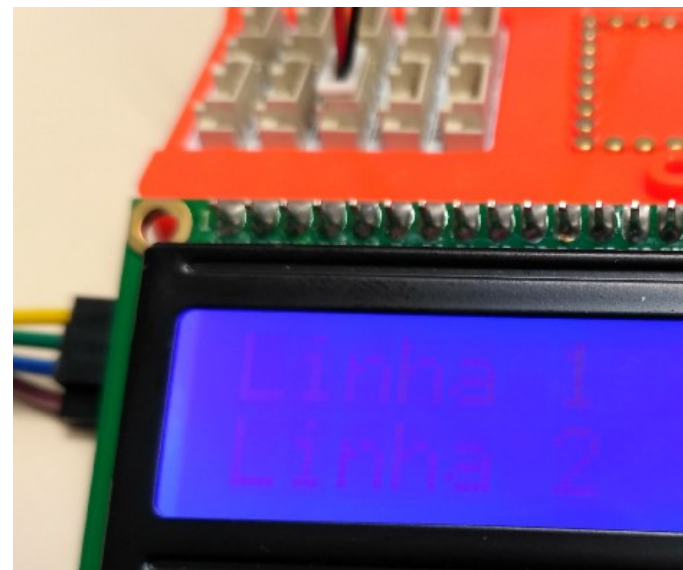
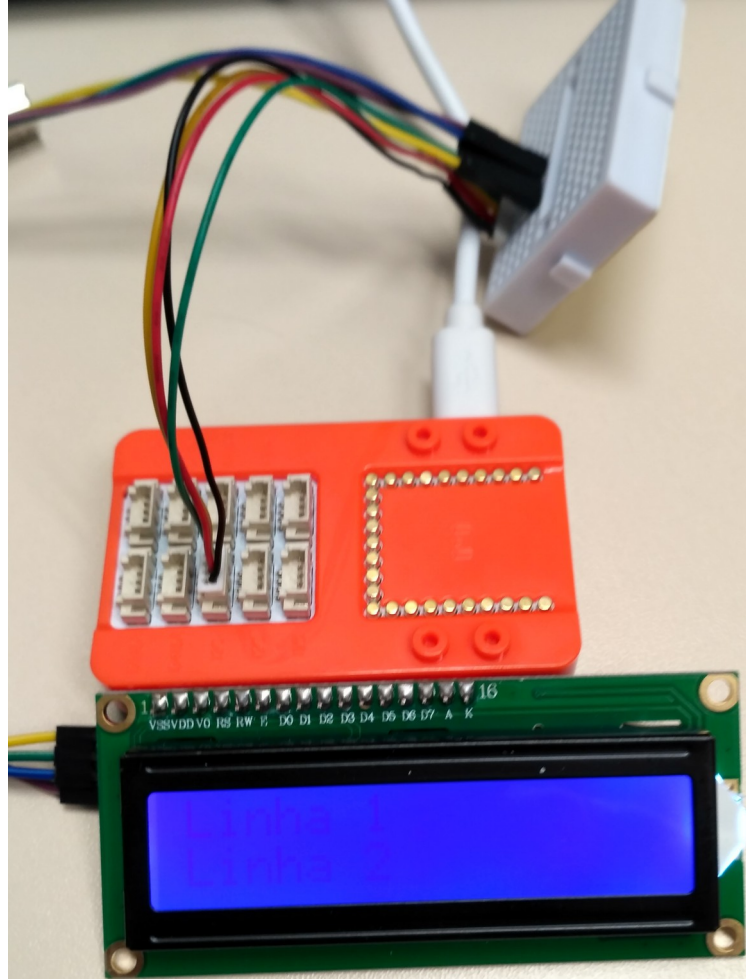
```
7 //  
8 //  
9 //  
10 //  
11 //  
12 //
```

Conector no LudoBot: usar um conector 'I2C'



Para ligar o *display* você vai precisar de um cabo





Precisa
ajustar o
contraste
(*trimpot* na
parte de
trás do
display)



Bons projetos !

