



FEITIÇOS MATEMÁTICOS

guia para interface lúdico

um jogo de Duarte Carneiro, Ivo Penedo, Vasco Borges

um projeto Universidade Lusófona — ECATI (licenciatura em Videojogos)
& Casa Pia de Lisboa – CED Jacob Rodrigues Pereira
disponível em <<http://educacaoacessivel.ulusofona.pt>>

FEITIÇOS MATEMÁTICOS é um videojogo que pode ser jogado com um teclado.
ou com um interface original concebido pelos autores do jogo.
Este documento contém as indicações fundamentais para a sua reprodução.



This work is licensed under Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 international License

> nota: docs. illustrator CS6, 123d Design e PDF + código arduino em ficheiros anexos.

COMPONENTES

1 x Arduino Micro

[http://www.ptrobotics.com/plataformaarduino-e-mod-
elos-alternativos-equivalentes/1776-arduino-micro.html](http://www.ptrobotics.com/plataformaarduino-e-mod-
elos-alternativos-equivalentes/1776-arduino-micro.html)

1x Veroboard 25x65mm 9x25 Buracos

[https://www.ptrobotics.com/pcb/413-veroboard-25x65mm-9x25bura-
cos.html?search_query=veroboard&results=37](https://www.ptrobotics.com/pcb/413-veroboard-25x65mm-9x25bura-
cos.html?search_query=veroboard&results=37)

6x Push Button SPST 24V 50mA 2Pin

[https://www.ptrobotics.com/tactile-switch/1890-push-but-
ton-spst-24v-50ma-2pin.html?search_query=tactile+push+button&results=
23](https://www.ptrobotics.com/tactile-switch/1890-push-but-
ton-spst-24v-50ma-2pin.html?search_query=tactile+push+button&results=
23)

1 x Led 10mm Red Ultra Bright

[https://www.ptrobotics.com/led-s-alto-bril-
ho/1834-led-10mm-red-ultra-bright.html?search_query=led+light&results=
176](https://www.ptrobotics.com/led-s-alto-bril-
ho/1834-led-10mm-red-ultra-bright.html?search_query=led+light&results=
176)

1 x Led 10mm Blue Ultra Bright

[https://www.ptrobotics.com/led-s-alto-bril-
ho/1989-led-10mm-blue-ultra-bright.html](https://www.ptrobotics.com/led-s-alto-bril-
ho/1989-led-10mm-blue-ultra-bright.html)

1x EasyFil PLA - Grey (Filamento de impressão 3D)

[https://www.formfutura.com/shop/product/easy-
fil-pla-grey-195?search=EasyFil+PLA](https://www.formfutura.com/shop/product/easy-
fil-pla-grey-195?search=EasyFil+PLA)

1x EasyFil PLA - Red (Filamento de impressão 3D)

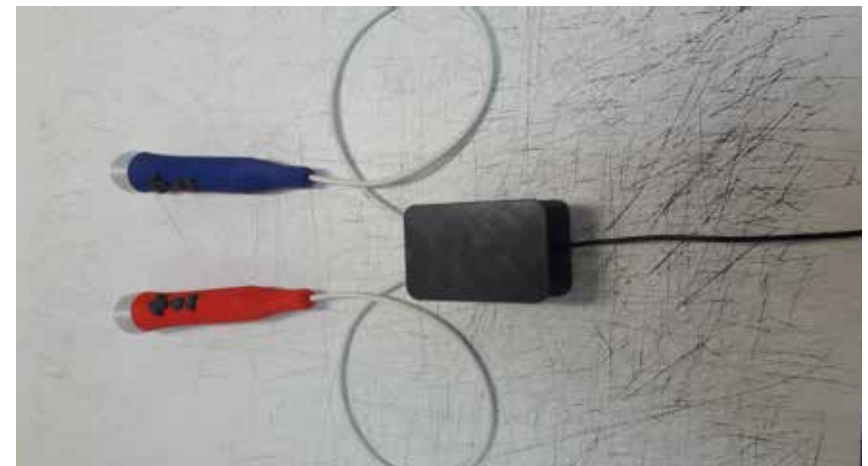
[https://www.formfutura.com/shop/product/easyfil-pla-red-203?-
search=EasyFil+PLA](https://www.formfutura.com/shop/product/easyfil-pla-red-203?-
search=EasyFil+PLA)

1x EasyFil PLA - Blue (Filamento de impressão 3D)

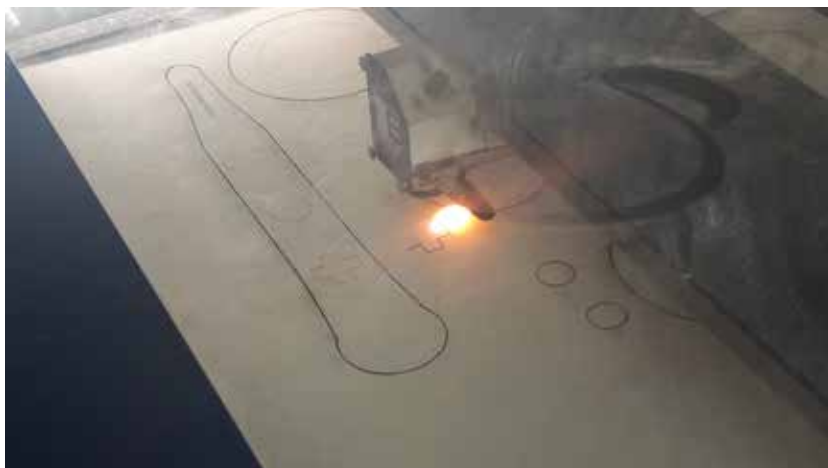
[https://www.formfutura.com/shop/product/easy-
fil-pla-light-blue-196?search=EasyFil+PLA](https://www.formfutura.com/shop/product/easy-
fil-pla-light-blue-196?search=EasyFil+PLA)

1 x EasyFil PLA - White (Filamento de impressão 3D)

[https://www.formfutura.com/shop/product/easy-
fil-pla-white-206?category=162](https://www.formfutura.com/shop/product/easy-
fil-pla-white-206?category=162)



MONTAGEM



Começámos por fazer um teste de cartão para confirmarmos as medidas do comando.

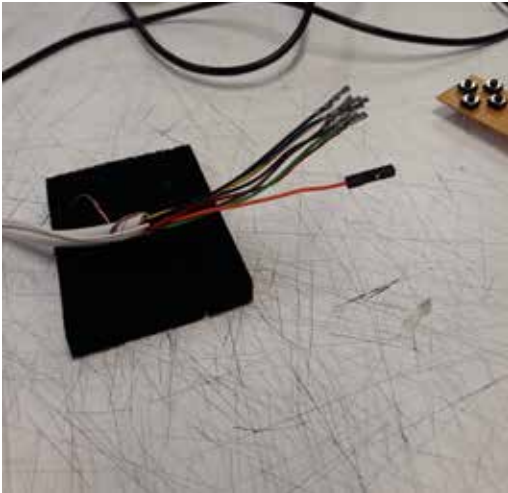


Depois de imensos testes e ajustes, chegámos ao produto final de 3D.

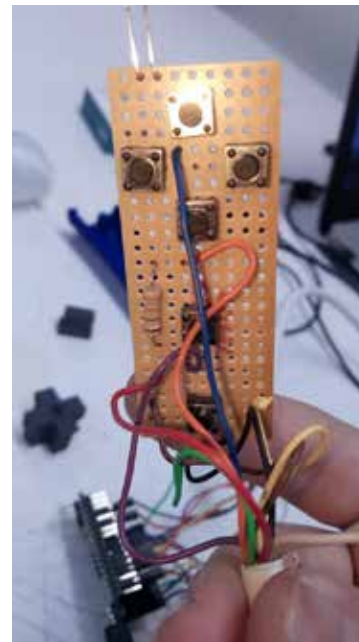
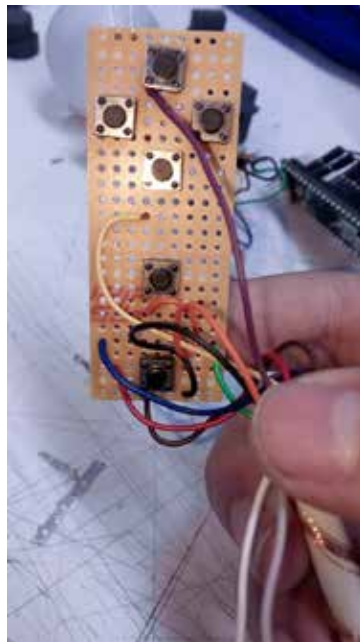


Com o design do comando fechado nós tivemos a montá-los sem o arduino e sem botões para ver se os buracos dos botões estavam bem ajustados e se havia espaço suficiente entre os mesmos e os botões eletrónicos. Depois de limarmos algumas imperfeições os comandos ficaram prontos para receberem as veroboards.

MONTAGEM



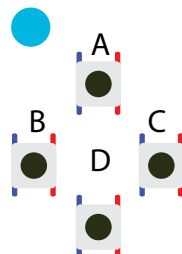
Depois do comando passámos aos cabos que iriam conectar os botões do interface ao arduino. Primeiro arranjámos um cabo que tivesse pelo menos seis cabos de ligação. Acabámos por arranjar um cabo com dez. Com um cabo apropriado começámos a prepará-lo para a soldagem. Primeiro descarnámos os cabos e colocámos cabeças de metal que permitem a eletricidade passar do arduino para o cabo e ativar o botão. Metemos também uma segunda cabeça, esta serve para que não haja interferências entre os outros botões.



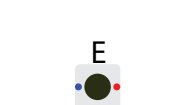
Com os cabos preparados fizemos a soldagem das pontas opostas aos botões da veroboard. A Veroboard da esquerda pertence ao comando azul e a da direita pertence ao comando vermelho. A última tem uma resistência porque tínhamos dois leds e apenas uma entrada de 3 volts. Logo utilizámos uma resistência na entrada de 5 volts, isto para não danificar o led do comando vermelho.

MONTAGEM

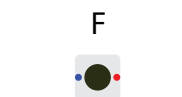
O circuito de ambas as veroboards são exatamente iguais tirando a cor dos cabos de cada botão e a adição da resistência no comando vermelho. No entanto ambos os circuitos estão feitos para que todos os botões tenham um ground comum que neste caso é o cabo preto e ambos têm o cabo vermelho a fazer ligação com o led.



A - Seta de cima; serve para andar para cima e escolher a opção de cima; está ligado ao cabo rocho na entrada 12



B - Seta da esquerda; serve para andar para a esquerda e escolher a opção da esquerda; está ligado ao cabo azul na entrada 11



C - Seta da direita; serve para andar para a direita e escolher a opção da direita; está ligado ao cabo verde na entrada 10



D - Seta de baixo; serve para andar para baixo e escolher a opção de baixo; está ligado ao cabo amarelo na entrada 9

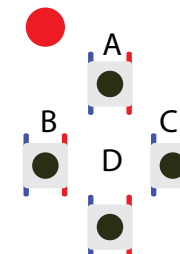


E - Tecla de selecionar; ligado ao cabo laranja na entrada 2



F - Tecla de pausa e retroceder; ligado ao cabo castanho na entrada 4

O led está ligado á porta dos 3V pelo cabo vermelho



A - Seta de cima; serve para andar para cima e escolher a opção de cima; está ligado ao cabo rocho na entrada A3



B - Seta da esquerda; serve para andar para a esquerda e escolher a opção da esquerda; está ligado ao cabo azul na entrada A2



C - Seta da direita; serve para andar para a direita e escolher a opção da direita; está ligado ao cabo verde na entrada A1



D - Seta de baixo; serve para andar para baixo e escolher a opção de baixo; está ligado ao cabo amarelo na entrada A0



E - Resistência; ligada ao cabo do led ou seja ao cabo vermelho 5V



F - Tecla de selecionar; ligado ao cabo laranja na entrada A4

G - Tecla de pausa e retroceder; ligado ao cabo castanho na entrada A5

CÓDIGO DO ARDUINO

```
#include <Keyboard.h>

const int btnEnter = 2;
const int btnSpace = A4;
const int btnEsc = 4;
const int btnP = A5;
const int btnDownP1 = 9;
const int btnRightP1 = 10;
const int btnLeftP1 = 11;
const int btnUpP1 = 12;
const int btnDownP2 = A0;
const int btnRightP2 = A1;
const int btnLeftP2 = A2;
const int btnUpP2 = A3;

char leftKey = KEY_LEFT_ARROW;
char rightKey = KEY_RIGHT_ARROW;
char upKey = KEY_UP_ARROW;
char downKey = KEY_DOWN_ARROW;
char enter = KEY_RETURN;
char escape = KEY_ESC;

void setup() {
  // put your setup code here, to run once
  pinMode(btnEnter, INPUT_PULLUP);
  pinMode(btnSpace, INPUT_PULLUP);
  pinMode(btnEsc, INPUT_PULLUP);
  pinMode(btnP, INPUT_PULLUP);
  pinMode(btnDownP2, INPUT_PULLUP);
  pinMode(btnRightP2, INPUT_PULLUP);
  pinMode(btnLeftP2, INPUT_PULLUP);
  pinMode(btnUpP2, INPUT_PULLUP);
  pinMode(btnDownP1, INPUT_PULLUP);
  pinMode(btnRightP1, INPUT_PULLUP);
  pinMode(btnLeftP1, INPUT_PULLUP);
  pinMode(btnUpP1, INPUT_PULLUP);

  Serial.begin(9600);
  Keyboard.begin();
}
```

Código para atribuir cada botão do arduino a uma tecla do computador. E começa a simulação do teclado.

Como o código está feito, fazemos com que cada botão simule as teclas do computador. Logo para estabelecer uma ligação entre o arduino e o Unity só precisamos de indicar as teclas que vamos utilizar no interface criando assim a ligação entre os dois.

```
void loop() {
  // put your main code here, to run repeatedly:

  if (digitalRead(btnEnter) == LOW) {
    Keyboard.press(enter);
    Keyboard.release(enter);
    delay(250);
  }

  if (digitalRead(btnSpace) == LOW) {
    Keyboard.press(' ');
    Keyboard.release(' ');
    delay(250);
  }

  if (digitalRead(btnEsc) == LOW) {
    Keyboard.press(escape);
    Keyboard.release(escape);
    delay(250);
  }

  if (digitalRead(btnP) == LOW) {
    Keyboard.press('p');
    Keyboard.release('p');
    delay(250);
  }
}
```

Código para detetar quando se pressiona o botão no arduino e converter esse input numa tecla do computador.

```
if (digitalRead(btnUpP1) == LOW) {
  Keyboard.press('w');
  Keyboard.release('w');
  delay(10);
}

if (digitalRead(btnLeftP1) == LOW) {
  Keyboard.press('a');
  Keyboard.release('a');
  delay(10);
}

if (digitalRead(btnRightP1) == LOW) {
  Keyboard.press('d');
  Keyboard.release('d');
  delay(10);
}

if (digitalRead(btnDownP1) == LOW) {
  Keyboard.press('s');
  Keyboard.release('s');
  delay(10);
}

if (digitalRead(btnDownP2) == LOW) {
  Keyboard.press(downKey);
  Keyboard.release(downKey);
  delay(10);
}

if (digitalRead(btnUpP2) == LOW) {
  Keyboard.press(upKey);
  Keyboard.release(upKey);
  delay(10);
}

if (digitalRead(btnLeftP2) == LOW) {
  Keyboard.press(leftKey);
  Keyboard.release(leftKey);
  delay(10);
}

if (digitalRead(btnRightP2) == LOW) {
  Keyboard.press(rightKey);
  Keyboard.release(rightKey);
  delay(10);
}
}
```