

Software e linguagem de programação Arduino

O *software Arduino* tem um ambiente gráfico muito semelhante ao do *Processing*, como podemos ver na Figura 1. É constituído pela Toolbar, Tab Menu e Menus (File, Edit, Sketch, Tools e Help).

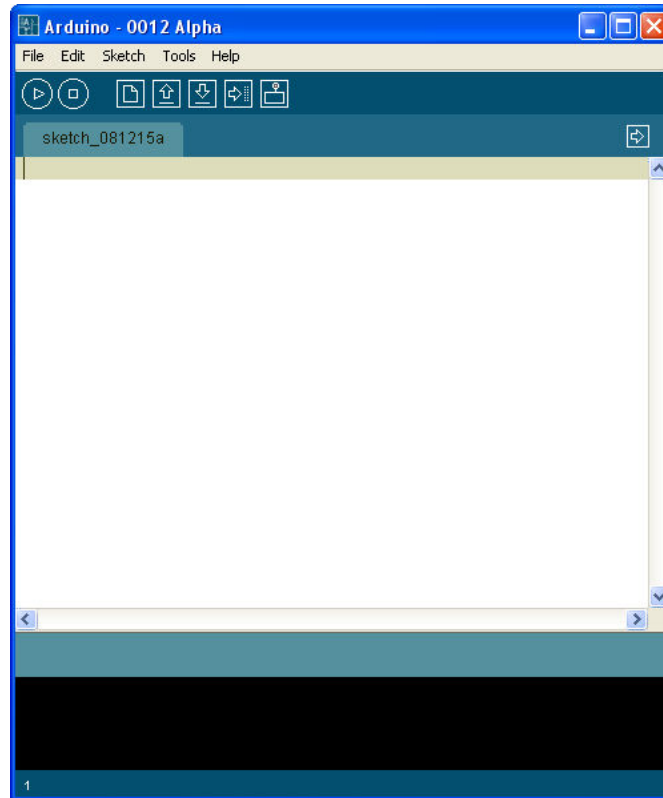









Figura 1 – Ambiente gráfico do Arduino.


A Toolbar é composta por vários botões com funções distintas (Tabela 1)

Tabela 1 – Comandos da Toolbar e suas funções.¹

Verify/Compile 	Verifica se o código tem erros
Stop 	Pára o serial monitor ou desactiva outros botões
New 	Cria um novo documento
Open	Abre uma lista dos documentos salvos e abre o que

¹ Arduino – Introduction to the Arduino Environment. In: <http://arduino.cc/en/Guide/Environment> (2008-12-15; 3h);

	for seleccionado
Save 	Salva o documento
Upload to I/O Board 	Faz upload do código para a placa do <i>Arduino</i>
Serial Monitor 	Mostra a informação enviada pela placa do <i>Arduino</i>

O Tab Menu  permite gerir documentos com mais do que um ficheiro, cada um aberto num *tab* independente. Esses ficheiros podem ser ficheiros normais de código *Arduino* (sem extensão), ficheiros C (extensão .c), C++ (.cpp) ou *header files* (.h)²

Os Menus File, Edit e Help são semelhantes em todos os programas, pelo que não vão ser pormenorizados aqui. No que respeita ao menu Sketch, este contém os seguintes comandos:

Tabela 2 – Comandos do Menu Sketch e suas funções.¹

Verify/Compile	Verifica se o código tem erros
Import Library	Adiciona livrarias ao código, o que lhe confere novas possibilidades
Show Sketch Folder	Abre a pasta do ficheiro no Ambiente de Trabalho
Add File...	Adiciona outro ficheiro ao ficheiro inicial. Esse ficheiro aparece numa nova <i>tab</i>

O menu Tools é composto por outras funções também intrínsecas a este programa:

Tabela 3 – Comandos do Menu Tools e suas funções.¹

Auto Format	Formata o código para uma melhor leitura, alinhando as chavetas e indentando o seu conteúdo.
Copy for Discourse	Copia o código para o clipboard, de forma que seja possível colocá-lo num fórum e manter a distinção da sintaxe por cores.
Board	Selecciona a placa que estamos a usar, o que controla a forma como o código é compilado e o modo como é feito o upload, assim como o comportamento dos itens do menu Burn Bootloader

² Para mais informações ver: Arduino – Arduino Build Process. In: <http://arduino.cc/en/Hacking/BuildProcess> (2008-12-15; 2h);

Serial Port	Contém todas as portas série que o computador possui, permitindo escolher a que está a ser usada. Para usar portas <i>USB</i> é necessário instalar um driver ³ . No Mac, a porta série <i>USB</i> tem uma apresentação semelhante a esta: <i>/dev/tty.usbserial-1B1</i> e no Windows a esta: <i>COM1</i> , variando os números finais.
Burn Bootloader	Os itens deste menu permitem gravar um bootloader na placa <i>Arduino</i> com variados programas, mas não é necessário para um uso normal do <i>Arduino</i> .

O código desenvolve-se a partir de funções e variáveis, como em qualquer linguagem de programação, pelo que se apresenta de seguida um código simples comentado de forma que se compreenda o papel de cada linha de código.⁴

```

/* Blinking LED
 * -----
 * turns on and off a light emitting diode(LED) connected to a digital
 * pin, in intervals of 2 seconds. Ideally we use pin 13 on the Arduino
 * board because it has a resistor attached to it, needing only an LED
 */ //Comentário com várias linhas.

int ledPin = 13; //variável ledPin do tipo int com o valor 13,
                correspondente ao pin digital da placa Arduino com o
                mesmo nome.

void setup() //função que é chamada automaticamente da primeira vez
            que o código é lido. Aqui dentro devem definir-se valores
            iniciais de variáveis, pinMode e introduzir livrarias.
            Tem de ser incluída no código.

{
  pinMode(ledPin, OUTPUT); // determina o pin digital representado
                           pela variável ledPin como sendo de output.
}

void loop() //função chamada de forma cíclica na leitura do código. É
           o centro de todo o programa, onde todas as acções são
           definidas e onde todas as outras funções (à excepção da
           setup) têm de ser chamadas para que sejam lidas. Tem de
           ser incluída no código.

{
  digitalWrite(ledPin, HIGH); // envia corrente para o pin e acende o
                              LED.
  delay(1000); // espera 1 segundo.
  digitalWrite(ledPin, LOW); // não envia corrente para o pin e apaga
                              o LED.
  delay(1000); // espera 1 segundo.
}

```

³ FTDI Chip – VCP Drivers. In: <http://www.ftdichip.com/Drivers/VCP.htm> (2008-12-15; 3h);

⁴ Para mais informações sobre o código, consultar: Arduino – Language Reference. In: <http://arduino.cc/en/Reference/HomePage> (2008-12-15; 3h);