

Olá! Muito obrigada por adquirir a Gedeainha, uma placa eletrônica programável para desenvolver circuitos vestíveis. Neste manual vamos aprender os primeiros passos para utilização da placa, com códigos baseados na plataforma Arduino.

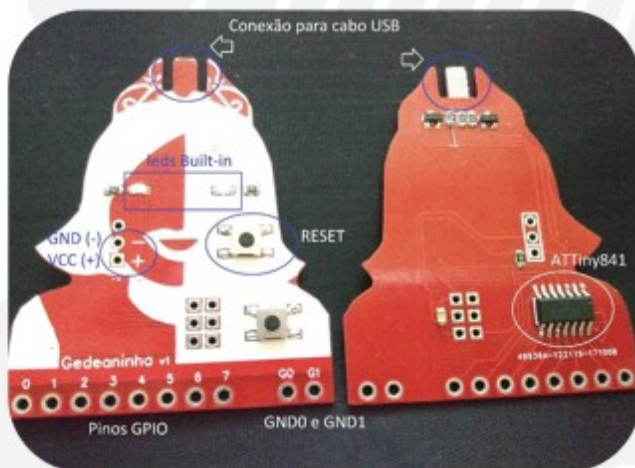


1. Propósito

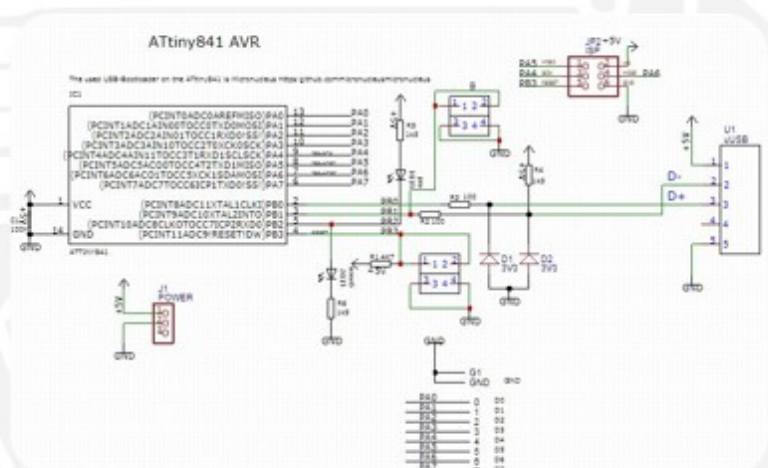
A Gedeainha nasceu no fim de 2017, através da iniciativa de Octavio Nogueira, da Tato Equipamentos. Ele já desenvolveu vários projetos, dentre eles a Tatuino. Expliquei a necessidade de uma versão Arduino para wearable que fosse fácil de encontrar e de preço acessível. Foi criada para ser uma alternativa à Lilypad Arduino, que possui diversos desafios na utilização, como por exemplo a falta de uma conexão USB nativa e a necessidade de conversor serial/USB.

2. Características

A Gedeainha foi baseada em meu logotipo, possui cerca de 52 mm de altura, 44 mm de base e 1 mm de espessura. Trata-se de uma plataforma com microcontrolador Attiny841, uma alternativa ao consagrado Attiny85 com mais pinos. Possui 8 pinos disponíveis na placa mais um para os LEDs dos 'olhos' (pino 8) e um para um botão (pino 9). Assim como o Arduino Gemma e Franzininho, a Gedeainha já possui entrada USB, porém seu maior diferencial é ter um micro USB na coroa (parte superior), onde você pode plugar o cabo micro USB de um celular Android.



3. Esquemático

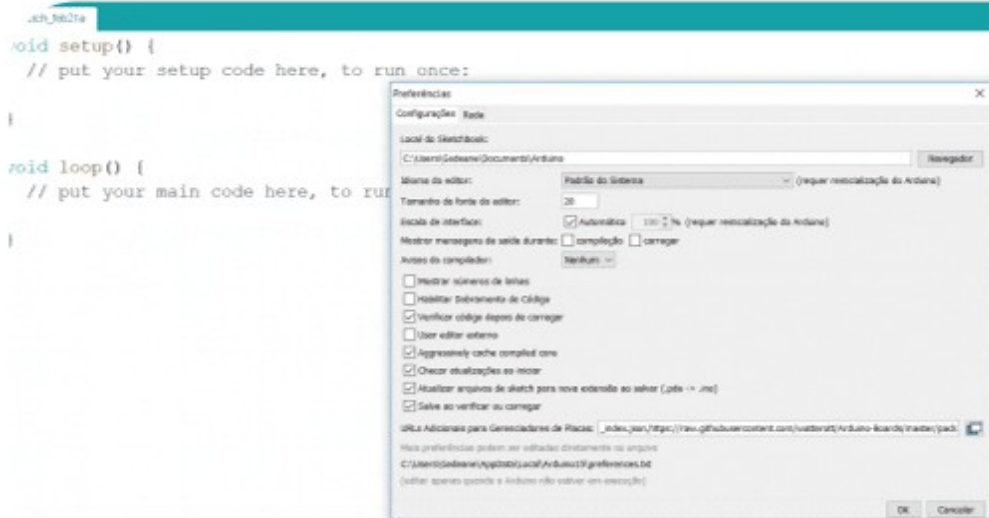


4. Como gravar sketches

Utilizei a versão 1.8.5 da IDE Arduino, porém, se a versão permitir a instalação de placas (como ESP8266, Adafruit, etc) já é possível a utilização.

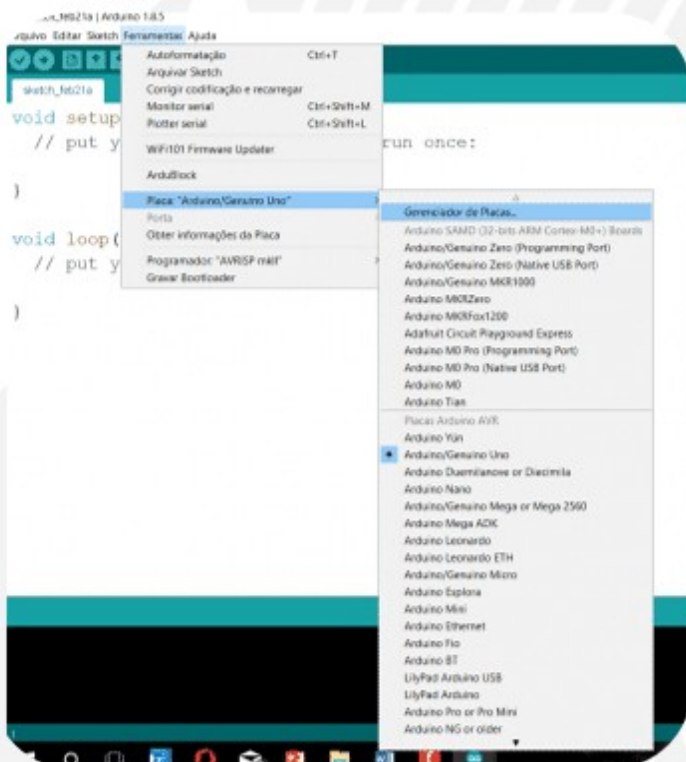
No menu Arquivo => Preferências... copie e cole o seguinte endereço JSON em URLs Adicionais para Gerenciadores de placas.

https://raw.githubusercontent.com/watterott/Arduino-Boards/master/package_watterott_index.json.



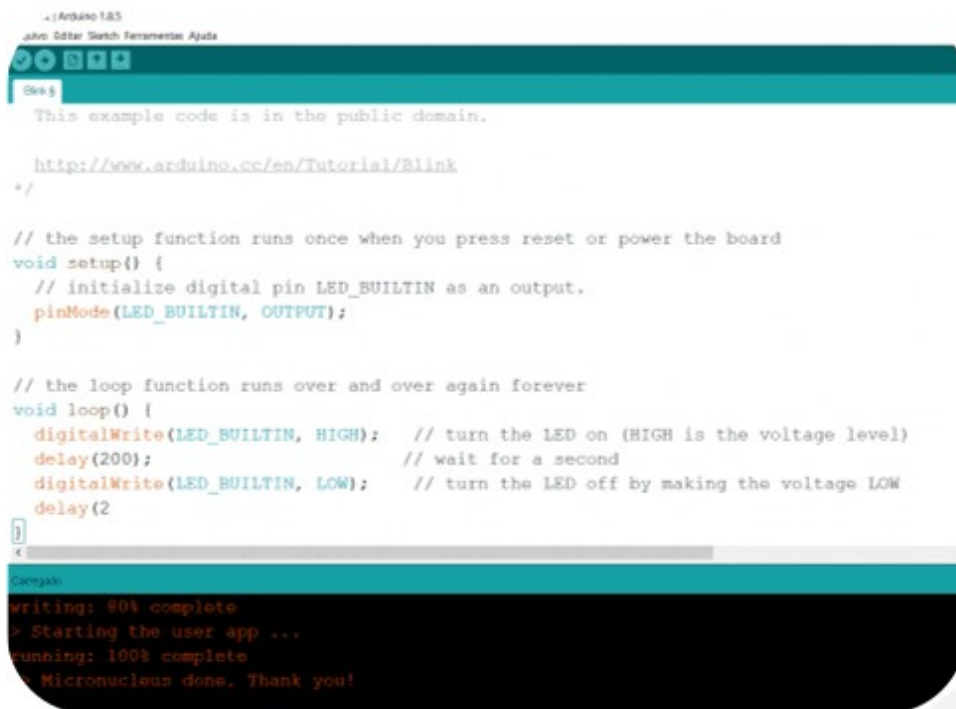
Caso já tenha instalado outras placas por este método, separe os links por vírgulas. Clique em OK para fechar a janela.

Em Ferramentas => Placas => Gerenciador de placas, escolha a opção Watterout AVR Boards e clique em Instalar.



Quando terminar, aparecerá a palavra INSTALLED em letras verdes no título, conforme a figura. Clique em OK.

Aperte o botão RESET (botão próximo aos 'olhos' da placa) para ativar o modo de bootloader. Se tudo ocorrer bem, aparecerá no rodapé a mensagem conforme a imagem abaixo:



The screenshot shows the Arduino IDE interface. At the top, there's a menu bar with 'Arquivo', 'Editar', 'Sketch', 'Ferramentas', and 'Ajuda'. Below it is a toolbar with icons for opening files, saving, compiling, and uploading. The main text area contains the following code:

```
// Arduino 1.8.5
// Este código está em domínio público.

// http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(200);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(200);
}
```

At the bottom, the status bar shows the following messages:

```
Carregando...
writing: 90% complete
> Starting the user app ...
uploading: 100% complete
Micronucleus done. Thank you!
```

5. Dicas

- A IDE no Windows (no meu caso) demora um certo tempo para compilar o arquivo. Preste atenção à barra de status sendo preenchida em verde à direita, próxima ao rodapé. Quando ver a palavra CARREGANDO aperte o botão RESET para ativar o modo bootloader;
- Nas versões MAC e Linux este carregamento é quase simultâneo à compilação. Ao clicar em UPLOAD, já aperte o botão RESET. Este procedimento deve ser rápido, pois pode gerar erros na hora de gravar.
- O LED_BUILTIN está no pino 8. O LED vermelho (esquerdo) acende em LOW e o LED verde (direito) acende em HIGH. Só acendem simultaneamente quando está em modo de gravação (ao apertar RESET).
- Como a versão 1.2 não possui pino específico de VCC, use um dos pinos disponíveis via código no void setup: : **digitalWrite** (2,HIGH);