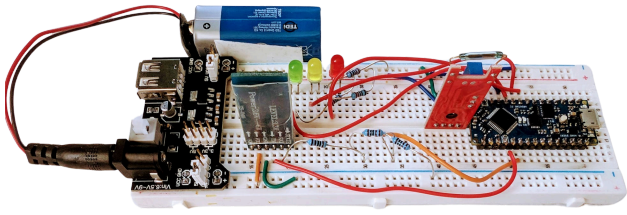


Do it yourself Arduino EMF reader



Required components

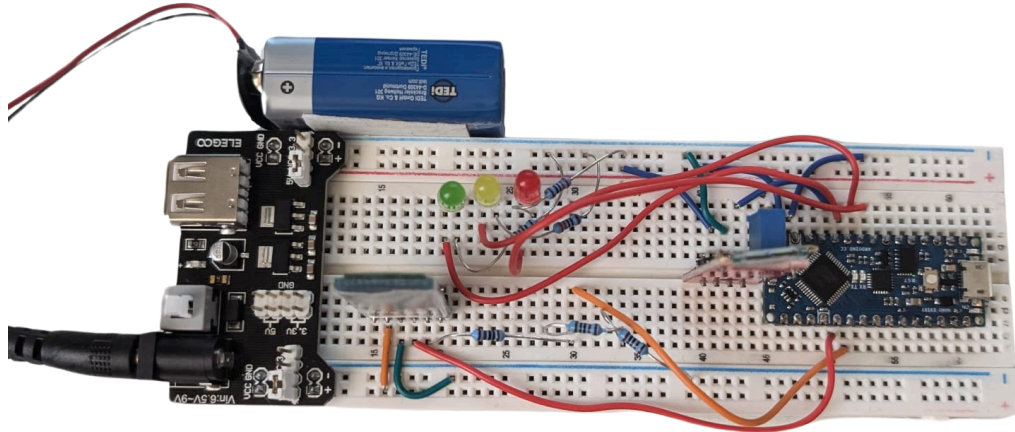
- [Any Arduino \(I personally would recommend an Arduino Nano Every\).](#)
- [A Bluetooth HC-05 module](#)
- A green LED
- A yellow LED
- A red LED
- 3 x 220-ohm resistors
- 3 x 1k ohm resistors
- [Power supply module](#)
- Battery
- [Battery to DC 9V converter](#)
- 12 male jumpers
- [Breadboard](#)
- [KY-025](#)

How to build

- Connect the power supply to the breadboard
- Connect the Bluetooth module
 - The Vcc pin has to go into the Arduino/Power Supply 5v
 - The GND pin has to go into the Arduino/Power Supply ground
 - Connect TXD to digital pin 2
 - Connect RXD
 - Connect one resistor of 1k ohm to a jumper going to Arduino digital pin 3
 - Connect two resistors together of 1k ohm to the Arduino/Power Supply ground
- Connect the KY-025
 - The + pin to the 5v Arduino/Power Supply
 - The G pin to the Arduino/Power Supply ground
 - The A0 pin to your Arduino A7 pin
- Connect the LEDs
 - Connect all the LEDs ground (small leg) to Arduino/Power Supply ground using a 220 ohm resistor
 - Green LED
 - Connect the long leg to Arduino pin number 20
 - Yellow LED

- Connect the long leg to Arduino pin number 19
- Red LED
- Connect the long leg to Arduino pin number 18
- Connect the battery to the power supply WITHOUT TURNING THE POWER SUPPLY ON
- Connect the Power Supply 5v to Arduino Vin
- Connect the Power Supply ground to Arduino gnd

How the circuit should look like



Customize Bluetooth settings

- [Step 1 Upload this script](#)
- Step 2 Disconnect your Arduino
- Step 3 hold the button on the front side of hc-05 while giving power to Arduino until the main power led of the module does not emit light
- Step 4 Upload it to your Arduino and open a serial monitor at the baud rate of 38400
- Step 5 customize your Bluetooth settings using [this table](#) Remember to use the command AT+PSWD to get your module password and AT+NAME to get your module name.
- Step 6 Remove the power from the sensor by shutting down Arduino
- Step 7 Upload [this sketch](#) to Arduino

How to connect to Bluetooth

Once you set up all your settings following steps 3-5, make sure to power the circuit via battery or via cable. Then, using the name and password you discovered earlier, connect to your Bluetooth device and install [this app](#).

From there, you can press the Menu button on the top left of your screen, go into devices -> Bluetooth Classic, and select your device (the HC-05 module).

1. This will connect you to the Bluetooth of the Arduino, where you can perform the following commands.
2. **Ping:** Sends a simple message to the Arduino, typically used to check the connection status.
 - **Command:** Ping

- **Function:** Verifies that the Bluetooth connection to the Arduino is active by sending a straightforward message.
- 3. **SetPin:** Allows you to set the value of a specific Arduino pin, enabling control over the pin's state.
 - **Command:** SetPin
 - **Function:** Sets the digital or analog value of an Arduino pin as specified.
- 4. **PauseAll:** Stops the entire program execution on the Arduino.
 - **Command:** PauseAll
 - **Function:** Halts all operations running on the Arduino until a resume command is received.
- 5. **PauseNormal:** Stops the entire program execution except for the Bluetooth command processing.
 - **Command:** PauseNormal
 - **Function:** Pauses all operations except for those related to Bluetooth communication, allowing remote control and management to continue.
- 6. **GetEMF:** Returns the Electromotive Force (EMF) reading from the Arduino, where 1024 indicates low EMF and 0 indicates high EMF.
 - **Command:** GetEMF
 - **Function:** Provides the current EMF reading from the Arduino's sensors.
- 7. **autoSendEMF:** Toggles automatic sending of EMF data approximately every 100 milliseconds. This command can be paused.
 - **Command:** autoSendEMF
 - **Function:** Automatically transmits EMF readings at regular intervals, which can be paused with relevant commands.