



A Weird Arduino Powered Clock © Apache-2.0

I needed a weird clock to go along with the rest of the weird projects in my workshop.

clock creator

8 , 9 0 8 8 W C S O M M 2 E 6 N E S P E C T S

COMPONENTS AND SUPPLIES



Arduino Micro & Genuino Micro

Any microcontroller will do.

×

1



Stepper Motor 28BYJ-48

Any stepper motor will do.

×

1



ULN2003 Driver Board

Any stepper motor controller will do.

×

1



NECESSARY TOOLS AND MACHINES

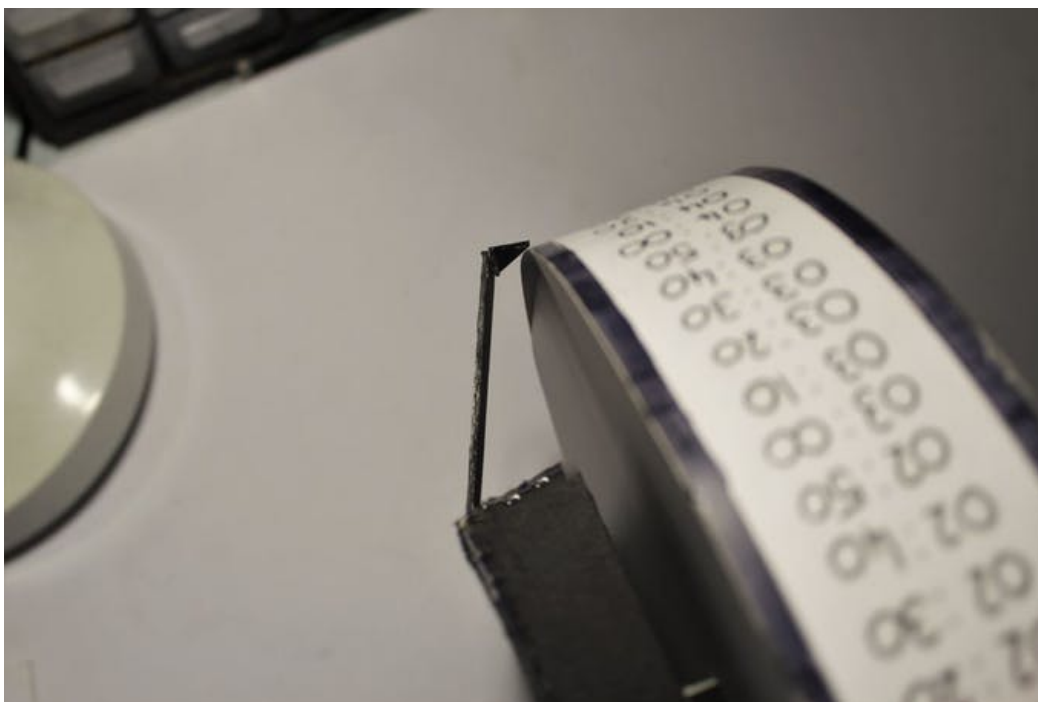


Soldering iron (generic)



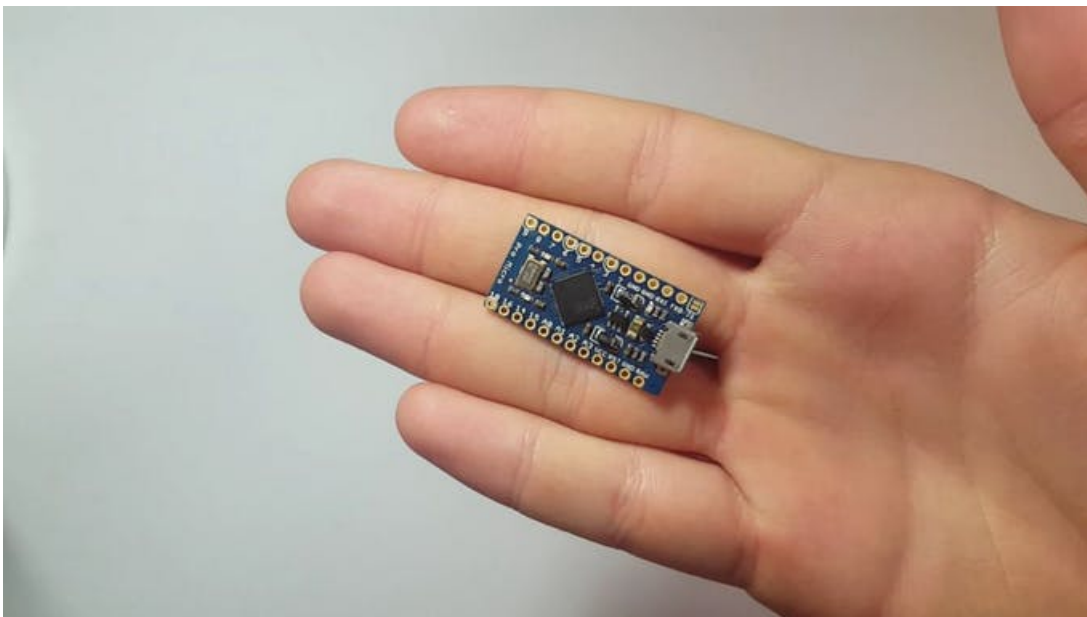
Hot glue gun (generic)

ABOUT THIS PROJECT



Hey, everyone. Today I'm going to show you how to build a very different looking clock, powered by an Arduino. The idea behind the clock is a circle with a circumference of 72 cm (28.3465 inches) that ticks at 1 cm every ten minutes which means every 72 ticks will equal 12 hours making it a 12 hour clock. (It's pretty complicated to explain so check the pictures and video for a better understanding.)

Step 1: Parts List



• • • • •

For this project you will need the following parts:

- A Microcontroller (I'm using an Arduino Micro but any will do)
- A StepperMotor (I'm using the 28BYJ-48 found here: <https://goo.gl/YSw79q>)
- A StepperMotor Controller (I'm using the paired board but any will do: <https://goo.gl/YSw79q>)
- Cardboard (thick and thin)

And these parts:

- A Soldering Iron
- A Glue Gun

- A Box Cutter (knife)

Step 2: The Stepper Motor Library

Name	Date modified	Type	Size
drivers	2016/01/07 5:20 PM	File folder	
examples	2016/01/07 5:20 PM	File folder	
hardware	2016/01/07 5:20 PM	File folder	
java	2016/01/07 5:20 PM	File folder	
lib	2016/01/07 5:20 PM	File folder	
libraries	2016/07/01 9:10 PM	File folder	
reference	2016/01/07 5:20 PM	File folder	
tools	2016/01/07 5:20 PM	File folder	
arduino	2014/01/08 9:46 PM	Application	840 KB
cygconv-2.dll	2014/01/08 9:45 PM	Application extens...	947 KB
cygwin1.dll	2014/01/08 9:45 PM	Application extens...	1 829 KB
libusb0.dll	2014/01/08 9:45 PM	Application extens...	43 KB
revisions	2014/01/08 9:45 PM	Text Document	38 KB
rxTxSerial.dll	2014/01/08 9:45 PM	Application extens...	76 KB
uninstall	2016/01/07 5:21 PM	Application	402 KB

• • •

Disclaimer. You only need to install this library if you are using the less common ULN2003 MotorController. Most other motor controllers will work with the pre-installed Arduino motor controller library. If you are using these skip this step.

Everything I explain here can be found on this website: (<http://goo.gl/Da5jEh>)

- Start by downloading the .cpp and .h file I have added
- Create a new file called StepperMotor
- Drag and drop both the .cpp and .h file into the new StepperMotor file
- Go to your Arduino libraries file and drop the StepperMotor file into it
- Open Arduino IDE and run this test code to make sure everything was installed correctly:

```
#include StepperMotor motor(8,9,10,11);
void setup(){
  Serial.begin(9600);
  motor.setStepDuration(1);
}
void loop(){
  motor.step(1000);
  delay(2000);
  motor.step(-1000);
  delay(2000);
}
```

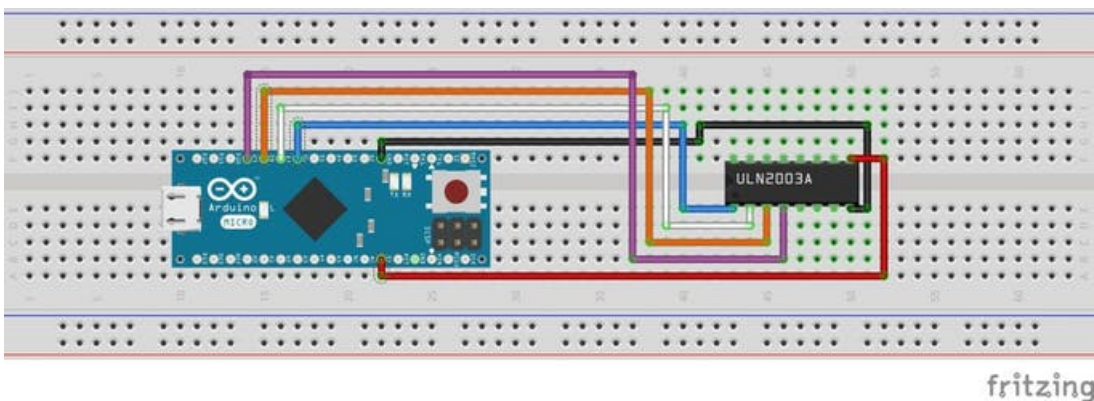
- Now verify the code in the Arduino IDE. If it compiles without any issues, you have installed everything

correctly.

 F G F Q X I N I Q 2 O 4 O R B . C P P

 F F V K D R E I Q 2 O 4 O R G . H

Step 3: The Wiring



First things first, plug your stepper motor into your stepper motor controller and get your soldering iron ready. Connect the Motor Controller to the Arduino in the this order:

- Connect Ground on the Arduino to Ground on the motor controller
- Connect 5v on the Arduino to 5v on the motor controller
- Connect Pin 6 on the Arduino (Blue) to int 1 on the motor controller
- Connect pin 7 on the Arduino (White) to int 2 on the motor controller
- Connect pin 8 on the Arduino (Orange) to int 3 on the motor controller
- Connect pin 9 on the Arduino (Purple) to int 4 on the motor controller

Check the colour coded pictures to double check your wiring.

Step 4: The Code

```
file edit sketch tools help
StepperMotorclock
#include <StepperMotor.h>

StepperMotor motor(6,7,8,9);

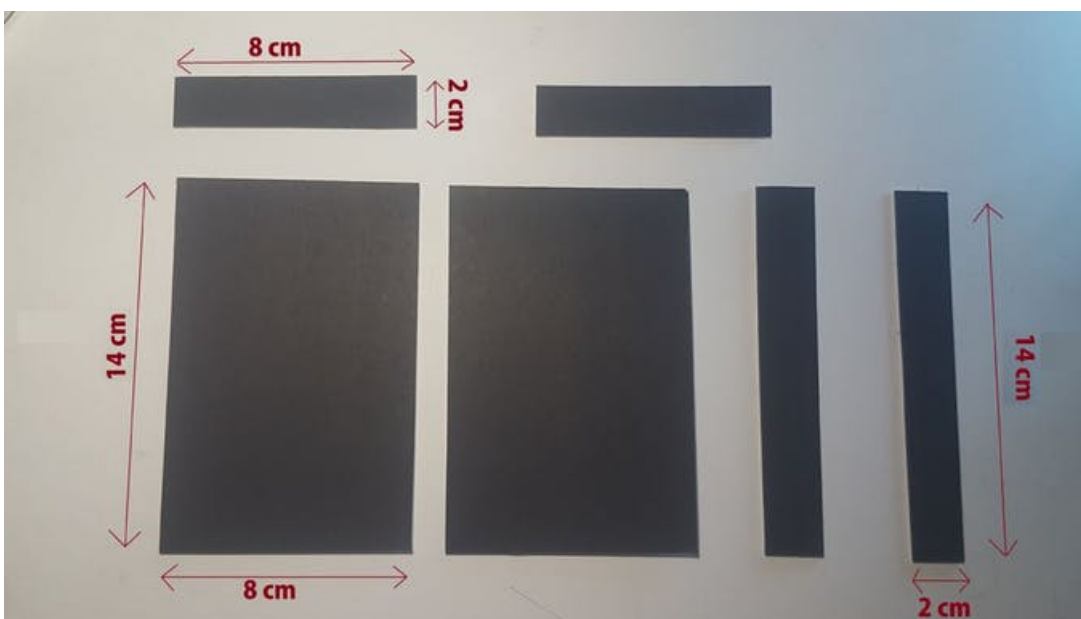
void setup() {
  Serial.begin(9600);
  motor.setStepDuration(1);
}

void loop() {
  motor.step(-64);
  delay(600000);
}
```

Copy the code below, paste it into your Arduino IDE and upload it to your Arduino.

```
#includeStepperMotor motor(6,7,8,9);void setup(){
  Serial.begin(9600);
  motor.setStepDuration(1);
}void loop(){
  motor.step(-64);
  delay(600000);
}
```

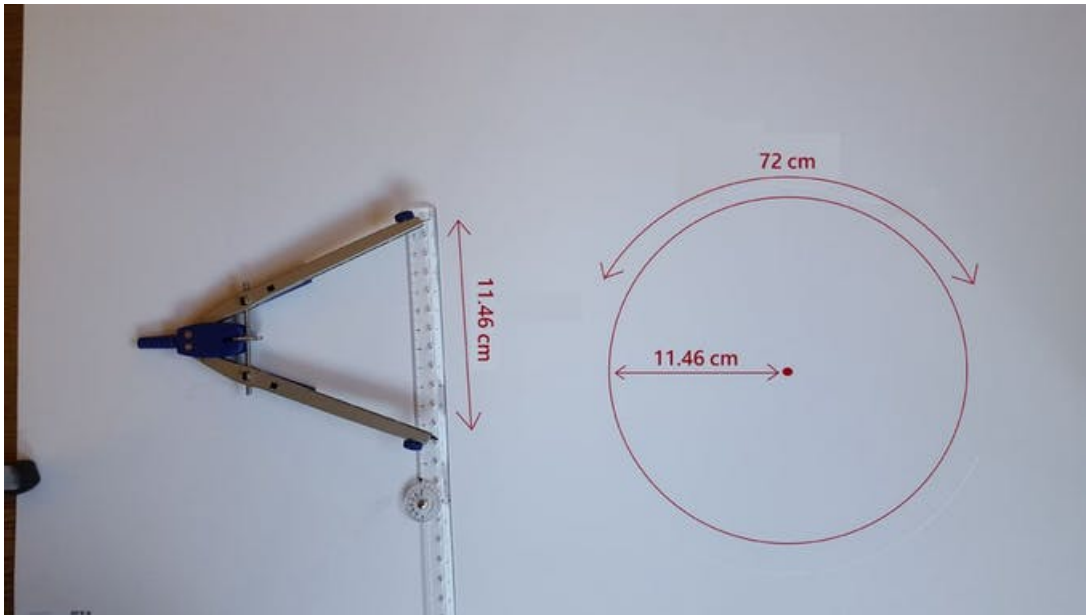
Step 5: The Electronics Casing



To make the casing, I'm going to start by creating a box around the electronics that has a small hole allowing the motor shaft to stick out allowing it to make contact with the moving part of the clock. To

make it easier to put the clock face on, we are going to make a cardboard square and glue it onto the motor shaft - see photos.

Step 6: Creating the Clock Face



To start off, we need to figure out the basics of the clock face. Normal clocks have 3 hands around a 12-hour face allowing the user to see seconds, minutes and hours. Now, with our clock, we only have 1 "hand" which means we have to maximize our space. My final idea was to create the circle with a circumference of 72 cm (not too big and not too small) which means each "tick" will move the circle by 1 cm; therefore meaning that 1 cm equals 10 minutes (720 minutes in 12 hours). All of this is a little confusing but the pictures do help with the understanding.

- Start by getting a compass (drawing tool) and opening it to 11.46 cm, which is the radius of the circle.
- Draw two circles and cut them out.
- Now grab a thin piece of cardboard and cut a strip measuring 72 cm in length. The strip should be 5 cm wide but its not very important how wide it is.
- On the strip, measure and mark every cm.
- Draw time intervals on each mark (like this 12:00, 12:10, 12:20, 12:30, 12:40, 12:50, 01:00, 01:10... until you get to 11:50 which would be at the end of the sheet).
- Cut square holes in the middle of both circles.
- Glue the strip we cut out in between the two circles to make a mini-cylinder.

Step 7: Putting It All Together



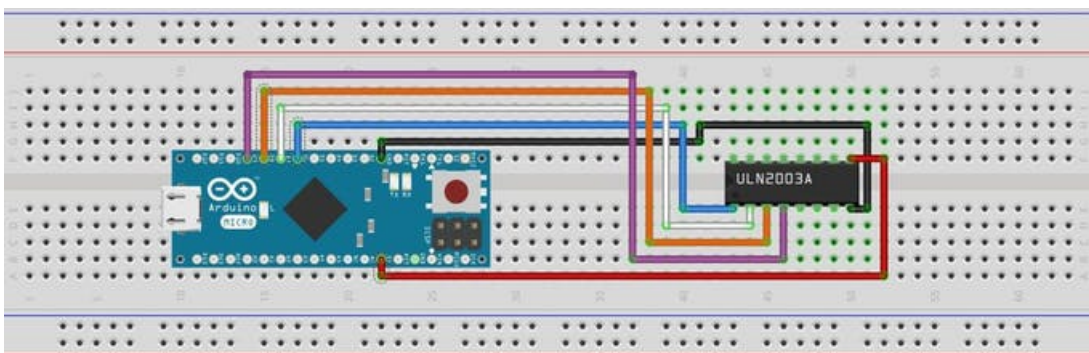
• • • •

Put the peaking out part of the motor (that turns) into the square we made in the circles and glue it together (see pictures). Once it's all together, we can test it out and make sure it works and ticks in time. If you have any questions, please message me or leave a comment and I'll try my best to get back to you. Thank you so much for reading.

SCHEMATICS

Arduino / Motor Controller

DOWNLOAD

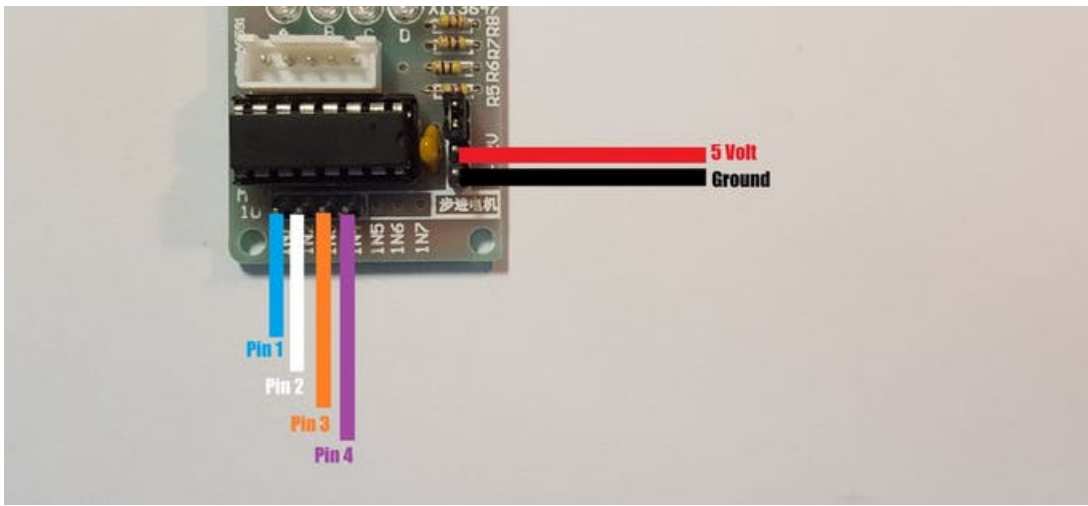


fritzing

f89zec4iq2o4wsa_large_Q5YDUk4353.jpg

DOWNLOAD





CODE

Code snippet #1 Plain text



```
1 #includeStepperMotor motor(6,7,8,9);void setup(){
2   Serial.begin(9600);
3   motor.setStepDuration(1);
4 }void loop(){
5   motor.step(-64);
6   delay(600000);
7 }
```

COMMENTS

Please [log in](#) or [sign up](#) to comment.



Davy Roscoe

9 months ago

What a great project especially for the less experienced like myself! It leaves the option to make the case as decorative as you see fit! If your starting out with arduino this project looks ideal for fast practical results.

The circuit could be added to e.g. ticking sound etc I suppose.



Dante Roumega

9 months ago

Adding additional circuits like adding a ticking sounds is a great idea. Thanks for the feed back!

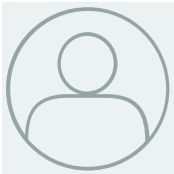


Amartya

4 months ago

Nice Project!

AUTHOR



Dante Roumega

5 PROJECTS 20 FOLLOWERS

FOLLOW

PUBLISHED ON

February 21, 2017

RESPECT PROJECT

GIVE FEEDBACK

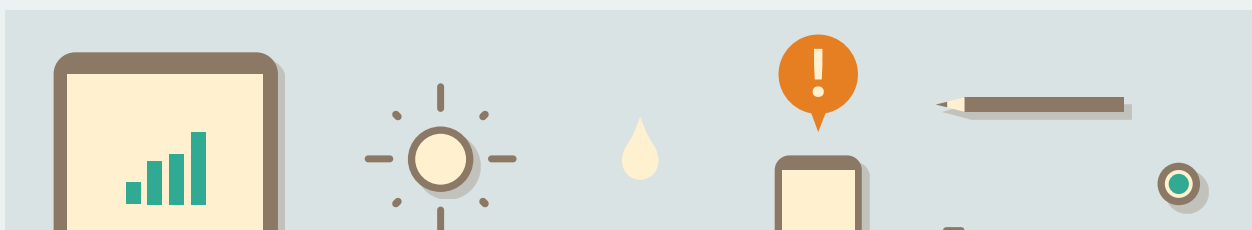
Share I made one

MEMBERS WHO RESPECT THIS PROJECT



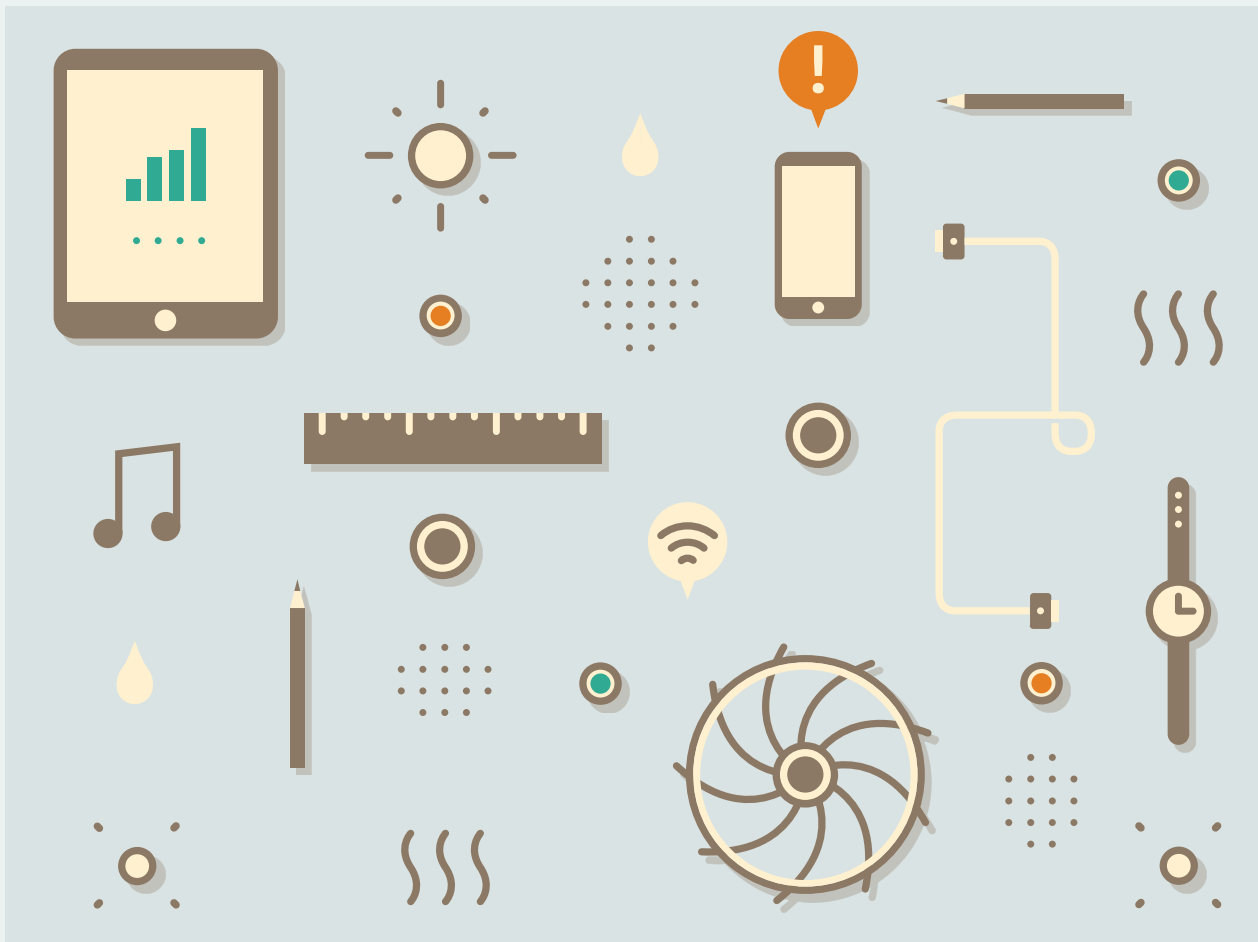
and 18 others

SIMILAR PROJECTS YOU MIGHT LIKE



Project tutorial by **gadget programmers**

4,174 VIEWS 13 COMMENTS 21 RESPECTS



Project tutorial by **gadget programmers**

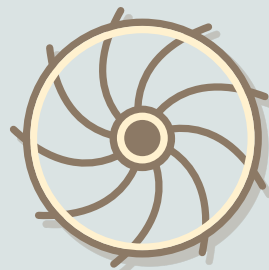
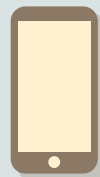
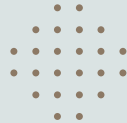
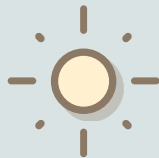
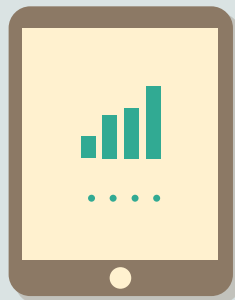
1,250 VIEWS 2 COMMENTS 5 RESPECTS





Project tutorial by **pittex**

331 VIEWS **0** COMMENTS **3** RESPECTS



Project tutorial by **javier Muñoz Sáez**

2,486 VIEWS **3** COMMENTS **4** RESPECTS

