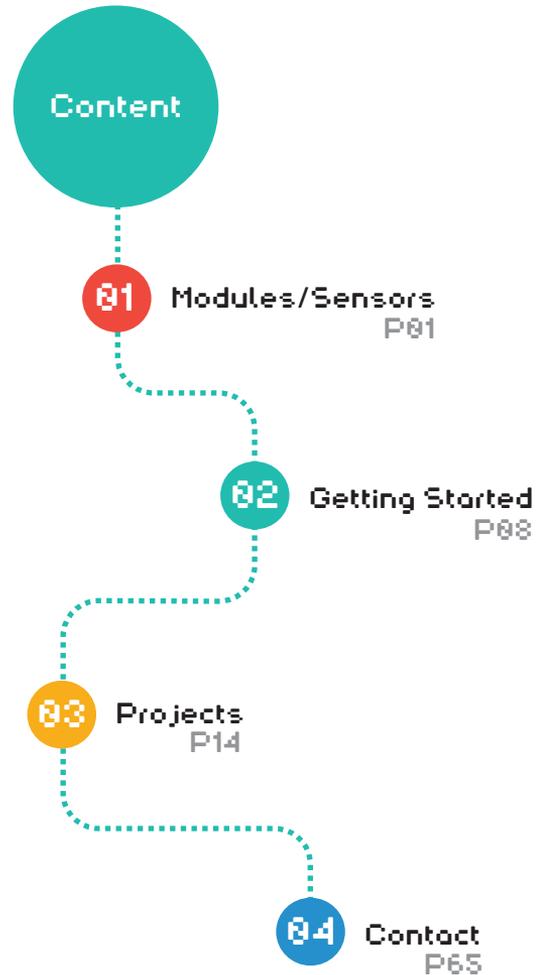




CREATIVITY IS **CONTAGIOUS**  
**PASS IT ON!**



# Modules/Sensors



## m cookie

**mCookie** is Microduino's second generation smart modules. Using the all new implementation of magnetic connectors and pogo pins, they are now much easier to use. The colorful modules can be paired with LEGO to create countless projects with ease.

With the mCookie, **Everyone is an Inventor!**



### Core

Red modules have to do with core functionalities such as uploading programs and controlling other modules. They are the brain of all projects.



### Function

Yellow modules are the function modules, each with its own particular use. Just stack the corresponding function modules to create what you have in mind.



### Communication

Blue modules are tasked with communicating with other devices, either wired or wirelessly.



### Extension

Green modules are extension boards. They are equipped with different types of connections to accommodate sensors and other devices.



## CoreUSB

The CoreUSB module is the control center for all projects. Each one is equipped with a standard MicroUSB port to transfer programs between the module and your computer.



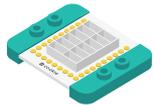
## Battery Management

The Battery Management module has a port to connect the mCookie Battery module to provide energy to your projects.



## Bluetooth

The Bluetooth module provides bluetooth functionality. With a mobile app, project functions can be controlled using your phone.



## Hub

The Hub module offers 12 different connection ports for attaching sensors and other external devices.



## Amplifier

The Amplifier module strengthens audio signals and can then be played by connecting speakers. The Amplifier is often used with the Audio module.



## Audio

The Audio module stores MP3 files and converts them into a playable format. Stack it on top of the Amplifier module to play music through speakers.



## Audio Shield

The Audio Shield module allows users to insert a Micro SD card, greatly increasing the storage capacity for more MP3 files. The Audio Shield is often used with the Audio module.



## Real Time Clock

The Real Time Clock (RTC) module adds a new dimension of time to projects.



## Motion

The Motion module can sense speed, angle as well as direction.



## Motor

The Motor module controls the speed and duration of motors.



## WiFi

The WiFi module connects your project to the Internet, greatly enhancing the possibilities.



## ColorLED

Multicolor LED. Supports 16 million colors and multiple LED connections, bringing vibrancy to your ideas.



## Crash

Using the lever principle, a firm click will establish a connection, effective as a button for many applications.



## Buzzer

A simple noise-making device that can be used as an alarm or reminder. You can also play simple music that has little pitch variation!



## Microphone Sensor

The cylindrical cone is the microphone. This particular sensor is unable to record audio contents, but instead, it measures the loudness of its surroundings.



## Humidity/Temp Sensor

This sensor is able to evaluate the temperature and humidity of its environment. Make sure to leave the black box exposed for accurate readings.



## Passive Infrared Sensor

This sensor detects the presence of heat given off by humans or animals. Make sure to leave the white protrusion exposed for accurate readings.



## Joystick

This allows easier control for projects that require directional maneuverability.



## Light Sensor

This sensor detects the intensity of ambient light. Make sure to leave the sensor exposed for accurate readings.



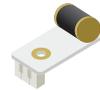
## Infrared Emitter

This device enables remote control over common appliances.



## Infrared Receiver

This device enables you to control projects with available remote controllers.



## Shake Sensor

This sensor detects vibrations.



## Color Sensor

This sensor is able to evaluate colors and return RGB values.



## Motor

Motors offer continuous rotations to bring your projects to life. Both their speed and duration can be programmed.



## Motor Accessory

These accessories are helpful when trying to attach motors onto your applications.



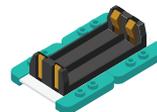
## Servo

A servo differs from a motor in that it can only swivel  $0-180$  degrees. However, it offers greater accuracy.



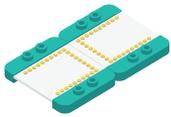
## Servo Connector

This serves as an adapter for connecting servos onto the Hub module. The connector supports up to two servos.



## Battery Module

Supports either two AAA batteries (1.5V) or one/two lithium batteries (3.7V). Use a switch on the bottom to switch between the two battery types.



## Duo-V

Is your stack of modules growing too high? No worries, this extension board allows the height to be halved.



## OLED

The OLED module serves as a simple, single color display.



## USB Cable

Connects the CoreUSB module to a computer. You can then upload programs into the module as well as provide it power.



## Remote Control

If you have the Infrared Receiver connected to your application, you can use this Remote Control to do really cool stuff!



## Sensor Cable

Used to connect various sensors and external devices.



## Speakers

Connect these to the Amplifier module to play sounds.



## LEGO® Connector

Snap these onto the bottom of your modules to easily attach them onto your LEGO !

# Getting Started!

## Install IDE&Driver For **WINDOWS**



### 1 Download the IDE and Drivers

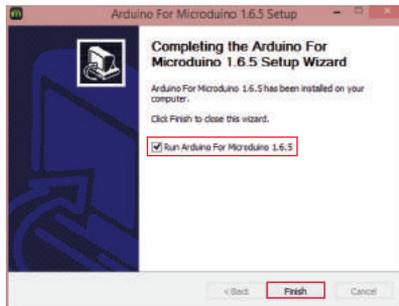
1. Download the Arduino IDE with integrated Microduino libraries for all your modules and sensors from our website:

[www.microduino.cc/download](http://www.microduino.cc/download)

2. Click on Download under Windows and follow the instructions to complete your installation:



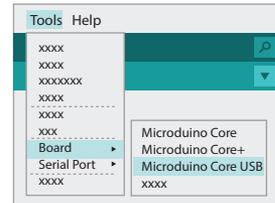
3. Follow the instructions until you see the window below. Click Finish to complete your installation.



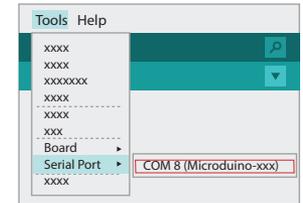
### 2 Verify IDE



1. Connect your CoreUSB module to your PC with the provided USB cable. Then open your Arduino IDE.



2. Select Tools > Board > Microduino CoreUSB.



3. If the drivers installed successfully, you should see COMx options under Tools > Serial Port.

## For **MAC**



### 1 Download the IDE and Driver

1. Select the download for your version of Mac OS from our website:

[www.microduino.cc/download](http://www.microduino.cc/download)

2. Open the downloaded file and you will see the Arduino IDE as well as a folder for drivers.



3. Open the Drivers folder and then install the corresponding drivers for your Mac OS version.



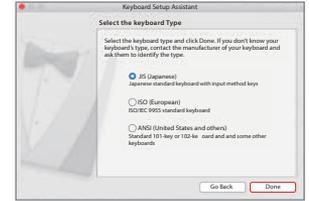
2. Your Mac may falsely detect the CoreUSB as a keyboard. If the above popup appears, press continue.



3. Follow the instructions and press down the key to the right of your shift key.



4. Naturally, your Mac will still be unable to identify the module as a keyboard. Press Skip.

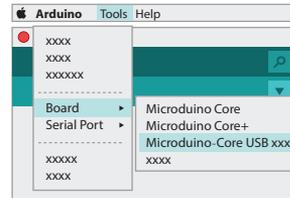


5. Lastly, choose the keyboard layout for your region and click Done. You have successfully installed Microduino drivers!

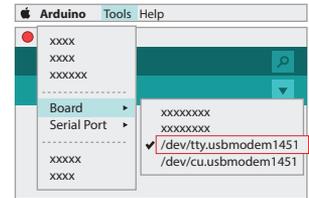
## 2 Verify IDE



1. Connect your CoreUSB module to your MAC with the provided USB cable. Start your Arduino IDE.

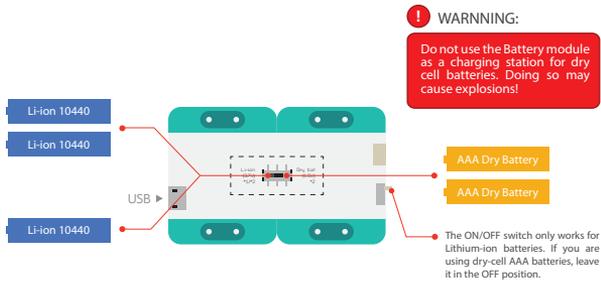
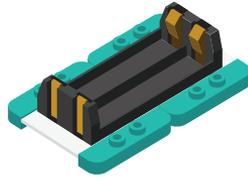


6. Select Tools > Board > Microduino CoreUSB.



7. If the drivers installed successfully, you should see /dev/tty.usbmodem1451 under Tools > Serial Port.

## Activate Battery Module



- The Battery module supports both rechargeable 10440 lithium-ion batteries and standard AAA dry cell batteries. You have the option of using either one or two lithium-ion batteries. Use the switch on the bottom to switch between the type of
- **Note:** Choose your desired mode prior to inserting the batteries. Changing modes while the batteries are inserted may affect operation.



- Prior to using the Battery module, it must first be activated. Use the provided USB cable to connect the module to your computer. After two seconds, the module is ready for project use.
- **Note:** The above step must be completed every time new batteries are inserted.

# Projects

## Try your mCookies

- 1 Grumpy Grandpa ..... P.16
- 2 Cowardly Night Light (BLE) ..... P.20
- 3 Say Cheese! ..... P.26
- 4 Thermo Cup ..... P.30
- 5 mWatch(Wi-Fi) ..... P.36
- 6 Music Box (Remote Control) ..... P.39
- 7 Bashful Chameleon ..... P.45
- 8 Dozing Donkey ..... P.50
- 9 Personal Trainer ..... P.55
- 10 Fortune Cat ..... P.60

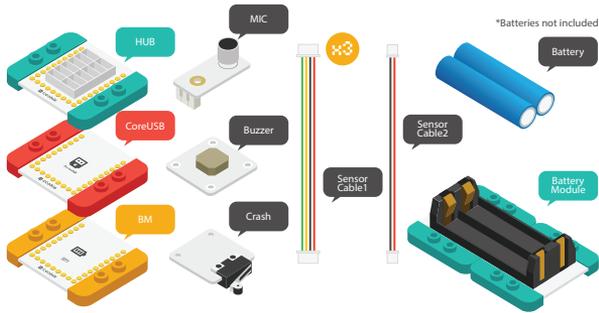


### PROJECT 1

## Grumpy Grandpa

Grumpy Grandpa hates loud noises!  
Whenever it gets too loud, an alarm goes off. Make sure to press a  
button if you want your own peace and quiet!

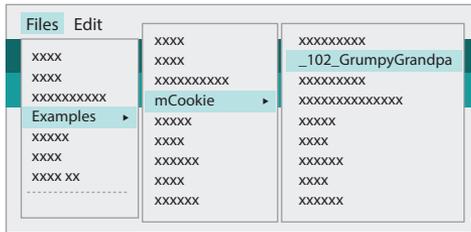
## You will need



### 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

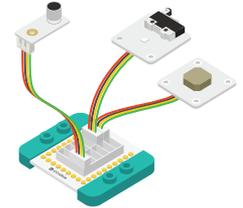
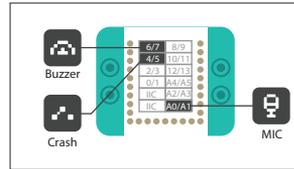


2. Select Files > Examples > mCookie > \_102\_GrumpyGrandpa.

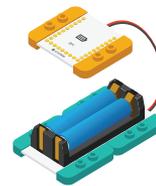


3. Click to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

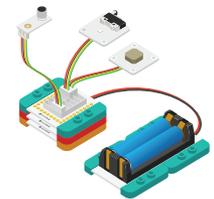
### 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.

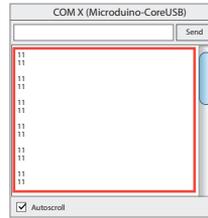


2. Connect the activated Battery module to your Battery Management module.



3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

### 3 Adjust Code

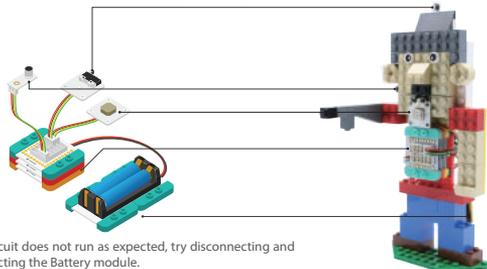
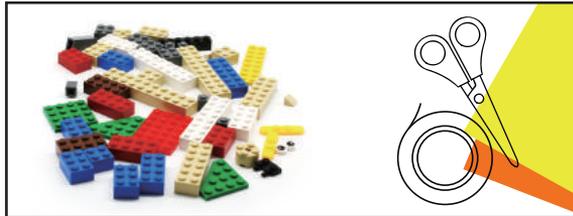


- Click  after uploading the code. A window will appear, displaying the loudness that the microphone sensor is receiving.

```
#include "key.h"  
#define voice 400
```

- In the code you uploaded earlier, find the above line. Currently, "voice" is defined to be 400. You can adjust this value to change at what value the buzzer starts buzzing. Upload your program again for it to take effect.

### 4 Make your LEGO® Grandpa!



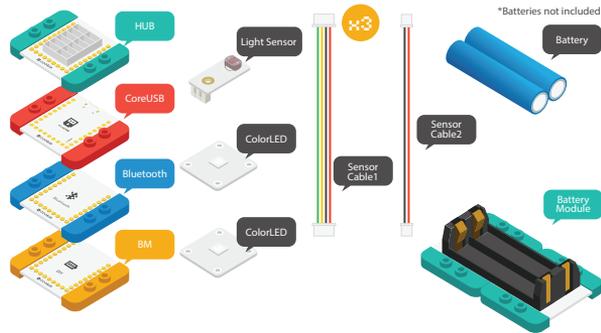
- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



## PROJECT 2 Cowardly Night Light (BLE)

When night falls, Cowardly Night Light turns on because he's afraid of the dark! When he lights up, you can use your phone's bluetooth function to control the brightness and colors.

## You will need



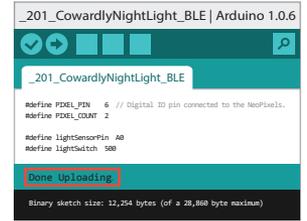
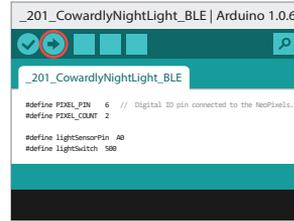
### 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

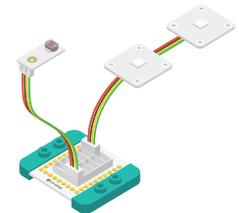
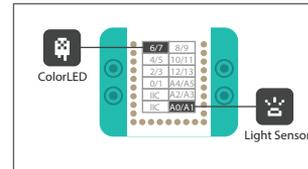


2. Select Files > Examples > mCookie > \_201\_Cowardly NightLight \_BLE.

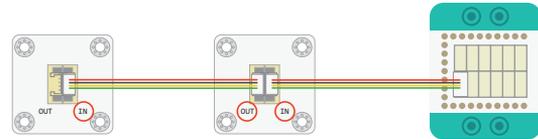


3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

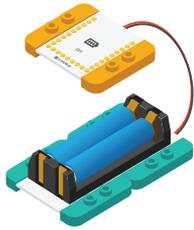
### 2 Build the circuit



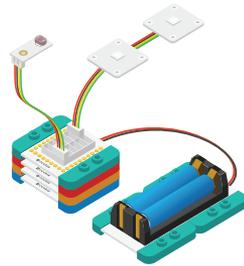
1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



• To connect multiple LEDs in a series, follow the above diagram. Notice that the sensor cable connects to the IN ports of the LEDs. You can support up to six LEDs!



2. Connect the activated Battery module to your Battery Management module.

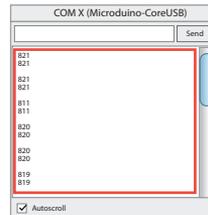


3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!



When the Light Sensor is covered or is in the dark, the LED will light up. That is when you can use bluetooth to control the brightness and color.

### 3 Adjust Code



- Click  after uploading the code . A window will appear, displaying the loudness that the microphone sensor is receiving.

```
#define lightSensorPin A0
#define lightSwitch 500
```

- In the code you uploaded earlier, find the above line. Currently, "lightSwitch" is defined to be 500. You can adjust this value to change at what brightness the LED turns on. Upload your program again for it to take effect.

### 4 Download the Android App

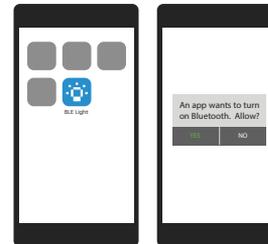


[www.microduino.cc/download](http://www.microduino.cc/download)

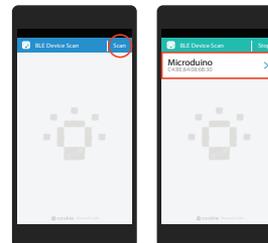
1. Scan the QR code to the left (or go to our website) and download the Bluetooth Light APP.

**ATTENTION:**

The Bluetooth Light APP currently only supports Android 4.0 or higher.



2. Begin the app after installation. If your bluetooth function is not turned on, Android will remind you. Choose to enable bluetooth.



3. Tap Scan. Your Android will begin searching for bluetooth devices near you. Select the device named Microduino.

There are three different preset ambiances.

Cool ambiance mode

Warm ambiance mode

Ambiance changes depending on the sounds of its environment

This area displays your previous color scheme.

LED ON/OFF switch



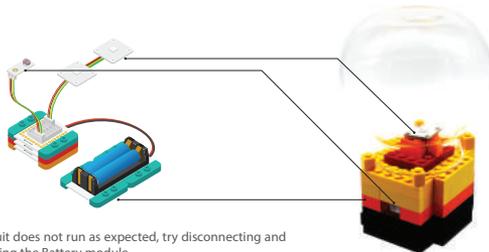
A green status bar indicates that the connection is successful. Tap "Disconnect" to disconnect your project from Android.

Use the slider to change between control modes.

Use this slider to adjust the brightness.

4. After your Android is connected, enjoy creating vibrant colors right from your phone!

### 5 Make your LEGO® Lamp!



• If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.

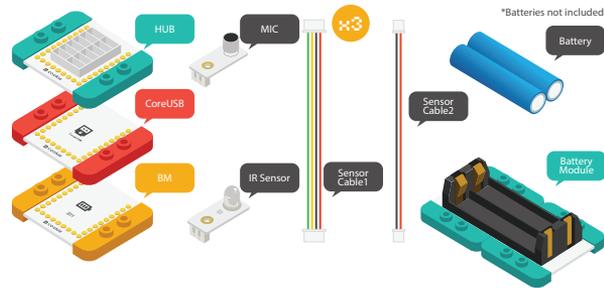
CHEESE!



## PROJECT 3 SAY CHEESE!

Think you look extra good today? Yell "Cheese!" and take a picture.

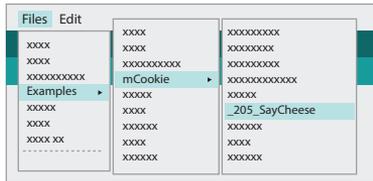
## You will need



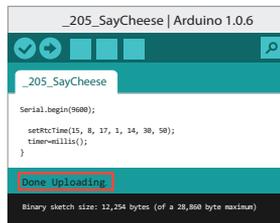
## 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

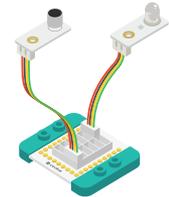
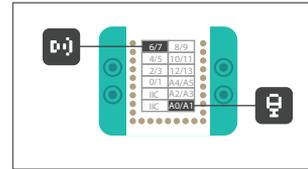


2. Select Files > Examples > mCookie > \_205\_SayCheese.

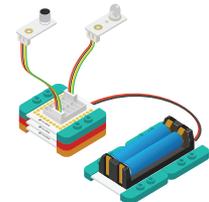


3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

## 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



2. Connect the activated Battery module to your Battery Management module.

3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

## 3 Adjust Code

```
#define voice 400
```

• The voice sensitivity has been preset to 400. You can adjust it to whatever you like.

```
delay(10);
Serial.println(num);
if (num < 20)
{
  take = true;
  takePhoto();
  Serial.println("takePhoto...");
}
```

• This number indicates the duration in which the sound must be sustained in milliseconds.

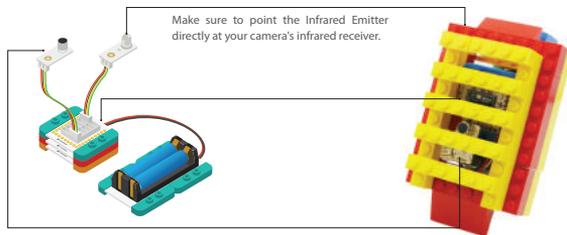
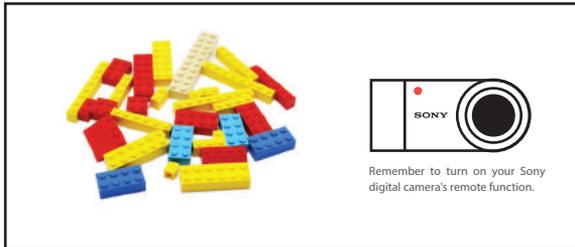


**ATTENTION:**

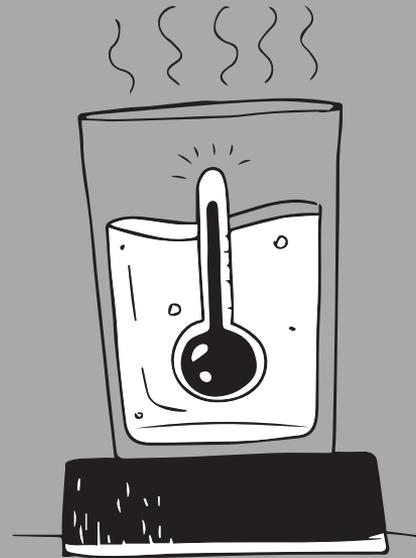
Currently this program only supports Sony digital cameras that have remote capabilities. We will update our list of supported cameras in our wiki as they become available.

- When the sound is sustained for a long enough time, the Infrared Emitter will send a signal to your camera. Smile!

## 4 Make your LEGO® Soy Cheese!



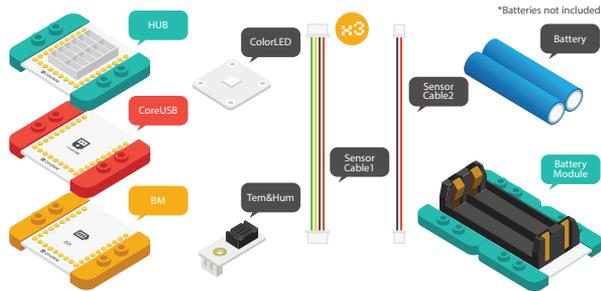
- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



## PROJECT 4 Thermo Cup

Not sure when your hot cocoa is ready to drink? No worries. Just take a look at the color!

## You will need



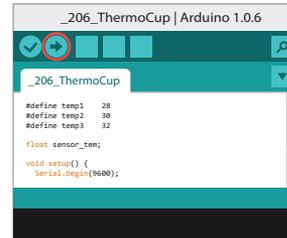
### 1 Upload the Code



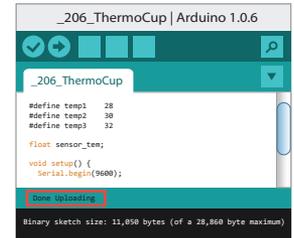
1. Connect the CoreUSB module to your computer and then start the Arduino IDE.



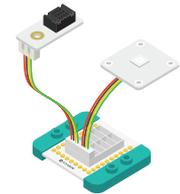
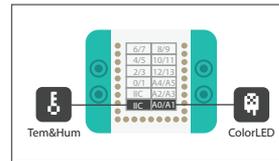
2. Select Files > Examples > mCookie > \_206\_ThermoCup



3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.



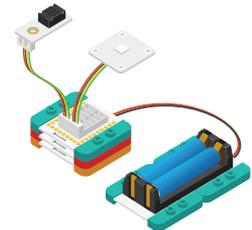
### 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



2. Connect the activated Battery module to your Battery Management module.



3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

### 3 Adjust Code



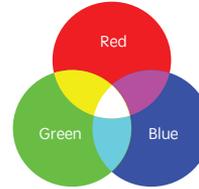
- Click  after uploading the code. A window will appear, displaying the temperature measured.

```
#define temp1 28  
#define temp2 30  
#define temp3 32
```

- There are currently 3 preset temperatures. You can change the values or even add additional temperatures.

```
if (sensor_tem <= temp1)  
  colorSet(strip.Color(0, 255, 0));  
  
else if (sensor_tem > temp1 && sensor_tem <= temp2)  
  colorSet(strip.Color(0, 0, 255));  
  
else if (sensor_tem > temp2 && sensor_tem <= temp3)  
  colorSet(strip.Color(255, 255, 0));  
  
else  
  colorSet(strip.Color(255, 0, 0));
```

- This code segment indicates what colors to display depending on the temperature. Try changing the color or adding even more temperature differences!



### About RGB

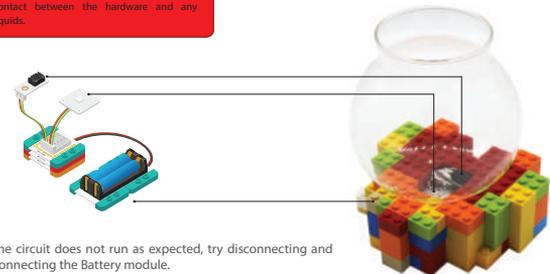
RGB values are used to create different colors. R (Red), G (Green), and B (Blue) are the primary colors. Each color has a value between 0 and 255 to represent its intensity. For example, (0, 255, 0) is the color green, because Red and Blue both have values of 0.

### 4 Make your LEGO® Thermo Cup



#### ! WARNING:

All modules and sensors are conducting electricity. Please do not directly make contact between the hardware and any liquids.



- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.

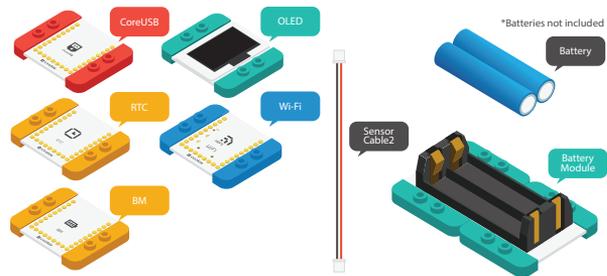


## PROJECT 5

### mWatch (Wi-Fi)

Use mCookie to make your own watch! If you are connected to WiFi it will automatically recalibrate itself.

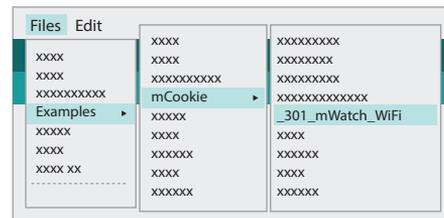
## You will need



## 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.



2. Select Files > Examples > mCookie > \_301\_mWatch\_WiFi.

```

_301_mWatch_WiFi | Arduino 1.0.6
_301_mWatch_WiFi | rtc.h
#define SSID "xxxxxxxx"
#define PASSWORD "xxxxxxxx"
#define HOST_NAME "pool.ntp.org"
#define HOST_PORT (123)
#define INTERVAL_NET
Done Uploading
Binary sketch size: 17,028 bytes (of a 28,868 byte maximum)

```

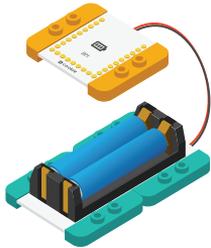
```

_301_mWatch_WiFi | Arduino 1.0.6
_301_mWatch_WiFi | rtc.h
#define SSID "xxxxxxxx"
#define PASSWORD "xxxxxxxx"
#define HOST_NAME "pool.ntp.org"
#define HOST_PORT (123)
#define INTERVAL_NET
Done Uploading
Binary sketch size: 17,028 bytes (of a 28,868 byte maximum)

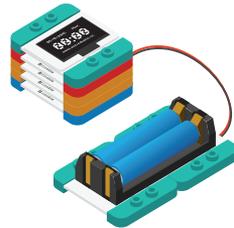
```

3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

## 2 Build the circuit



2. Connect the activated Battery module to your Battery Management module.



3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

## 3 Adjust Code

```

_301_mWatch_WiFi
#define SSID "xxxxxxxx"
#define PASSWORD "xxxxxxxx"
#define HOST_NAME "pool.ntp.org"
#define HOST_PORT (123)

```

• Find the above lines in the program. Change the content between the parenthesis to WiFi name and password. **Note: Do not remove the parenthesis.**

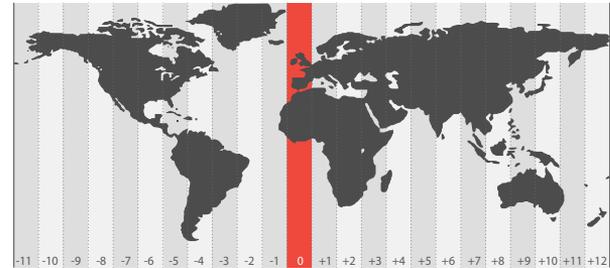
```

rtc.h
#define TIME_ZONE +8

```



• In the rtc.h file, find the above line. Change the TIME\_ZONE to yours. Afterwards, upload the program one more time.

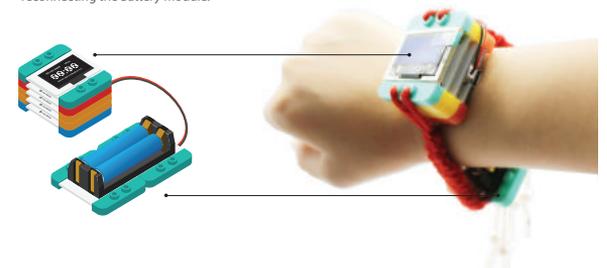


\* Picture only for reference.

## 4 Make your mWatch



• If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



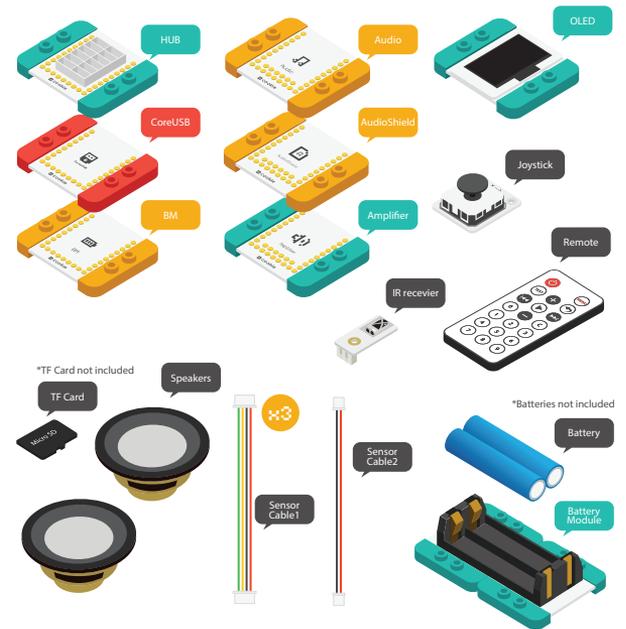


## PROJECT 6

# MUSIC BOX (Remote Control)

Ready to get funky? Play music from a micro SD card and use the joystick or Remote Control to select songs and adjust the volume.

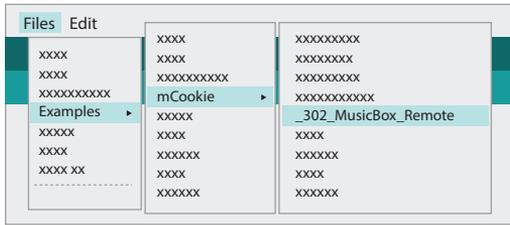
## You will need



## 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

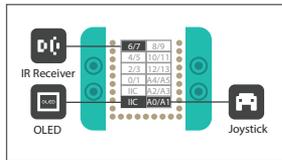


2. Select Files > Examples > mCookie > \_302\_MusicBox\_Remote.

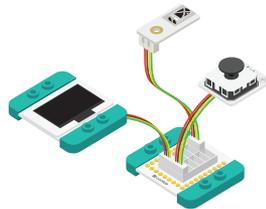


3. Click to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

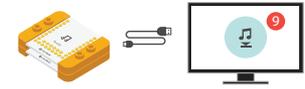
## 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



2. Insert a micro SD card into the Audio Shield.



3. Stack the Audio Shield and Audio modules. Even though your micro SD card may be able to store many songs, the program we provide only supports up to 9 songs.



4. Connect the activated Battery module to your Battery Management module.



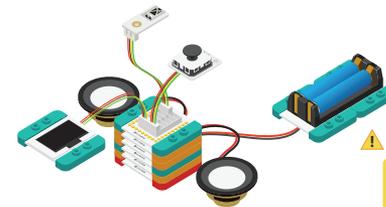
5. Connect the speakers into the Amplifier module.



6. Stack the Amplifier, Audio, and Audio Shield modules.

### ⚠ ATTENTION:

All three modules must be stacked together for proper operation.



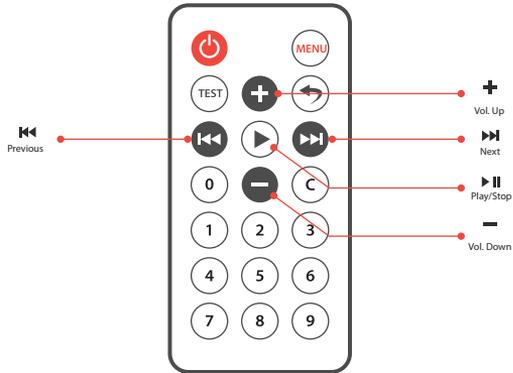
### ⚠ ATTENTION:

The Audio module needs to be powered by the Battery module. Supplying power by connecting a USB cord to the CoreUSB module may cause issues.

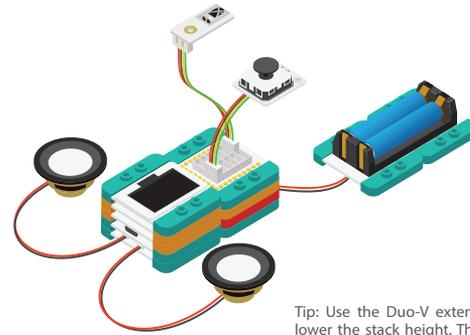
7. Stack all the modules together in any order you like (Except three Audio modules, they must be together). Congratulations, you have finished building the circuit!

## 3 How to Operate

- You can use the Remote Control to play your favorite songs.



- You can also use the joystick as shown below.

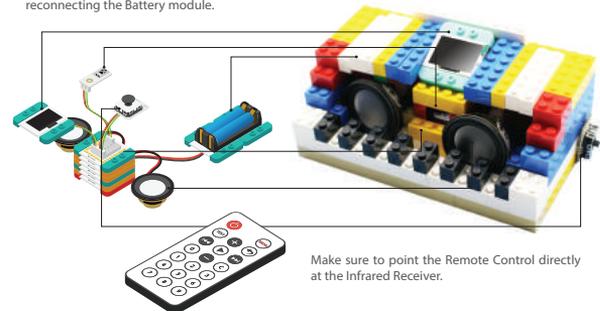


Tip: Use the Duo-V extension module to lower the stack height. The OLED module would not even have to be connected to the Hub.

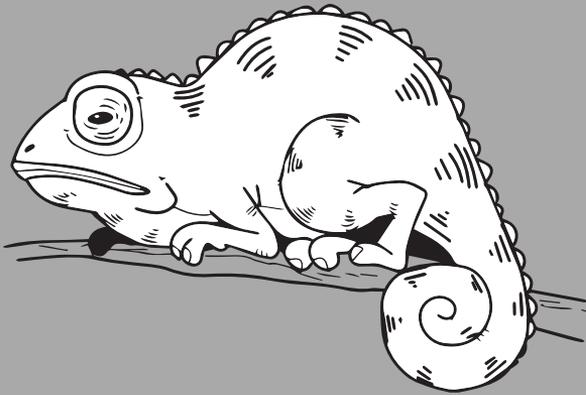
## 4 Make your LEGO® Music Box



- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



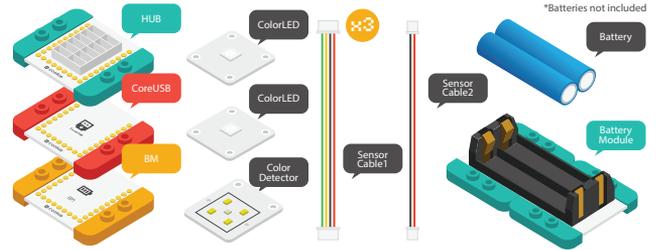
Make sure to point the Remote Control directly at the Infrared Receiver.



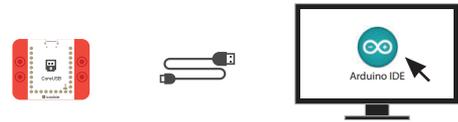
PROJECT 7  
**Bashful Chameleon**

Use the Color Sensor to detect the surrounding colors of your Chameleon. Watch as his  
 ambiance changes!

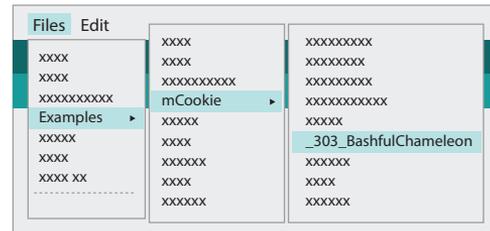
You will need



1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.



2. Select Files > Examples > mCookie > \_303\_BashfulChameleon.

```

_303_BashfulChameleon | Arduino 1.0.6
colorSensor.h
#include <iostream.h>
#include <math.h>
#include <Adafruit_NeoPixel.h>
#define redPin 32767
#define greenPin 32767

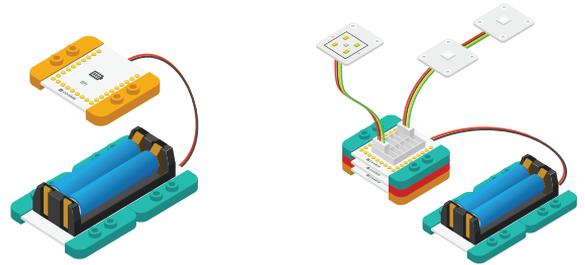
```

```

_303_BashfulChameleon | Arduino 1.0.6
colorSensor.h
#include <iostream.h>
#include <math.h>
#include <Adafruit_NeoPixel.h>
#define redPin 32767
#define greenPin 32767

```

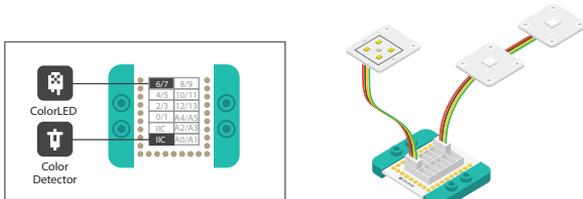
3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.



2. Connect the activated Battery module to your Battery Management module.

3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

## 2 Build the circuit

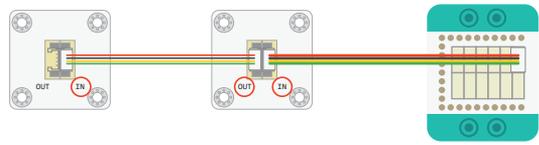


1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.

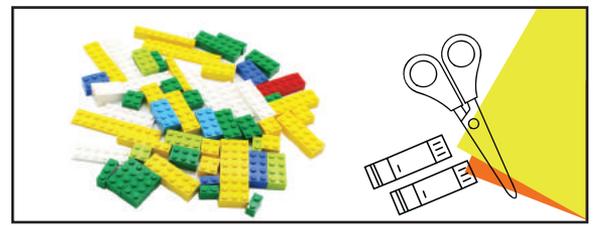


• If the surroundings are dark, you can turn on the LED light on the sensor.

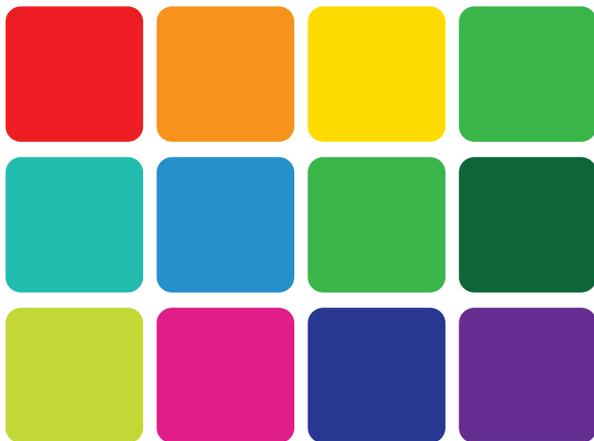
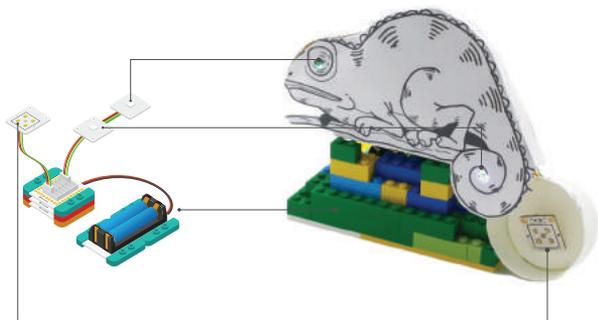
## 3 Make your LEGO Chameleon



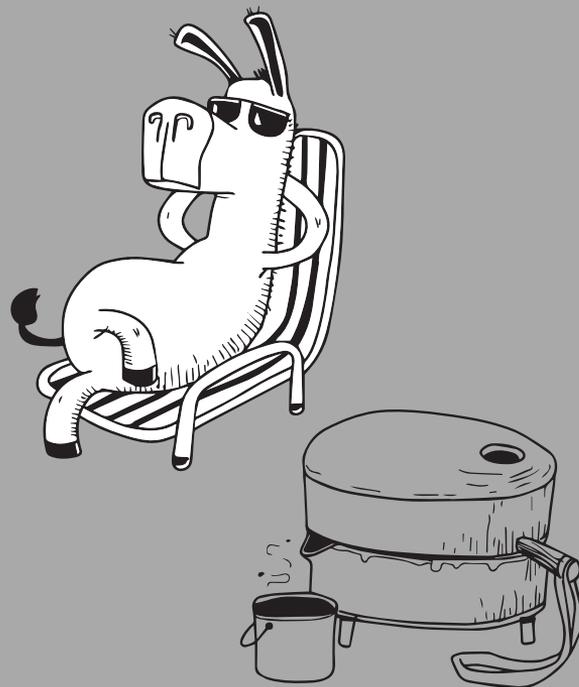
To connect multiple LEDs in a series, follow the above diagram. Notice that the sensor cable connects to the IN ports of the LEDs.



- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



- Try holding up the above color palettes to your Chameleon. Due to interference from other lights in the environment, there may be some discrepancies in LED color.

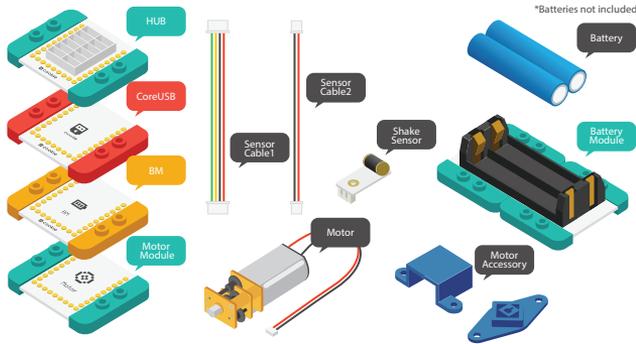


## PROJECT 8

### Dozing Donkey

Mr. Donkey is quite lazy. Make sure he doesn't slack off by tapping the ground!

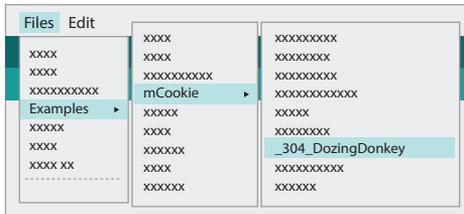
## You will need



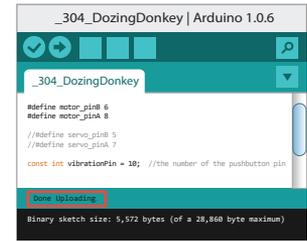
### 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

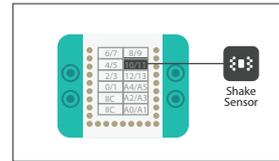


2. Select Files > Examples > mCookie > \_304\_DozingDonkey

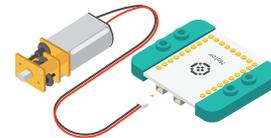


3. Click to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

### 2 Build the circuit



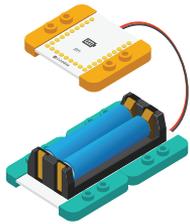
1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



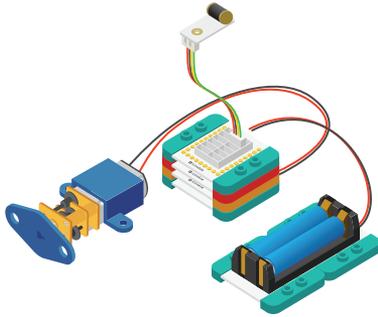
2. Connect the motor to the Motor module as shown above.



3. According to your project needs, you can attach the motor accessories for attachment to other devices.



2. Connect the activated Battery module to your Battery Management module.



**⚠ ATTENTION:**

The Motor module needs to be powered by the Battery module. Supplying power by connecting a USB cord to the CoreUSB module may cause issues.

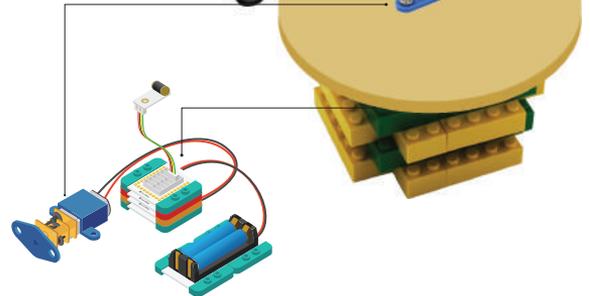
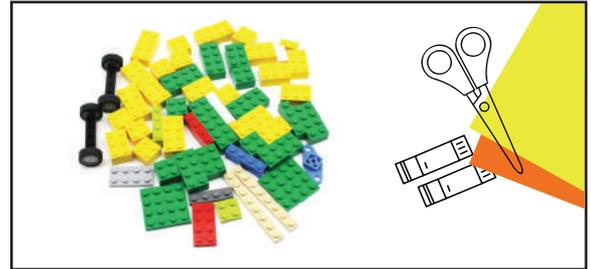
3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

### 3 Adjust Code

```
void fullForward() {
  //digitalWrite(motor_pinB, HIGH);
  analogWrite(motor_pinB, 255); //0-255
  digitalWrite(motor_pinA, LOW);
  delay(1000);
}
```

- Find the above line of code. 255 is the motor's largest speed value. Adjust it as you like. (We recommend choosing values higher than 60 to maintain signal stability.)
- Adjust the delay() value to determine how long the motor runs. It is preset to 1000 milliseconds (or 1 second).

### 4 Make your LEGO® Donkey



- If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.

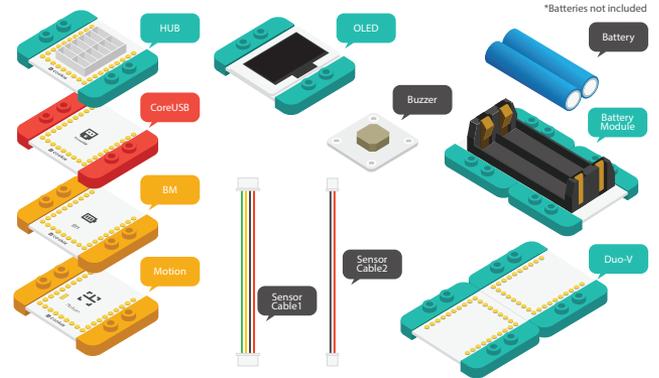


## PROJECT 9

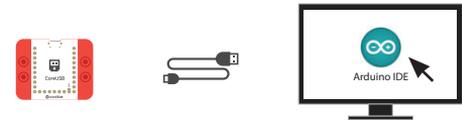
# Personal Trainer

Set the work out duration and the buzzer will sound. Only until you are done with the exercise will it turn

## You will need



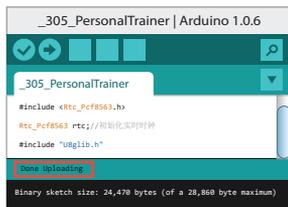
### 1 Upload the Code



1. Connect the CoreUSB module to your computer and then start the Arduino IDE.

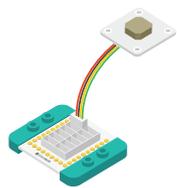
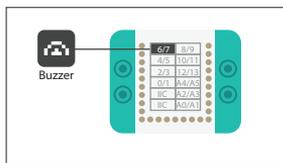


2. Select Files > Examples > mCookie > \_305\_PersonalTrainer.

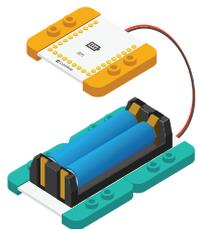


3. Click  to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

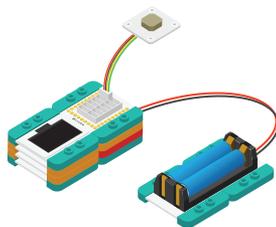
## 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



2. Connect the activated Battery module to your Battery Management module.



3. Stack all the modules together in any order you like. Congratulations, you have finished building the circuit!

## 3 Adjust Code

```
setRtcTime( 15, 5, 18, 1, 00, 00, 00 );
```

↓ Year  
 ↓ Month  
 ↓ Date  
 ↓ Week  
 ↓ Hr.  
 ↓ Min.  
 ↓ Sec.

1. Find the above line of code to set your watch's time. After you are done, upload your program again.

```
//setRtcTime( 15, 5, 18, 1, 00, 00, 00 );
```

2. Now, add // in front of this line of code, turning the line gray. Upload your program again. Now your RTC module functions as a normal clock! If you need to adjust the time, repeat from the above step.

```
#define set_time_hour 00
#define set_time_min 01
```



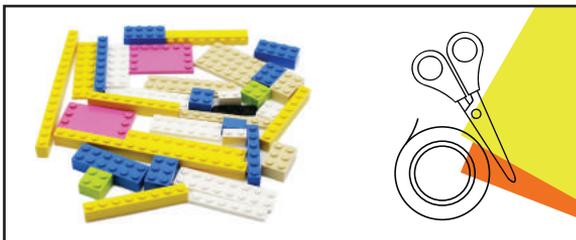
• Adjust above values to set when you want to exercise every day.

```
int Sport_num = 8;
```

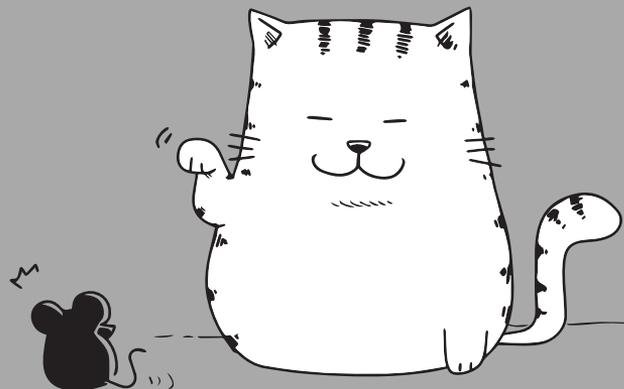
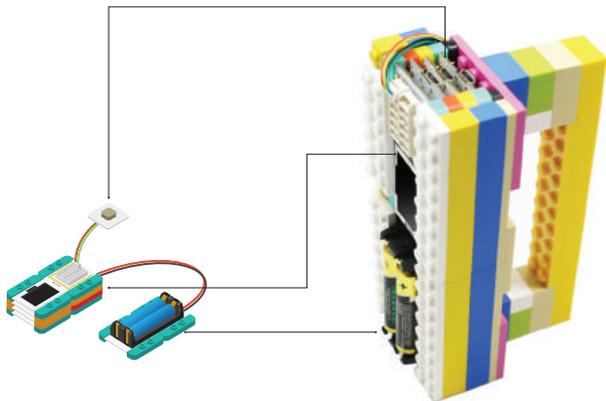


• The above number indicates number of repetitions in your exercise. Once the buzzer goes off at your set time, pick up the module and start moving! The buzzer will stop once you have completed the required number of repetitions. Upload the program again.

#### 4 Make your LEGO® Personal Trainer



• If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.

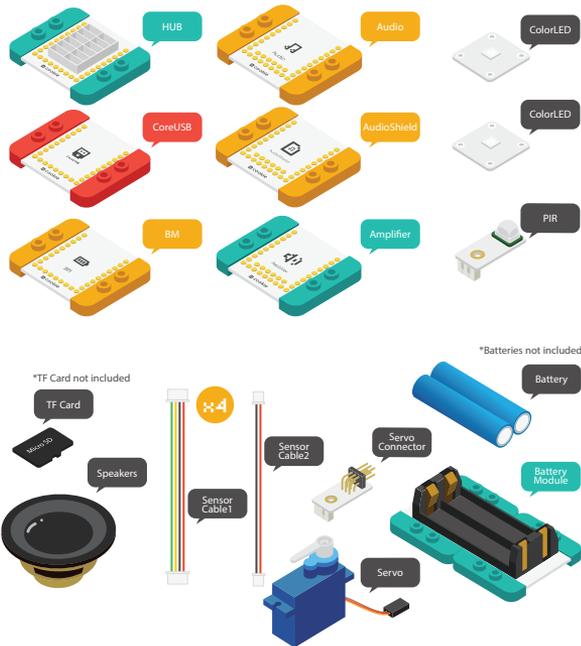


#### PROJECT 10

### Fortune Cat

This friendly companion's eyes will light up with delight and wave when you pass by. And of course, Fortune Cat will purr contently as well!

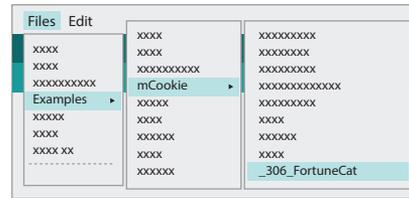
## You will need



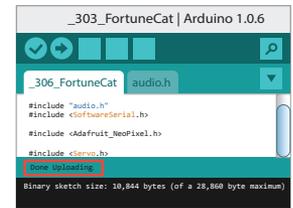
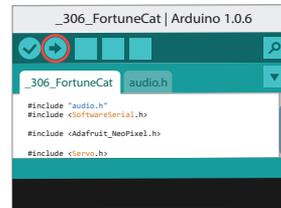
### 1 Upload the Code



1. Connect the CoreUSB module to your sensor computer and then start the Arduino IDE.

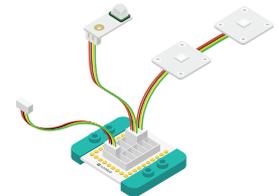
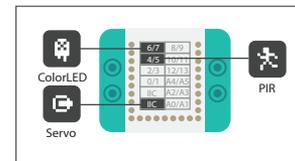


2. Select File > Examples > mCookie > \_306\_FortuneCat

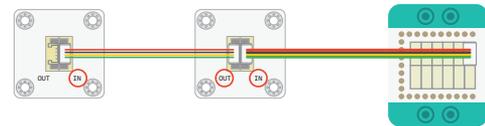


3. Click to upload the program. When "Done Uploading" appears, the program should have successfully been uploaded into the CoreUSB.

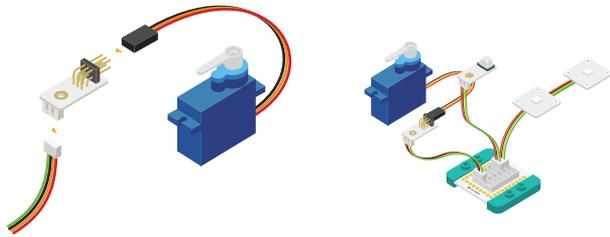
### 2 Build the circuit



1. Using the above diagram as reference, connect the sensors into the corresponding ports on the Hub module.



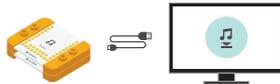
To connect multiple LEDs in a series, follow the above diagram. Notice that the sensor cable connects to the IN ports of the LEDs. You can support up to six LEDs!



2. Connect the servo to the Servo Connector. Then connect that to the IIC port in the Hub module.



3. Insert a micro SD card into the Audio Shield.



4. Stack the Audio Shield and Audio module. Connect them to your PC. Then, upload one greeting sound.

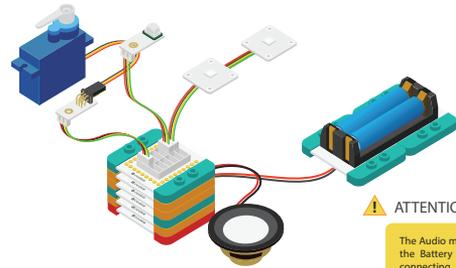


**ATTENTION:**  
All three modules must be stacked together for proper operation.

5. Stack the Amplifier, Audio, and Audio Shield modules.



6. Connect the activated Battery module to your Battery Management module.



**ATTENTION:**

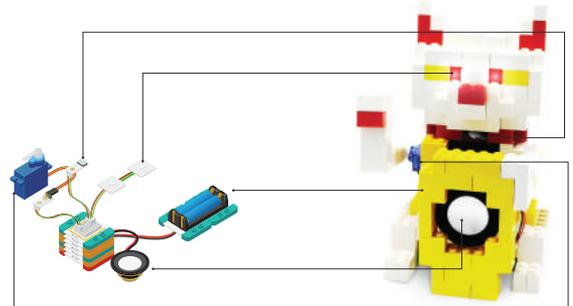
The Audio module needs to be powered by the Battery module. Supplying power by connecting a USB cord to the CoreUSB module may cause issues.

7. Stack all the modules together in any order you like (Except three Audio modules). Congratulations, you have finished building the circuit!

### 3 Make your LEGO® Fortune Cat!



• If the circuit does not run as expected, try disconnecting and reconnecting the Battery module.



# Contact

## More Projects & Information

[www.microduino.cc](http://www.microduino.cc)  
[wiki.microduino.cc](http://wiki.microduino.cc)

## Contact

Support: [support@microduino.cc](mailto:support@microduino.cc)  
Media: [community@microduino.cc](mailto:community@microduino.cc)

Community and Technical Support: [www.microduino.cc/forum/](http://www.microduino.cc/forum/)

## Find us on

Facebook / Twitter / G+: Search **Microduino**  
Youtube: Search **Microduino Studio**

NOTE: