

LI-FI

Arduino Lifi Project
by **Ahmed Fawzy** .

what is **LI-FI** ?!

- **LI-FI** Stands for “Light Fidelity” .
- A type of wireless communication using visible light .

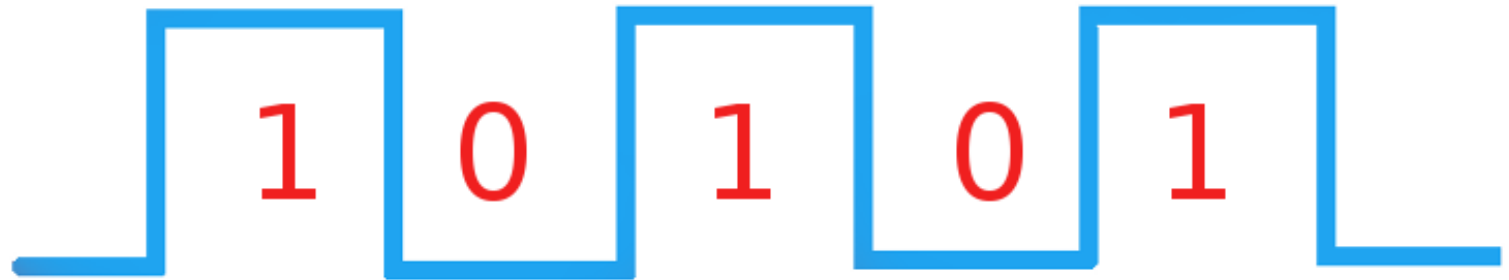
How **wireless communication** works ?!

- Simply instead of using wires to transfer “**Data Signal**” we mix it with a “**carrier wave**” using any type of signal modulation like **AM** (**A**mplitude **M**odulation) in our case .

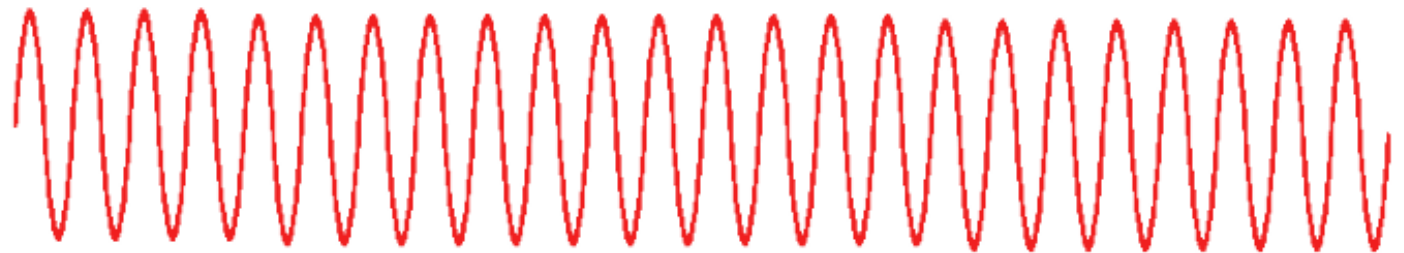
How **AM works ?!**

AM [Amplitude Modulation]

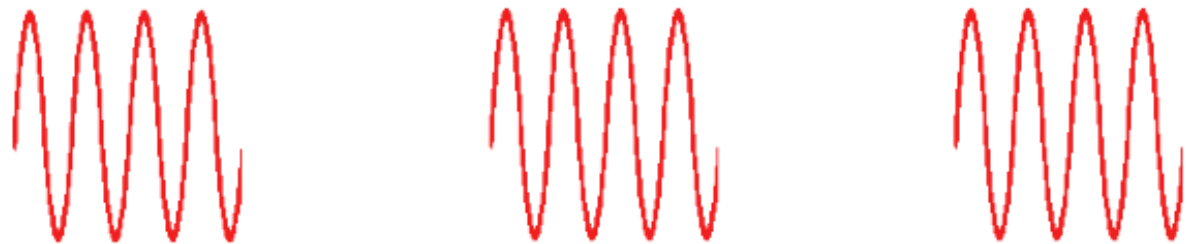
Data signal



Carrier signal



AM signal



**And when it is received it is “Demodulated”
and converted to a “data signal” again .**

Arduino LI-FI Project

First how to **connect** two **Arduino**s together ?!

There are a lot of protocols like
(**SPI**,**I2C**,**UART**,...)
in our case we wil use **UART** .

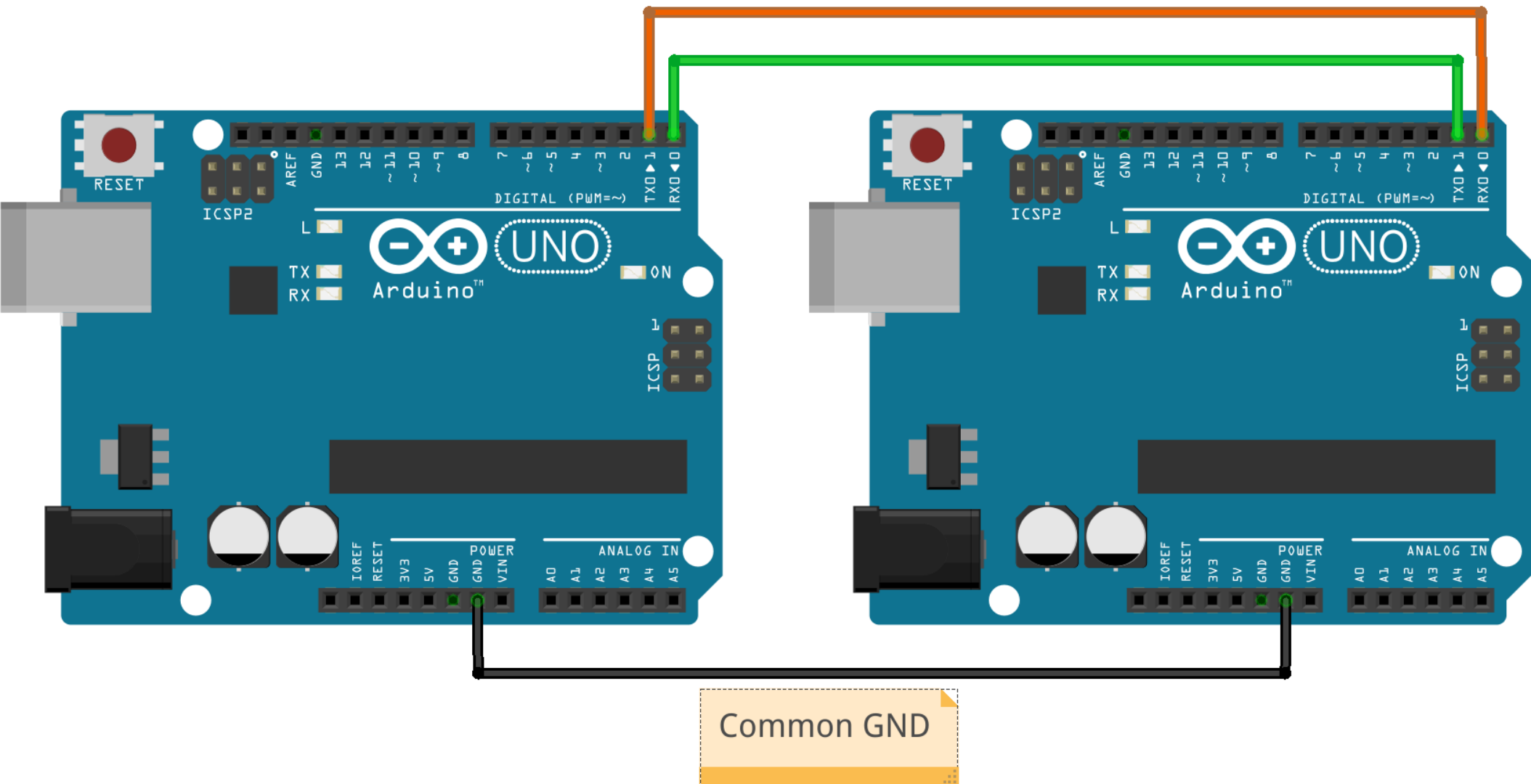
How UART works ?!

UART communication



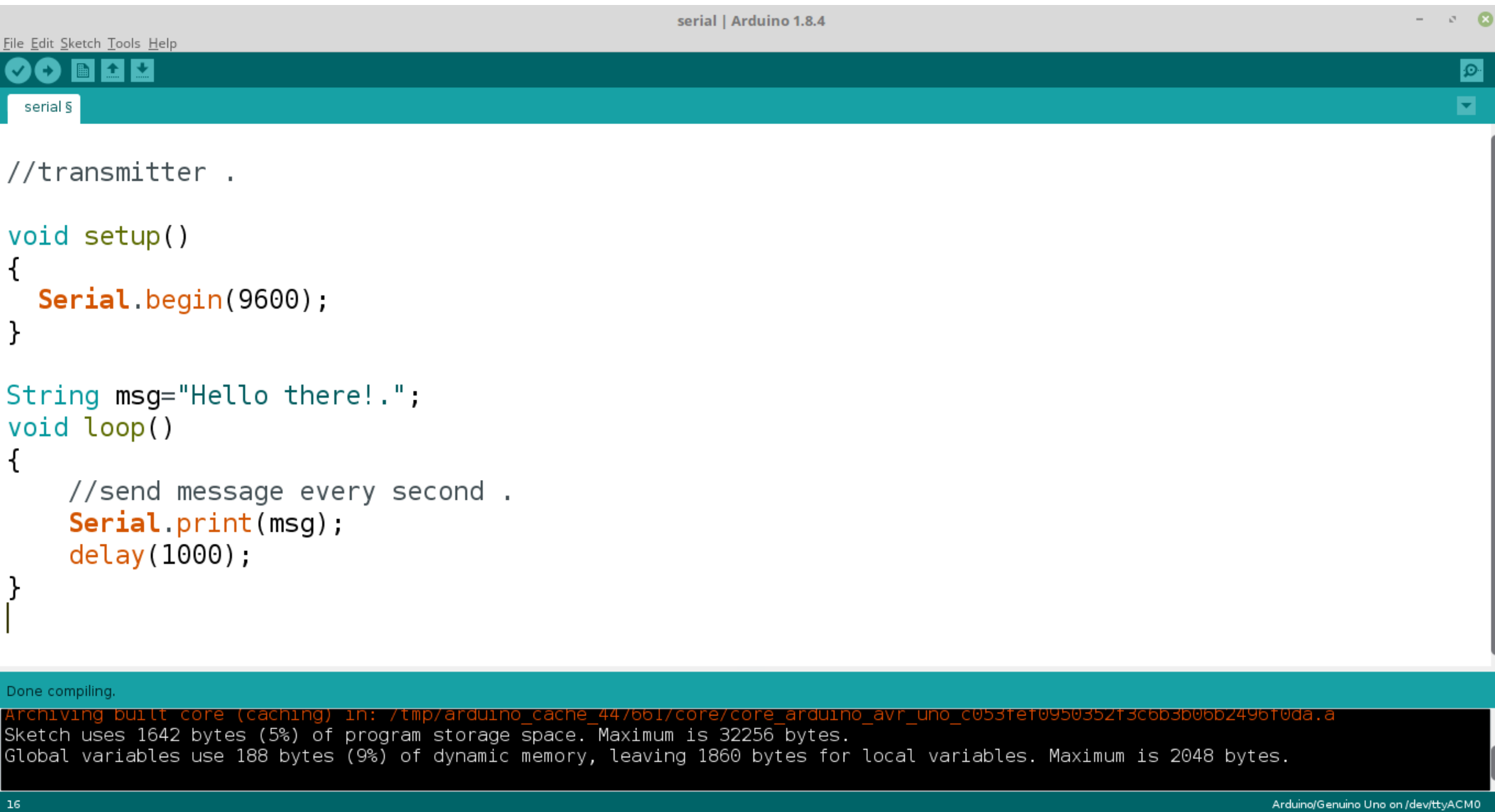
Just kidding :D .

Here how it looks in real life :



{ Code ; }

Transmitter



```
serial | Arduino 1.8.4
File Edit Sketch Tools Help
serial
//transmitter .

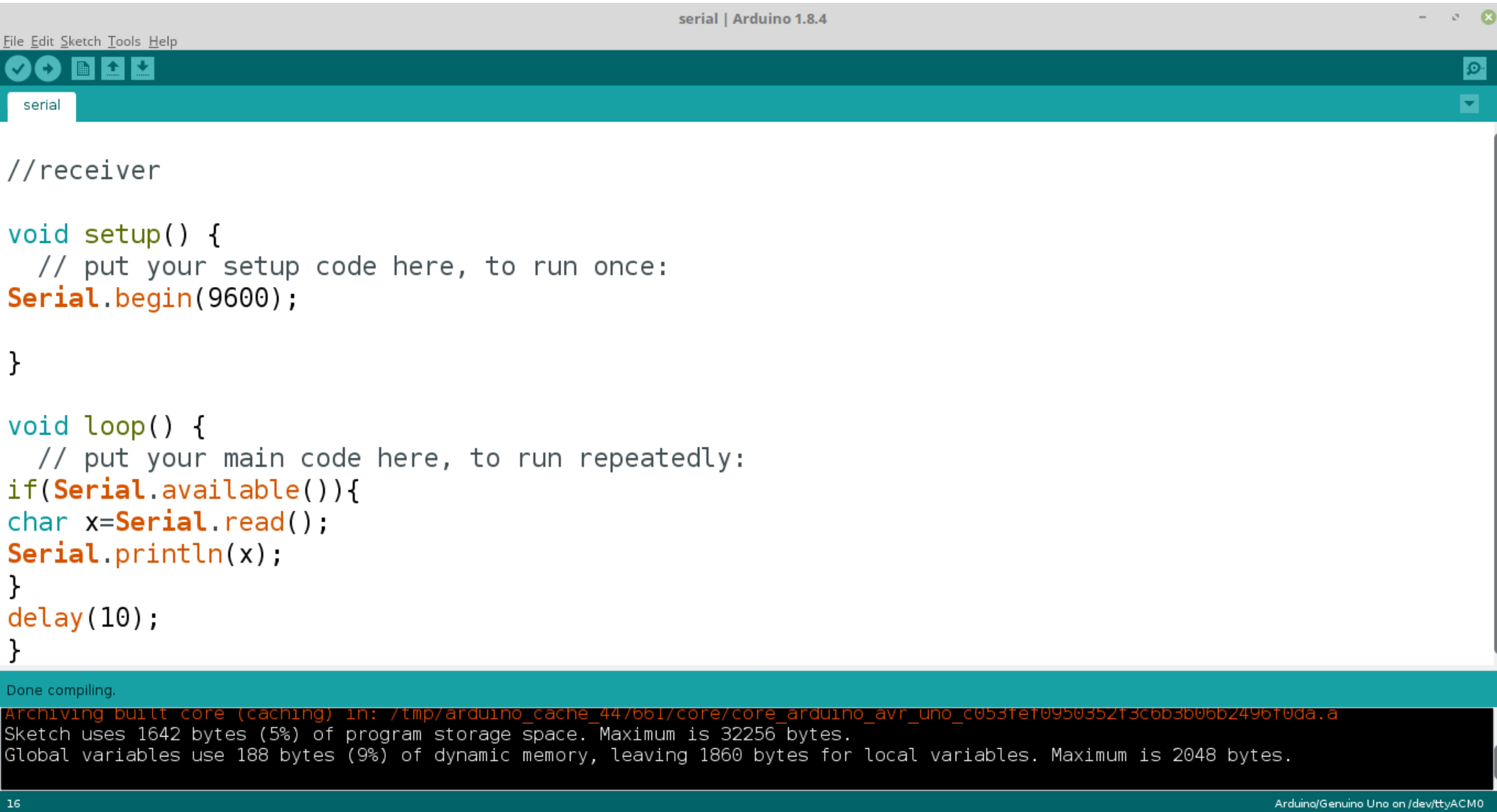
void setup()
{
  Serial.begin(9600);
}

String msg="Hello there!.";
void loop()
{
  //send message every second .
  Serial.print(msg);
  delay(1000);
}

Done compiling.
Archiving built core (caching) in: /tmp/arduino_cache_44/661/core/core_arduino_avr_uno_c053fef0950352f3c6b3b06b2496f0da.a
Sketch uses 1642 bytes (5%) of program storage space. Maximum is 32256 bytes.
Global variables use 188 bytes (9%) of dynamic memory, leaving 1860 bytes for local variables. Maximum is 2048 bytes.
```

16 Arduino/Genuino Uno on /dev/ttyACM0

Receiver



```
serial | Arduino 1.8.4

File Edit Sketch Tools Help

serial

//receiver

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  if(Serial.available()){
    char x=Serial.read();
    Serial.println(x);
  }
  delay(10);
}

Done compiling.
Archiving built core (caching) in: /tmp/arduino_cache_44/661/core/core_arduino_avr_uno_c053fef0950352f3c6b3b06b2496f0da.a
Sketch uses 1642 bytes (5%) of program storage space. Maximum is 32256 bytes.
Global variables use 188 bytes (9%) of dynamic memory, leaving 1860 bytes for local variables. Maximum is 2048 bytes.

16 Arduino/Genuino Uno on /dev/ttyACM0
```

Now lets get rid of wires and use **Light** as
our **Transmitting** signal (**AM**).

How ?!

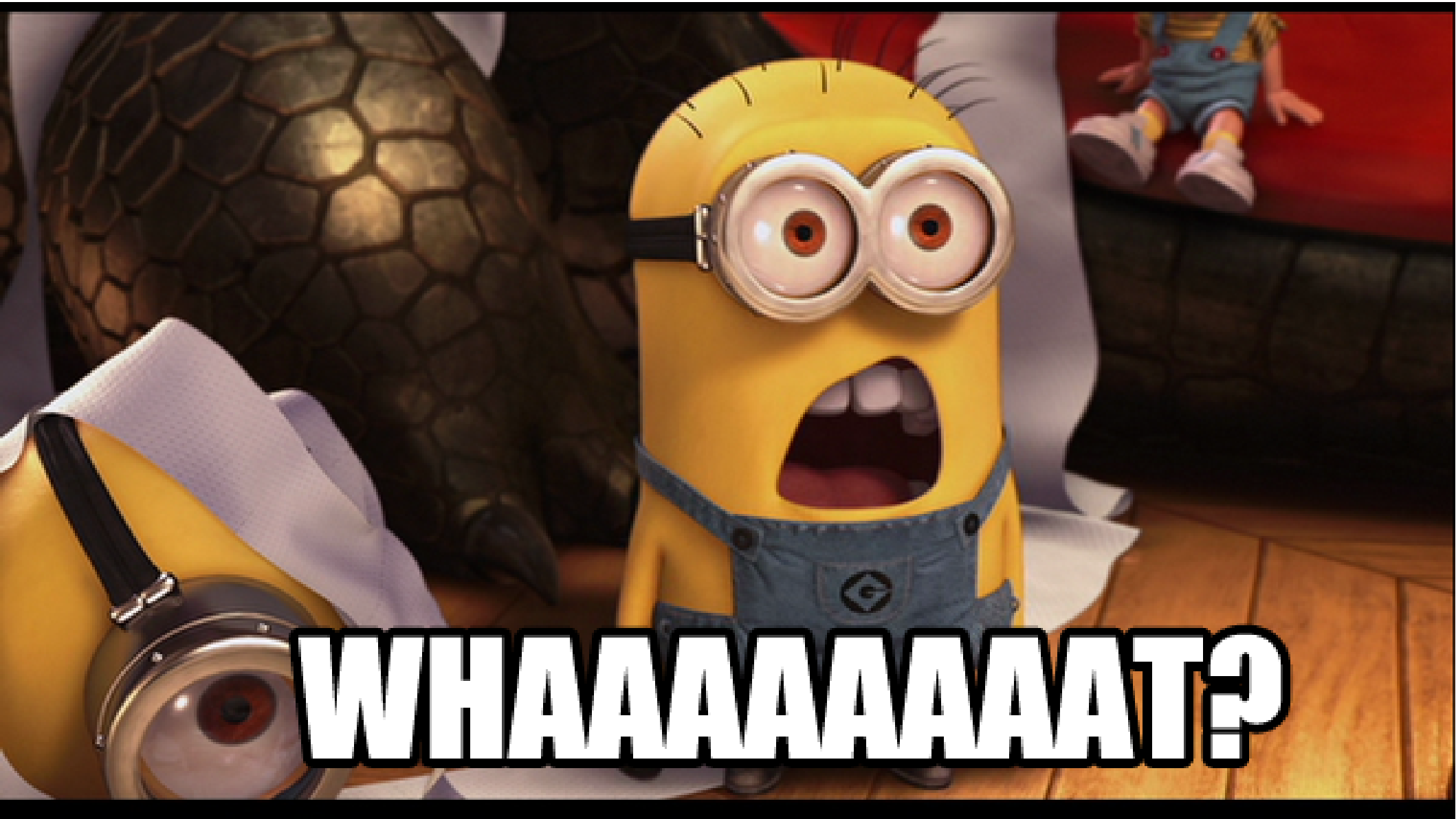
Mostly they use a **Light** source (**LED**) at the **Transmitter** side .

And a **Light** sensor (**Solar Cell**) at the **Receiver** side .

**But its hard to get a Solar Cell
and if you can it will be
expensive .**

Solution :

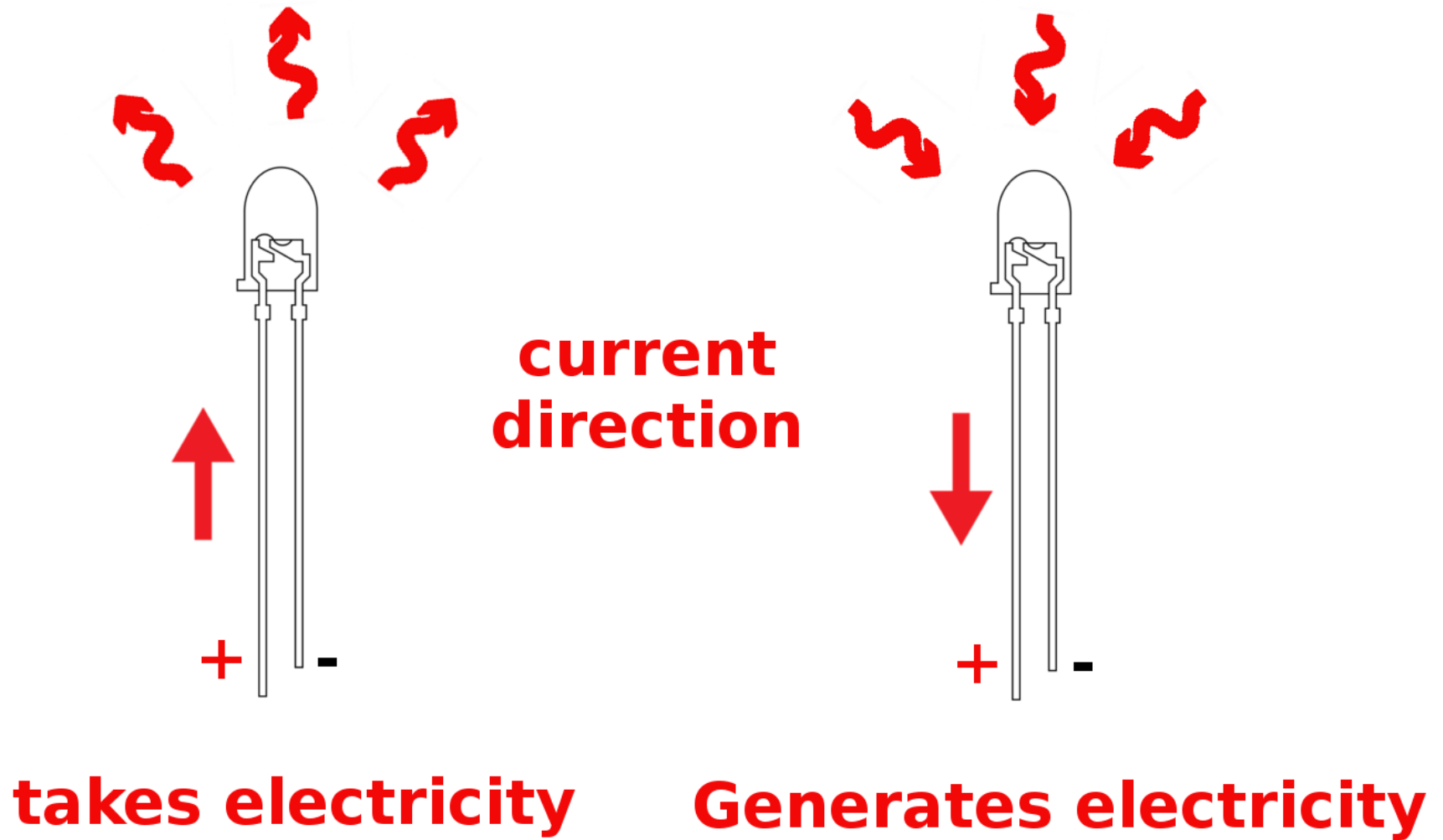
You can use an LED as a Solar Cell .



Yes , LEDs and Solar Cells has the same internal structure .

**If you give an LED electricity it will emit light
but if you give an LED light it will generate
electricity.**

Using LED as a solar cell



Again we have a **Problem** :(

The **LED** can only generate **2 volts Maximum**

.

Again where is the **problem** ?!

In the **physical** world a logic **1** is (**3.5-5**) volts
and a logic **0** is (**0-2.5**) volts .

That means that whatever **signal** the **LED** generates
it will be logic **0** (**0-2**) volts .

In other words it will be read as **0** all **the time** .

Solution :

Using a simple circuit(module**) that acts as an **amplifier** and a **filter**(using **threshold**) .**

Vcc(+5,+3.3)

RX

Data LED

meter ==>

D as Solar cell - +

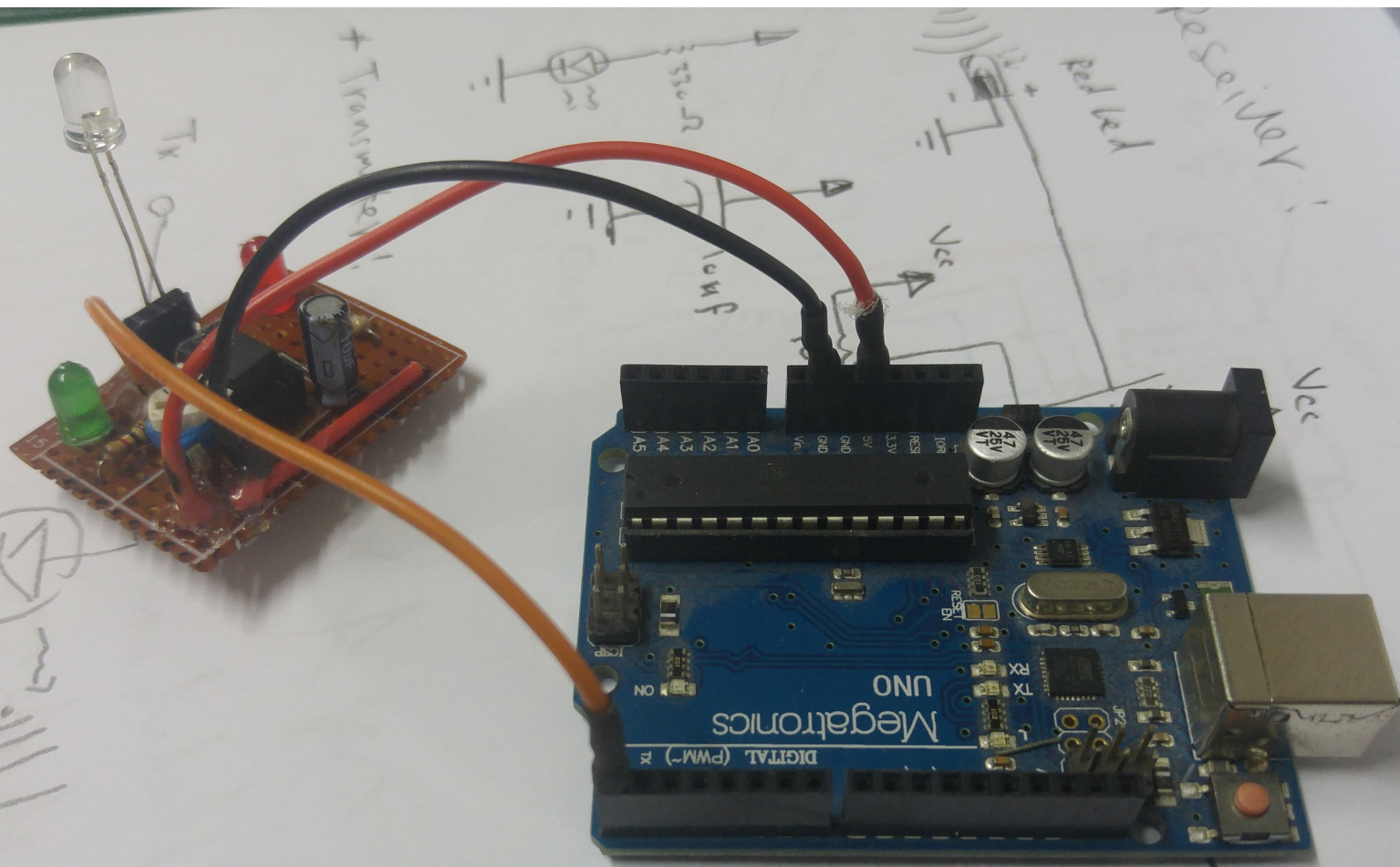
4

Red Red

4 Transm. 10/1

天

22



Data LED

RX LED-RED

Power LED

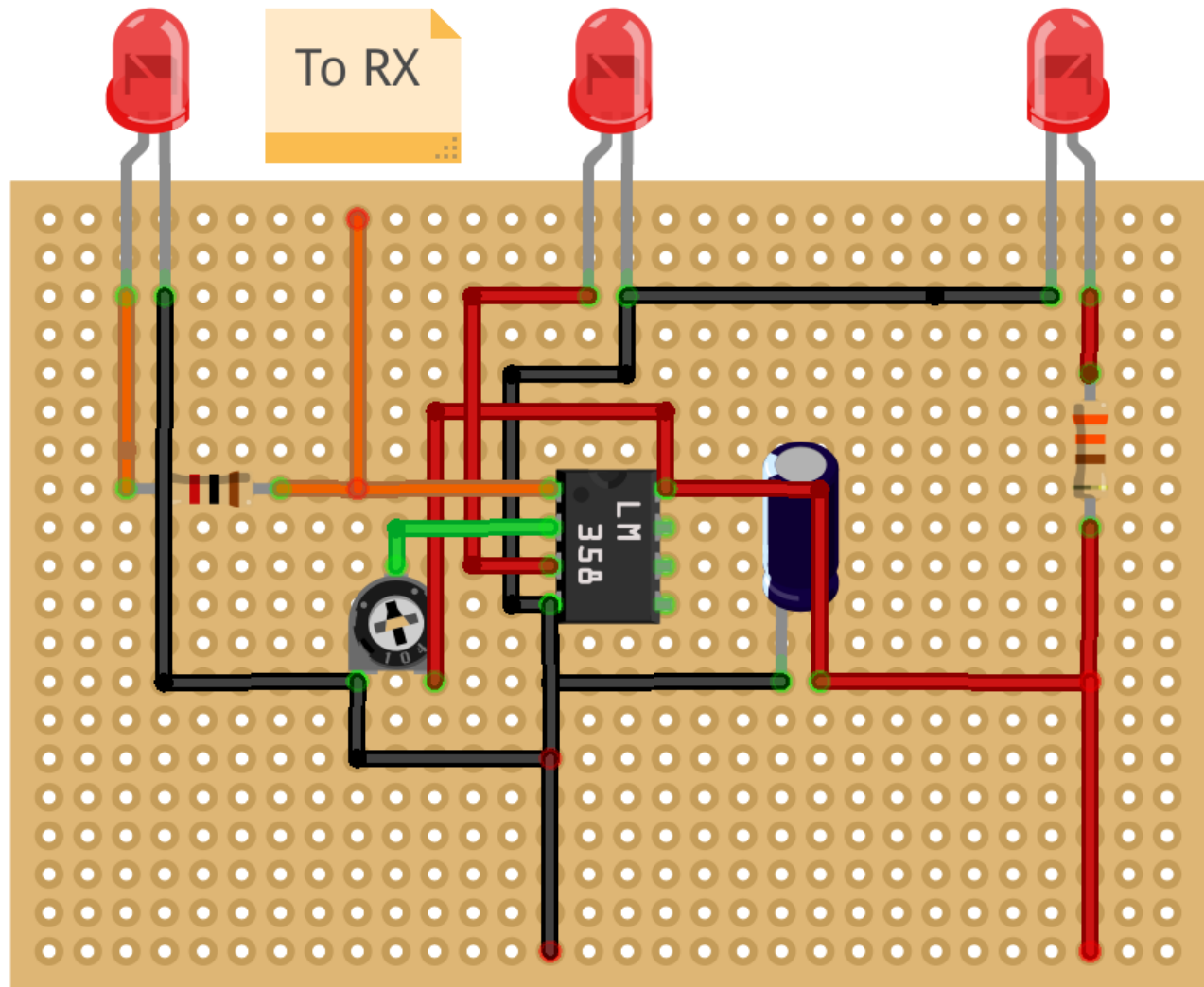
To RX

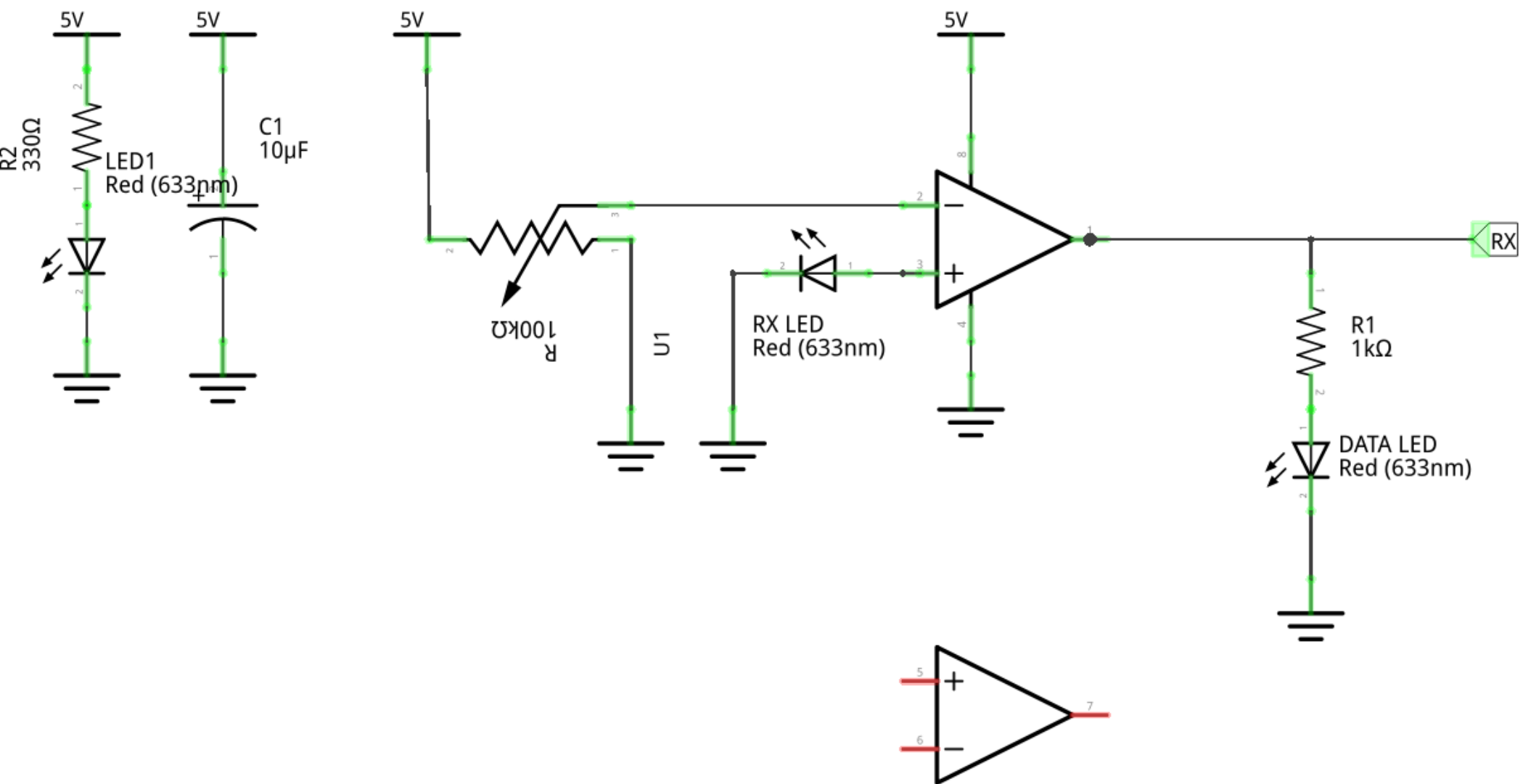
LIFI RX

GND wire

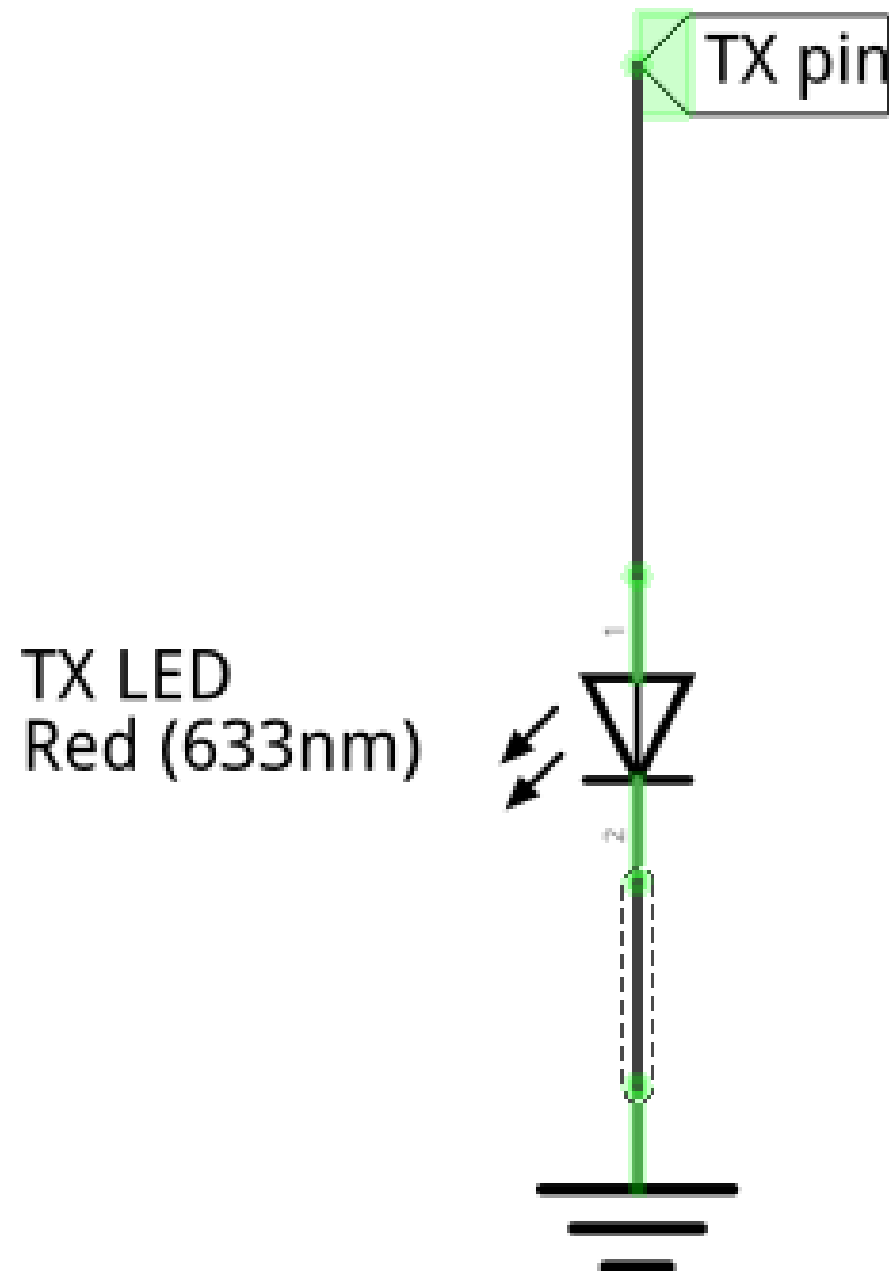
Vcc wire

fritzing





fritzing



fritzing