

```

int LED1=13, LED2 = 12 , LED3 = 11, LED4 =10;
void setup ()
{
  pinMode(LED1, OUTPUT) ;// pinMode (pinno, mode
  pinMode(LED2, OUTPUT) ;
  pinMode(LED3, OUTPUT) ;
  pinMode(LED4, OUTPUT) ;
}
void loop()
{
  digitalWrite(LED1, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)

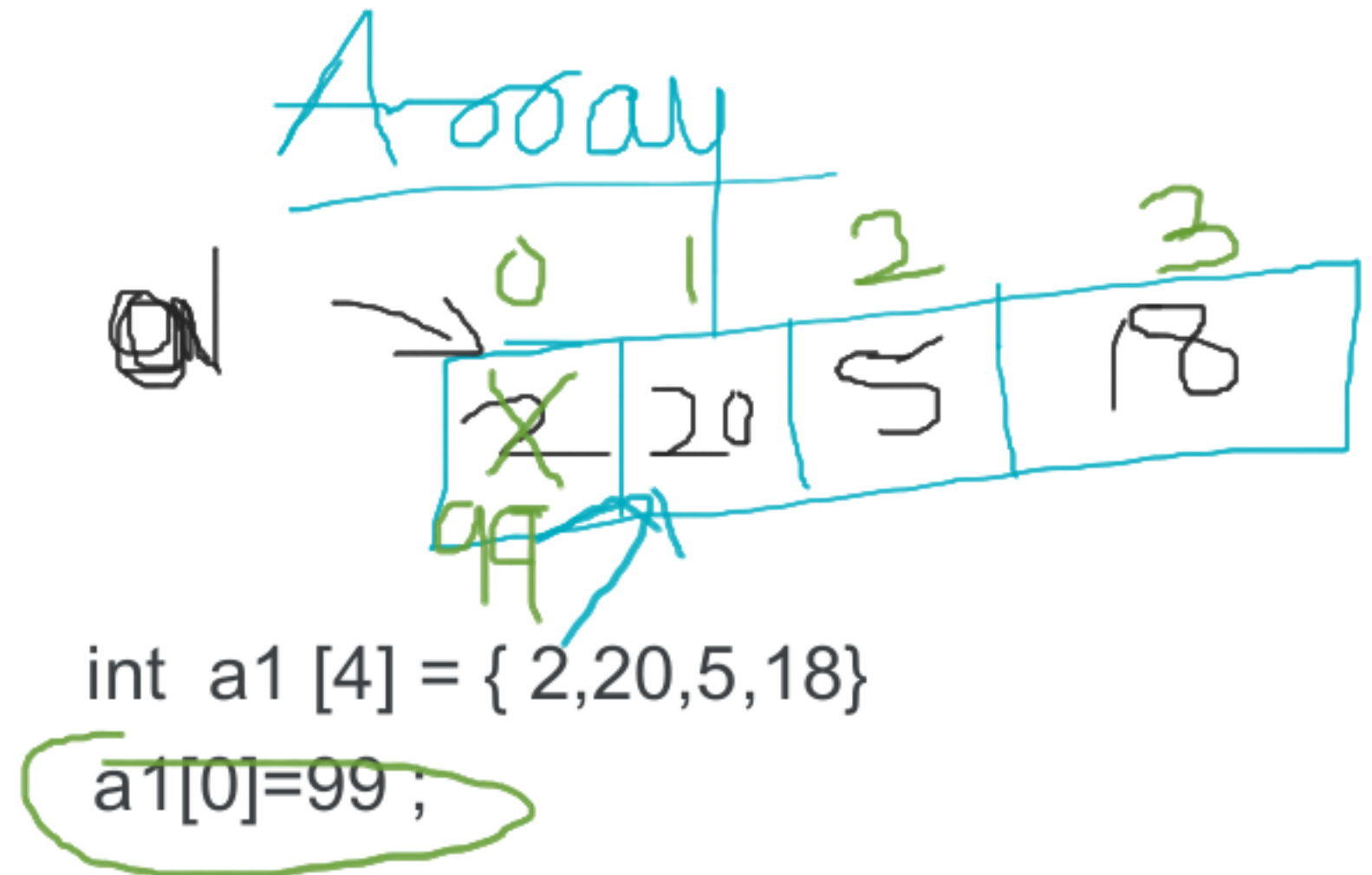
  digitalWrite(LED2, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)

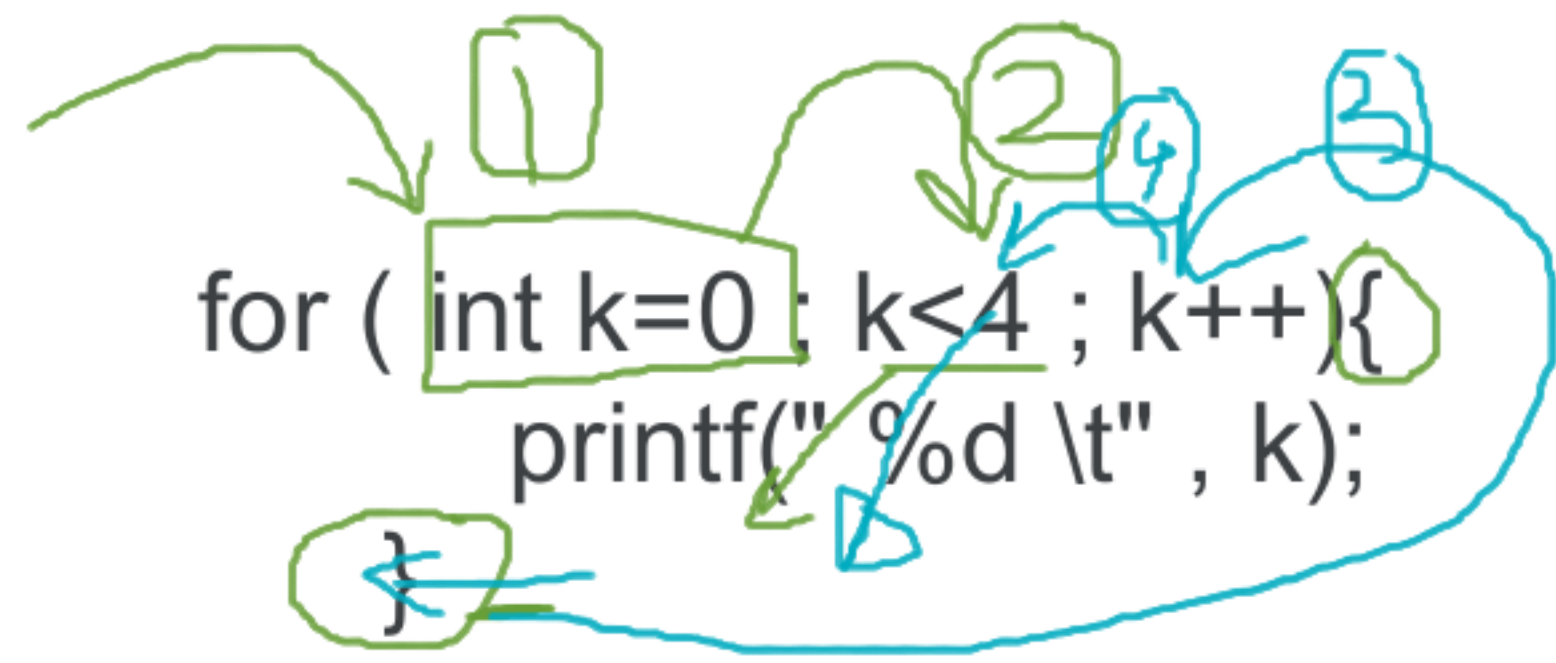
  digitalWrite(LED3, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)

  digitalWrite(LED4, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)

  digitalWrite(LED1, LOW);digitalWrite(LED2, LOW);
  digitalWrite(LED3, LOW); digitalWrite(LED4, LOW);
  delay(1000); // Wait for 1000 millisecond(s)
}

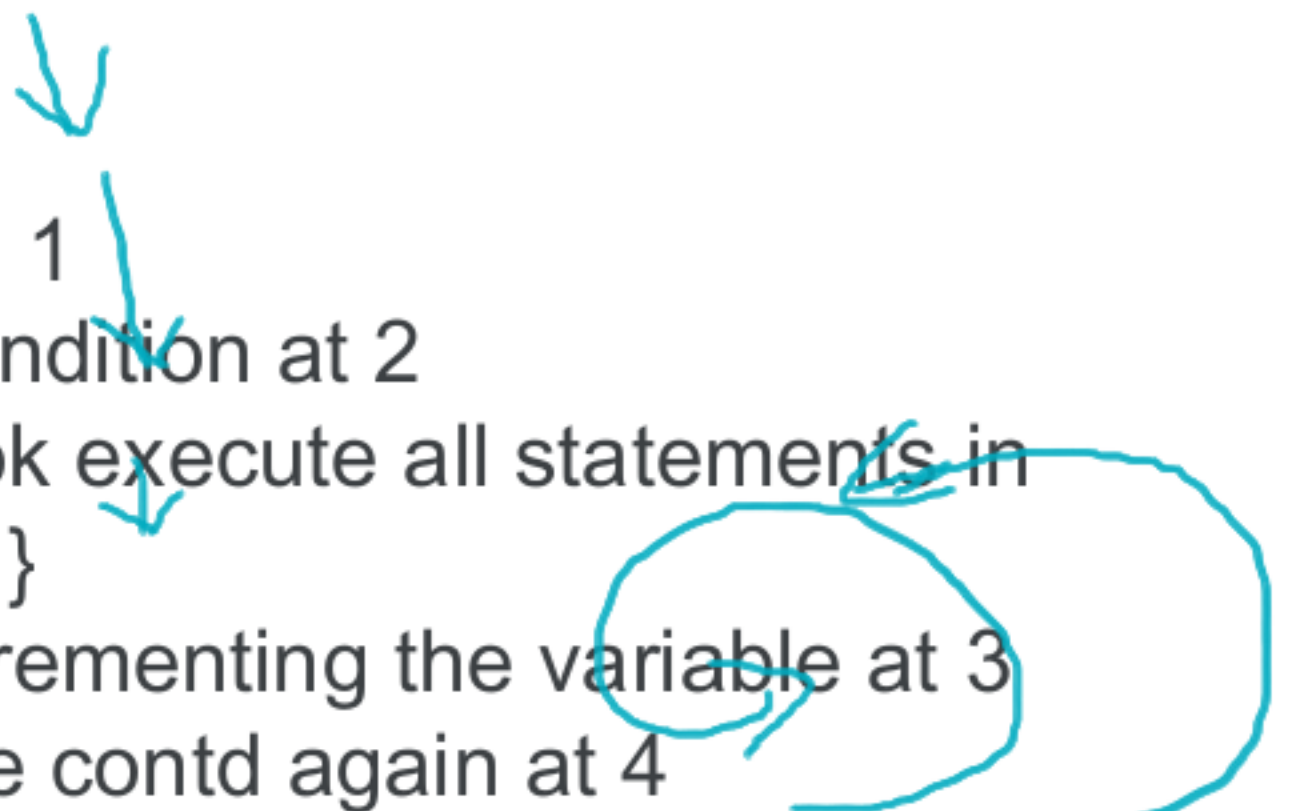
```



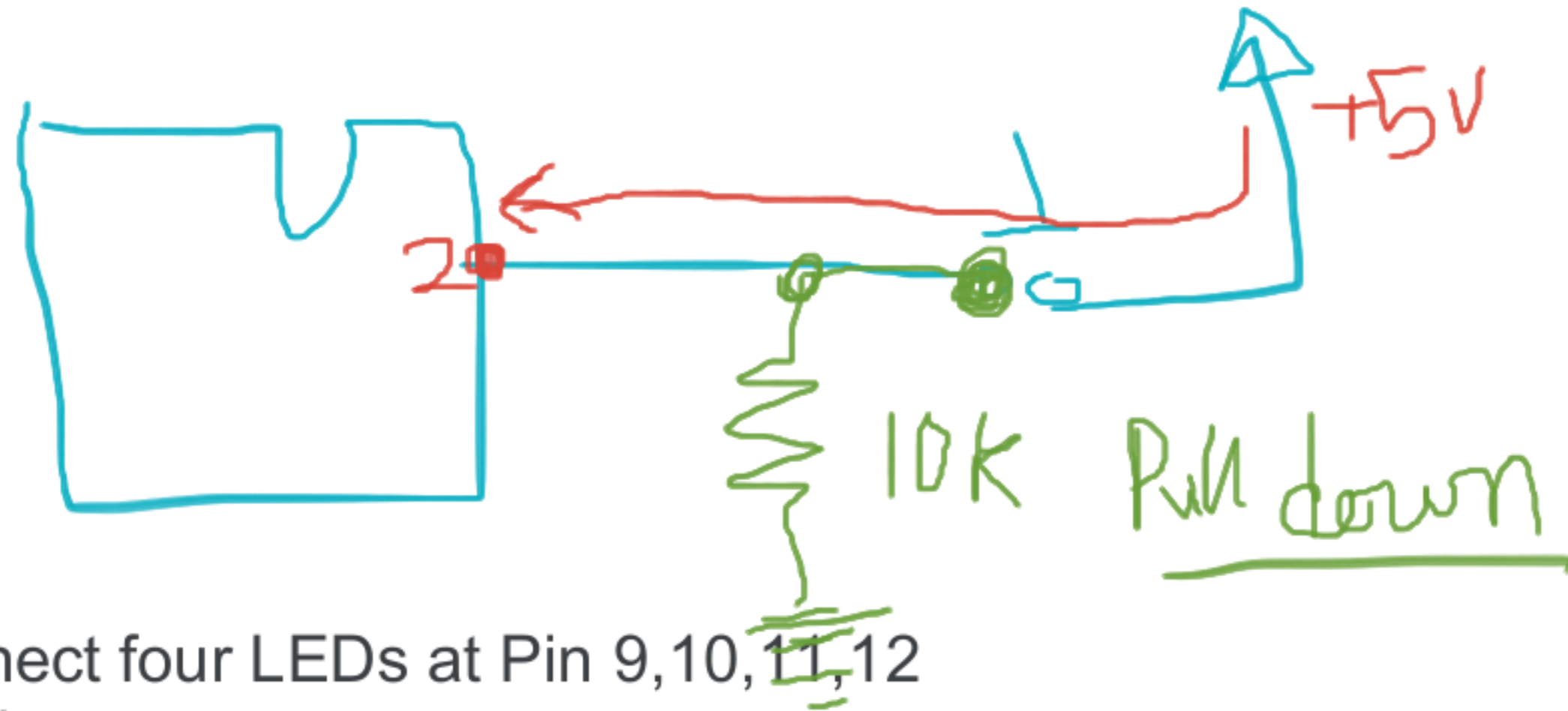


```
for ( int k=0 ; k<4 ; k++){  
    printf(" %d \t" , k);  
}
```

```
int LEDs [] = {13,12,11,10,9};  
  
void setup ()  
{  
    for ( int k=0 ; k<5 ; k++){  
        pinMode(LEDs[k], OUTPUT) ;  
    }  
}  
  
void loop()  
{  
    for ( int k=0 ; k<5 ; k++){  
        digitalWrite(LEDs[k], HIGH);  
        delay(1000);  
    }  
    for ( int k=0 ; k<5 ; k++){  
        digitalWrite(LEDs[k], LOW);  
    }  
    delay(1000);  
}
```



start with 1  
check condition at 2  
if contd ok execute all statements in  
{ block }  
go to incrementing the variable at 3  
check the contd again at 4  
if contd ok execute all statements in  
{ block }  
continue step 3 and 4



c  
see  
sea

Q 1 connect four LEDs at Pin 9,10,11,12  
write code :

- blink all @ delay of 1.5 sec
- first LED is high for 1 sec then second LED goes HIGH after 1 sec third LED goes high ...
- interface button at pin 7 . on push of button all LEDs should glow
- button at pin 7 . First time push of button makes first LED glow , next push makes second glow and so on .
- two buttons at pin 6 (dec) and 7 (inc). when the button at pin 7 is pushed next LED should glow and when button at pin 6 is pressed the latest glowing LED should be put off.  
analogous to volume button of remote control

22 sep

## button toggle

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(7, INPUT);
}
int butStatus = 0;
void loop()
{
  if ( digitalRead(7) == HIGH ) {
    // toggle the value of butstatus
    if (butStatus == 0) butStatus =1;
    else butStatus=0;
    // keep waiting till user leave the button
    while( digitalRead(7) == HIGH ) {}
  }
  if (butStatus ==1 )
  { digitalWrite(13, HIGH);
  }else {
    digitalWrite(13, LOW);
  }
}
```

~~1 sec~~

100ms | 10

arduino uno with wire  
connecting wires  
(m-f, m-m- f-f)  
breadboard  
led , button,  
resistance (330 ohm, 10k)  
buzzer  
LCD , Keypad  
I2c RTC

nodeMCU  
relay module (2relay)  
bluetooth (HC05)



table

2\* 1= 2

2\* 2= 4

2\*3 = 6

....

2\*10 = 20

cal

changin

<https://www.programiz.com/c-programming/online-compiler/>

// Online C compiler to run C program online

#include <stdio.h>

int main() {

// loop to perform repeatative task

int cnt = 1;

while (cnt <=10){

//printf(" %d \t ", cnt);

printf("2 \* %d= %d \n",cnt, cnt\*2);

cnt++;

}

// easy way out instead of while is for

for (int tnt=1; tnt<=10 ; tnt++){

printf("3 \* %d= %d \n",tnt, tnt\*3);

}

return 0;

}

# LCD

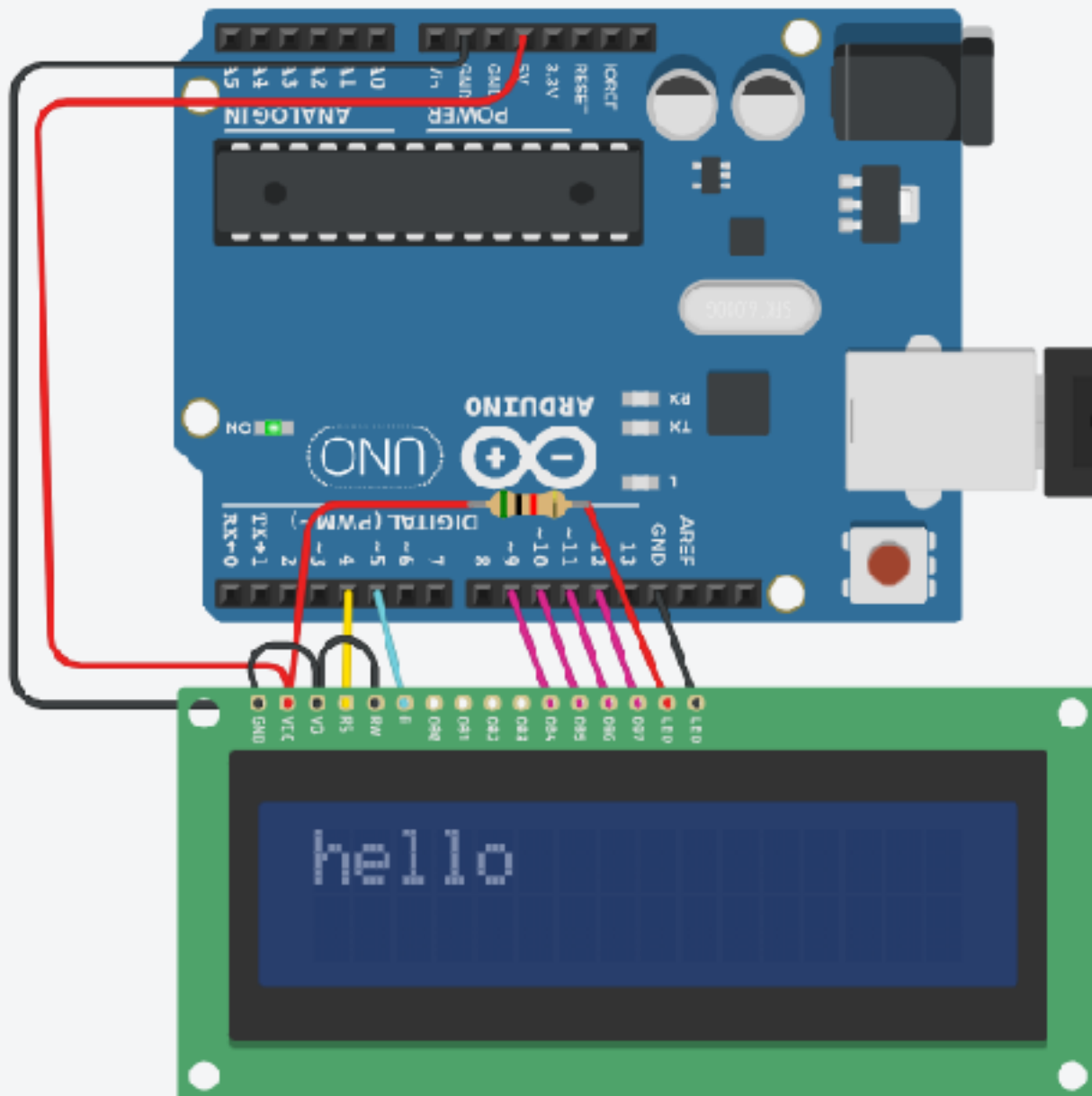
Circuit design Sizzling Bigery x Circuit design Copy of LEDs\_E x CEPTAM\_Embedded\_2 - Goog x CEPTAM\_Embedded\_1 - Goog x PowerPoint Presentation x

https://www.tinkercad.com/things/beA1/4e1c0Ua-sizzling-bigery/editel/tenant-circuits

**TINKERCAD** LCD interfacing All changes saved

Simulator time: 00:00:04 Code Stop Simulation Export Share

1 (Arduino Uno R3)



The circuit diagram shows an Arduino Uno R3 board connected to an LCD display. The connections are as follows:

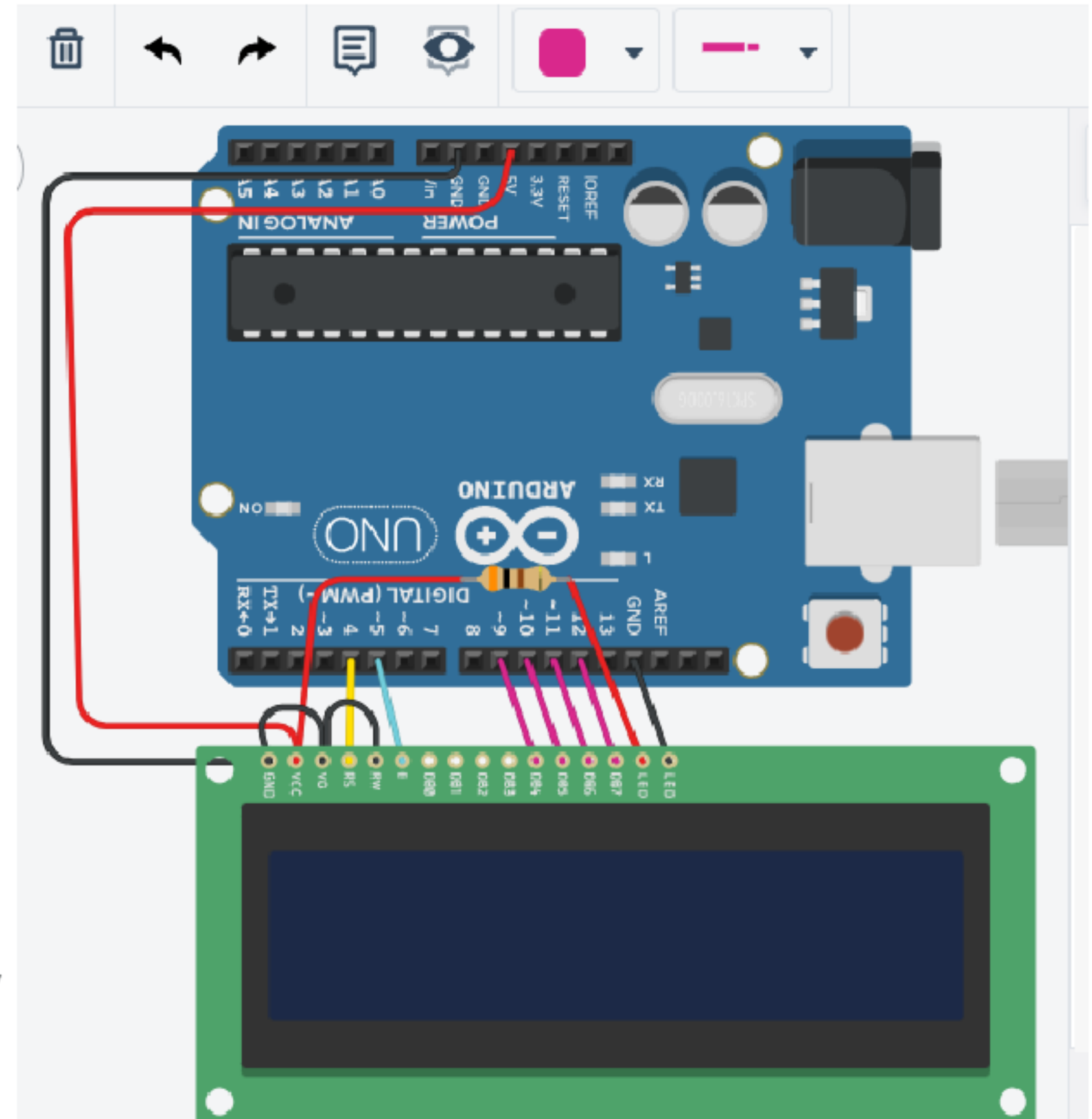
- Red wire: Arduino 5V to LCD VCC
- Black wire: Arduino GND to LCD GND
- Yellow wire: Arduino D4 to LCD D4
- Blue wire: Arduino D5 to LCD D5
- Pink wire: Arduino D6 to LCD D6
- Purple wire: Arduino D7 to LCD D7
- White wire: Arduino D8 to LCD D8
- White wire: Arduino D9 to LCD D9

```
1 #include <LiquidCrystal.h>
2 int rs=4, en=5, d4=9, d5=10, d6=11, d7=12;
3 LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
4
5 void setup()
6 {
7   lcd.begin(16, 2);
8   lcd.print("hello");
9 }
10
11 void loop()
12 {
13
14 }
```

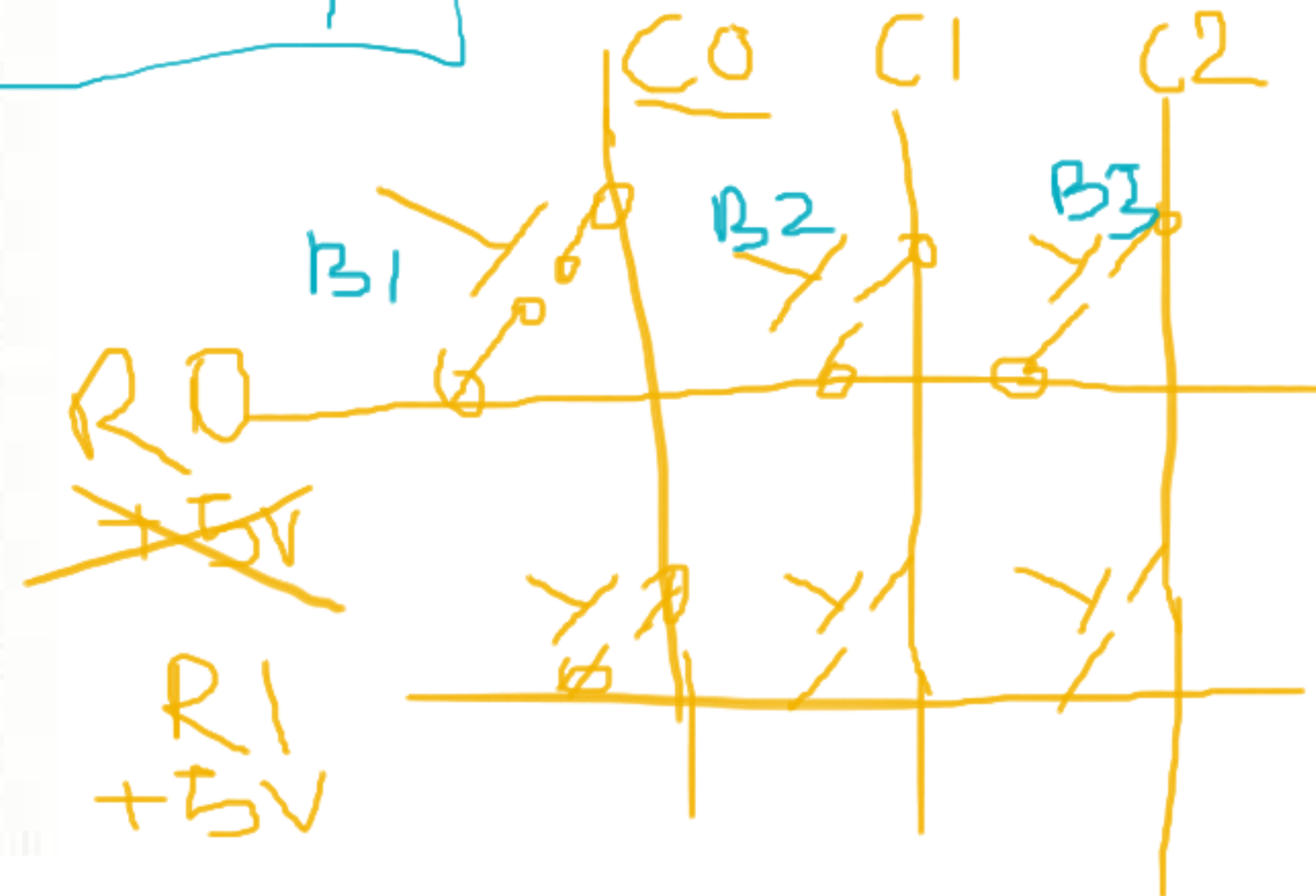
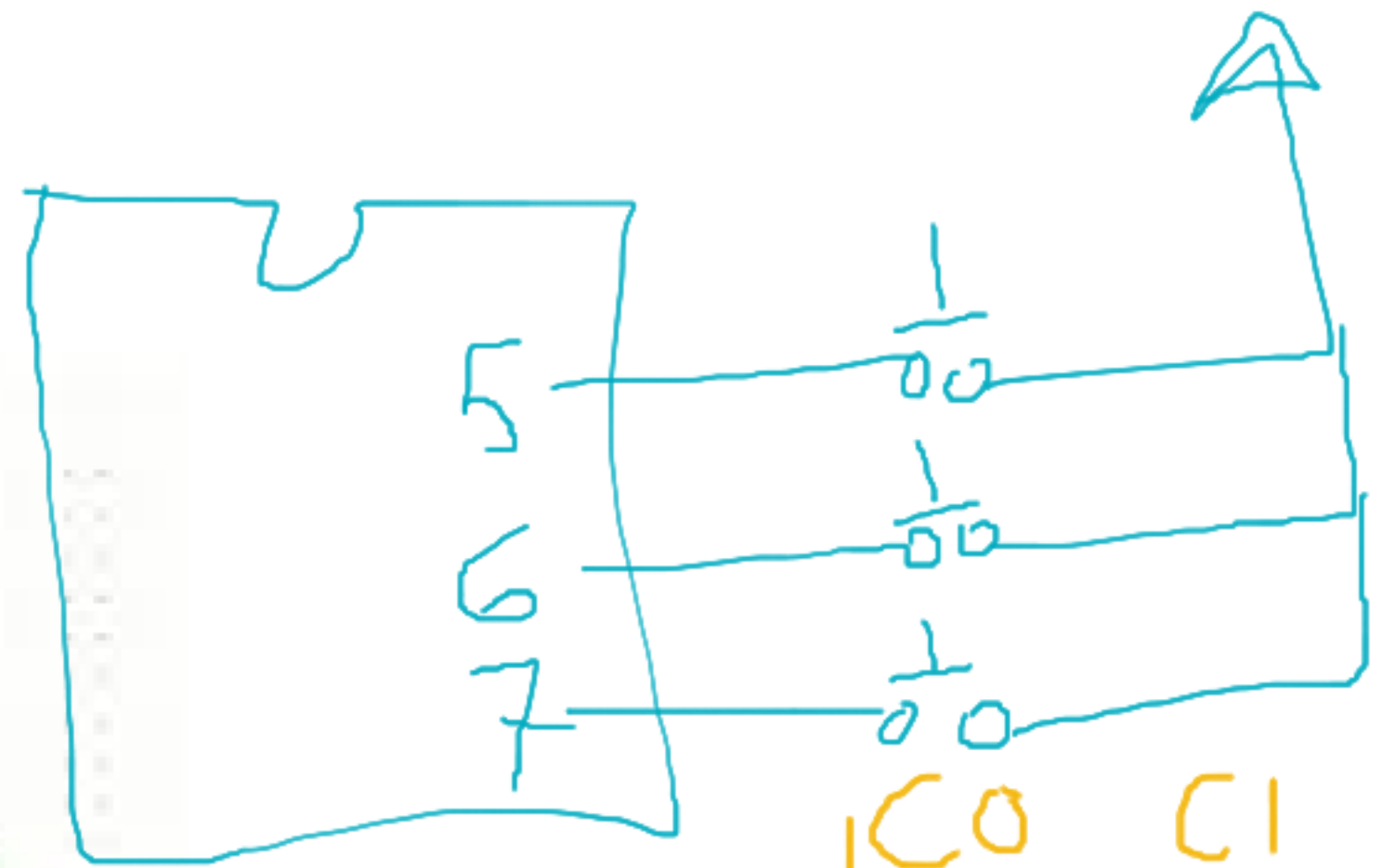
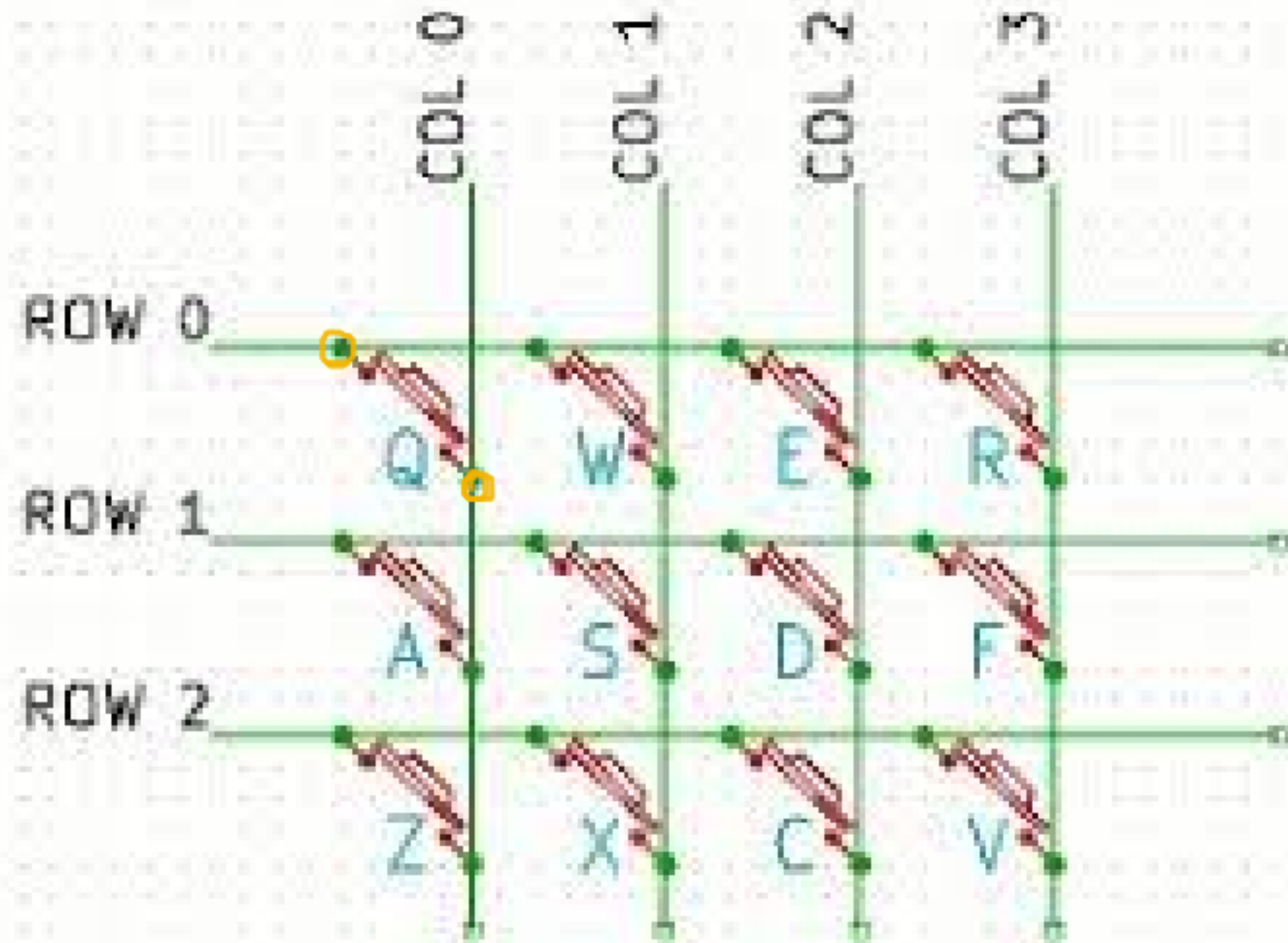
Serial Monitor

Windows taskbar: Type here to search, 28°C Rain, 11:14, 22 09 2021

Pin 1 - grd  
pin 2 - vcc  
pin 3- contrast (gnd)  
pin 4 (RS) = controller  
pin 5 (RW) - GND  
pin 6 (E) - controller  
pin 7 (DB0) -  
pin 8 (DB1) -  
pin 9 (DB2) -  
pin 10 (DB3) -  
pin 11 (DB4) - controller  
pin 12 (DB5) - controller  
pin 13 (DB6) - controller  
pin 14 (DB7) - controller  
pin 15 (backlight) through resistance +5v  
pin 16 (backlight) gnd

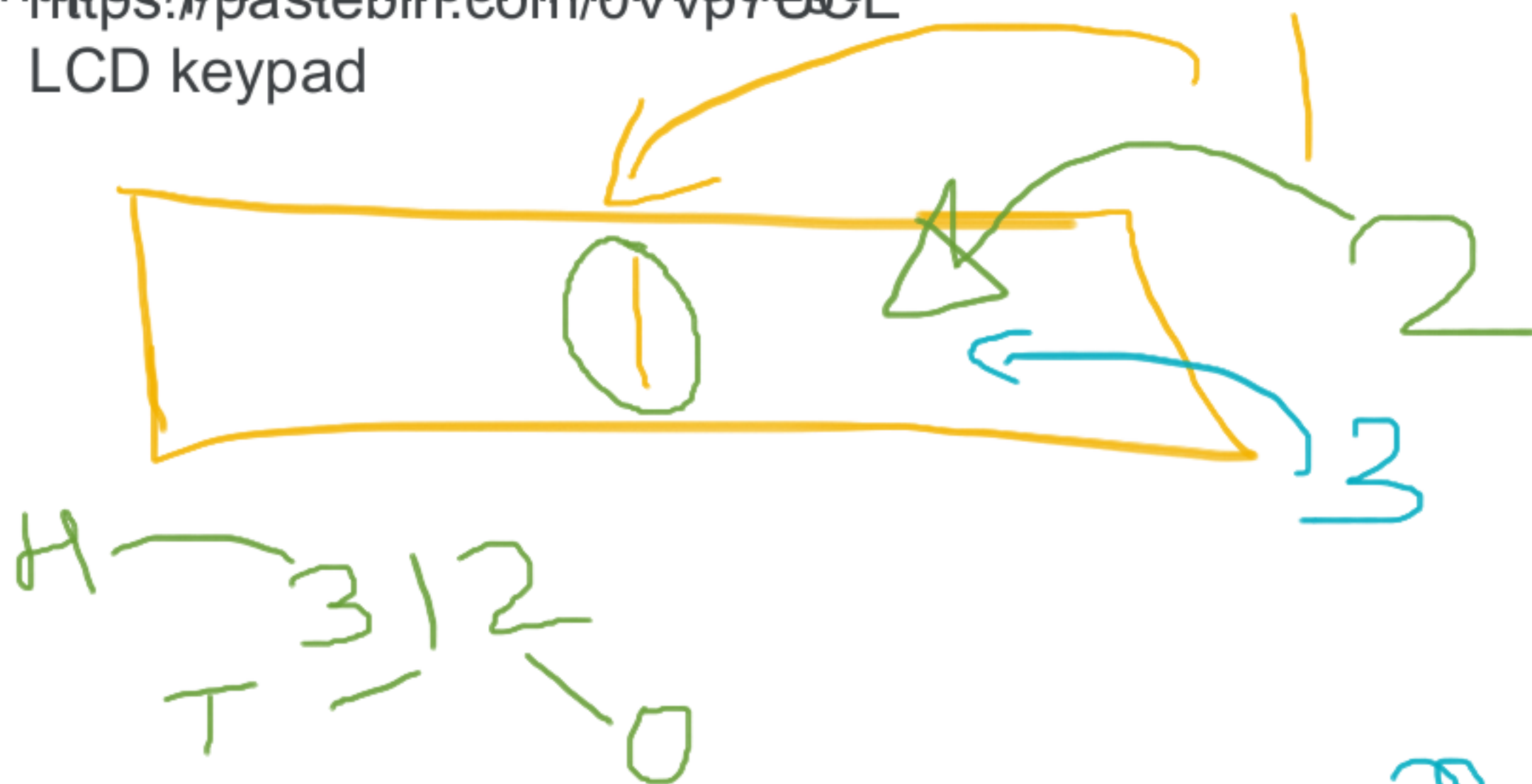


keypad





<https://pastebin.com/nNwQazgc>  
<https://pastebin.com/0Vvp7UCE>  
LCD keypad



$$n = 1$$

$$n = n \times 10 + \text{new no}$$

$$n = 1 \times 10 + 2 \\ = 12$$

$$n = n \times 10 + 3 \\ = 12 \times 10 + 3 \\ = \underline{\underline{123}}$$



processor - 133Mhz

RAM - 16 MB

HDD - 1.2GHz

50-60k



~ 1 GHz



male / female berg

|                                     | <b>I<sup>2</sup>C</b>               | <b>SPI</b>                  |
|-------------------------------------|-------------------------------------|-----------------------------|
| <b>Originator</b>                   | Philips (1982)                      | Motorola (1979)             |
| <b>Plug &amp; Play</b>              | Yes                                 | No                          |
| <b>Interface type</b>               | Serial (2 wires)                    | Serial (3+N wires)*         |
| <b>Distance</b>                     | Short (In-box communication)        |                             |
| <b>Application</b>                  | Multi-master register-access        | Transfer of data-streams    |
| <b>Protocol Complexity</b>          | Low                                 | Lower                       |
| <b>Design Cost</b>                  | Low                                 | Lower                       |
| <b>Transfer rate</b>                | Limited (100 & 400 KHz and 3.4 MHz) | Free (n x MHz to 10n x MHz) |
| <b>Power Consumption</b>            | Low<br>(2 pull-up resistors)        | Lower                       |
| <b>Transfer type</b>                | Half Duplex                         | Full Duplex                 |
| <b>Time Constraint</b>              | Synchronous                         |                             |
| <b>Multi Master<sup>+</sup></b>     | Yes                                 | No                          |
| <b>Multi Slave</b>                  | Yes                                 | Yes                         |
| <b>I/O constraints</b>              | Open-drain with pull-up resistors   | No constraint               |
| <b>Addressing<sup>+</sup></b>       | Software (7/10 bits)                | Hardware (Chip Select)      |
| <b>Flow Control<sup>+</sup></b>     | Yes                                 | No                          |
| <b>Clock Stretching<sup>+</sup></b> | Yes                                 | No                          |

\* : N is the number of devices connected to a single master on the bus.

+ : Feature inducing substantial area overhead.

23 sept

## Security lock system using keypad lcd

LCD - "welcome" 2s

LCD - "Enter password" 1 line

\*\*\*\*

2 line

✓ user press # to check password

if the password entered by user matches with save pwd then

green LED

else show

LCD - "wrong password" 1 line

and accept the pwd for second time

every time password is incorrect buzzer goes on

and red LED is on for 1 sec

user can try for three times only

after that

LCD - "System lock"

LCD

```
int rs=8, en=13, d4=9, d5=10, d6=11, d7=12;
```

```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

keypad

```
int r1=0, r2=1, r3=2, r4=3, c1=4, c2=5, c3=6, c4=7;
```

Buzzer 14

LED red 15, green 16

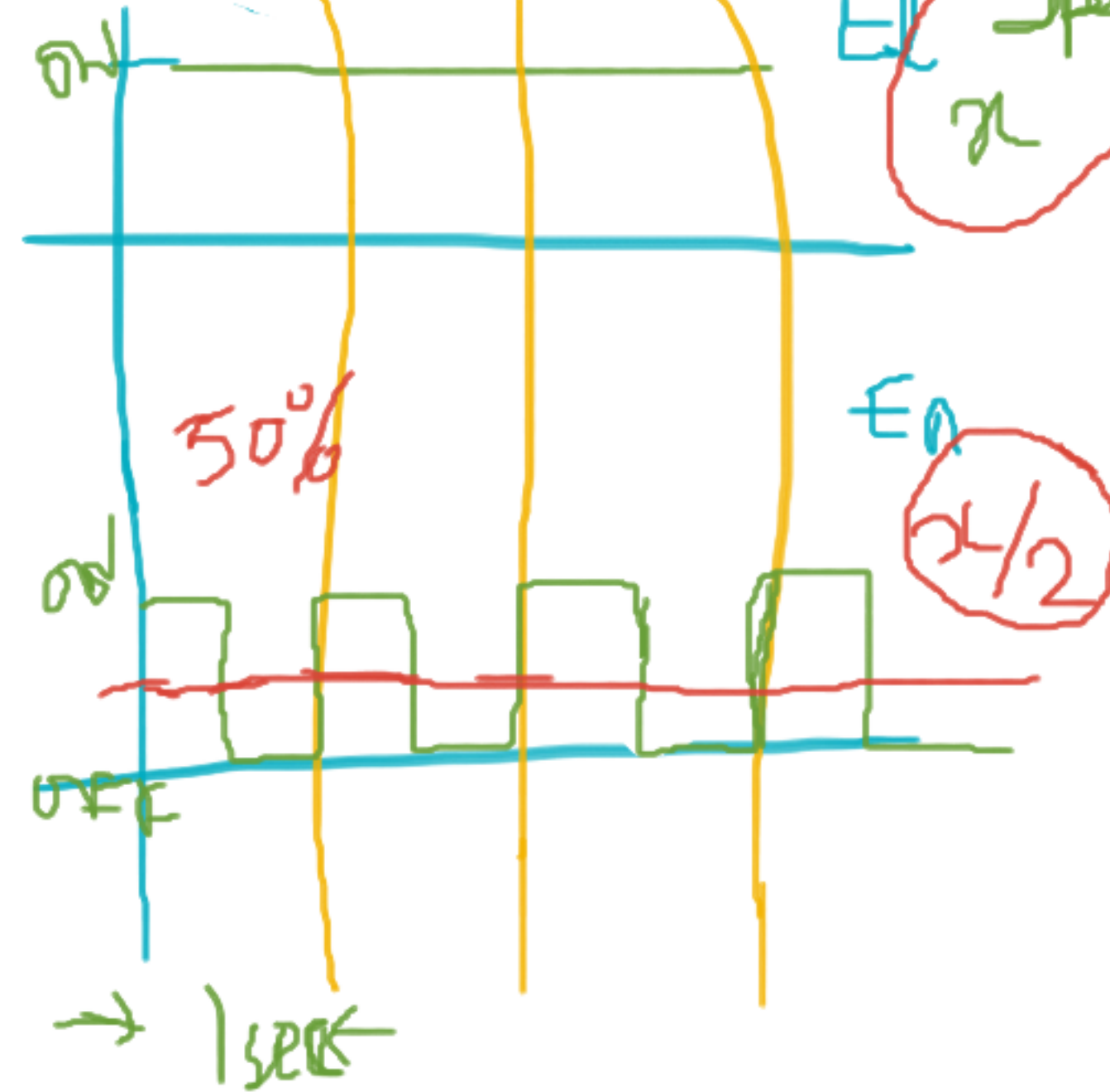
3 times



# fan regulator

PWM

Pulse width modulation



Electronic  
step by step inc/dec  
diac/triac ( diode )

long life  
smooth performance  
power is efficiently used

input power is directly  
proportional to load/speed of fan

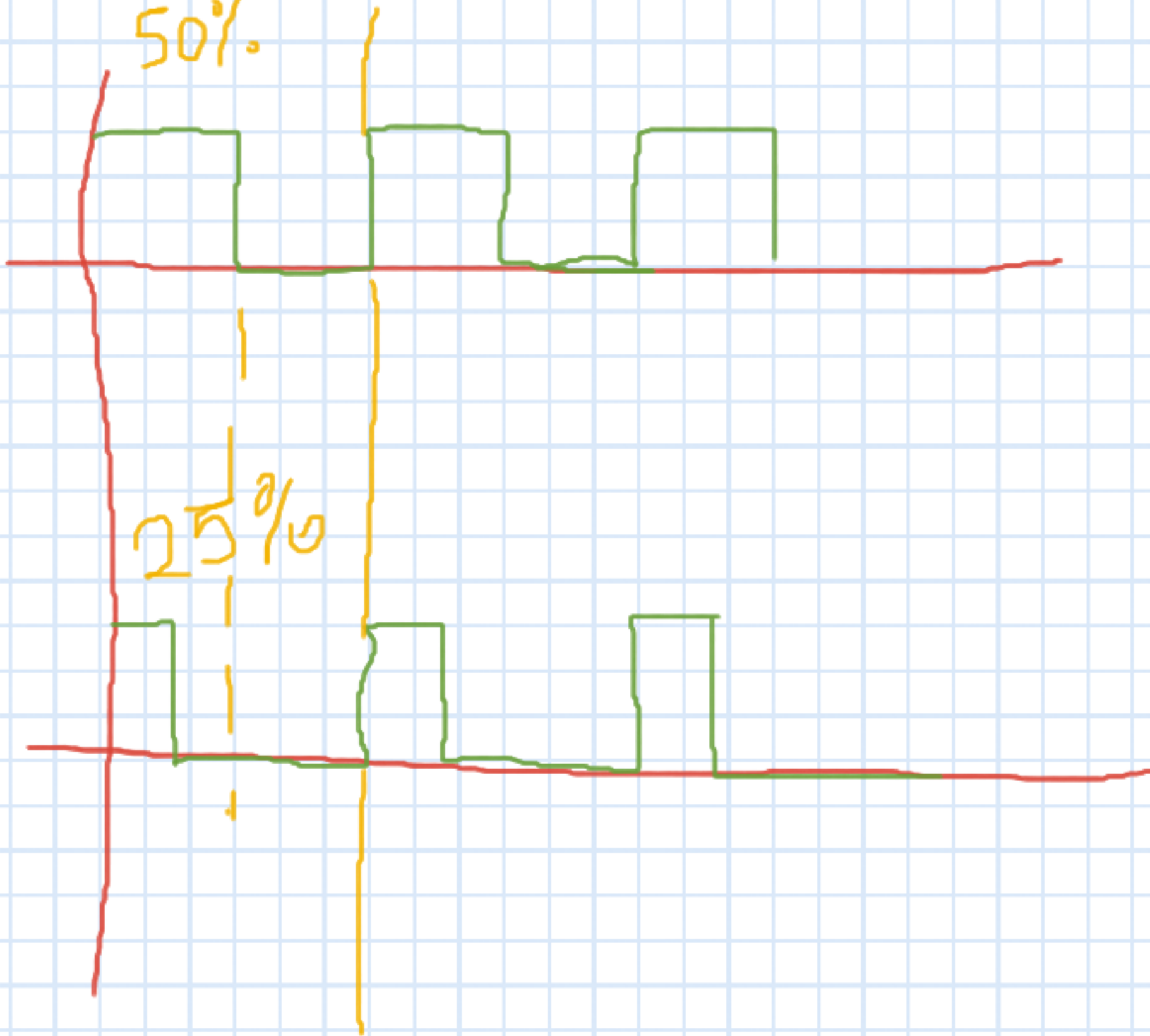
Electrical  
big resistance  
Losses are high

the decrease in speed of  
fan leads to high losses/  
heat, but input power is  
same

50%

PWM

25%



analog input

10 bits

min = 0

max = 1023

all analog pin in arduino  
uno has 10 bit resolution  
input can be calibrated on  
the scale of 0-1023

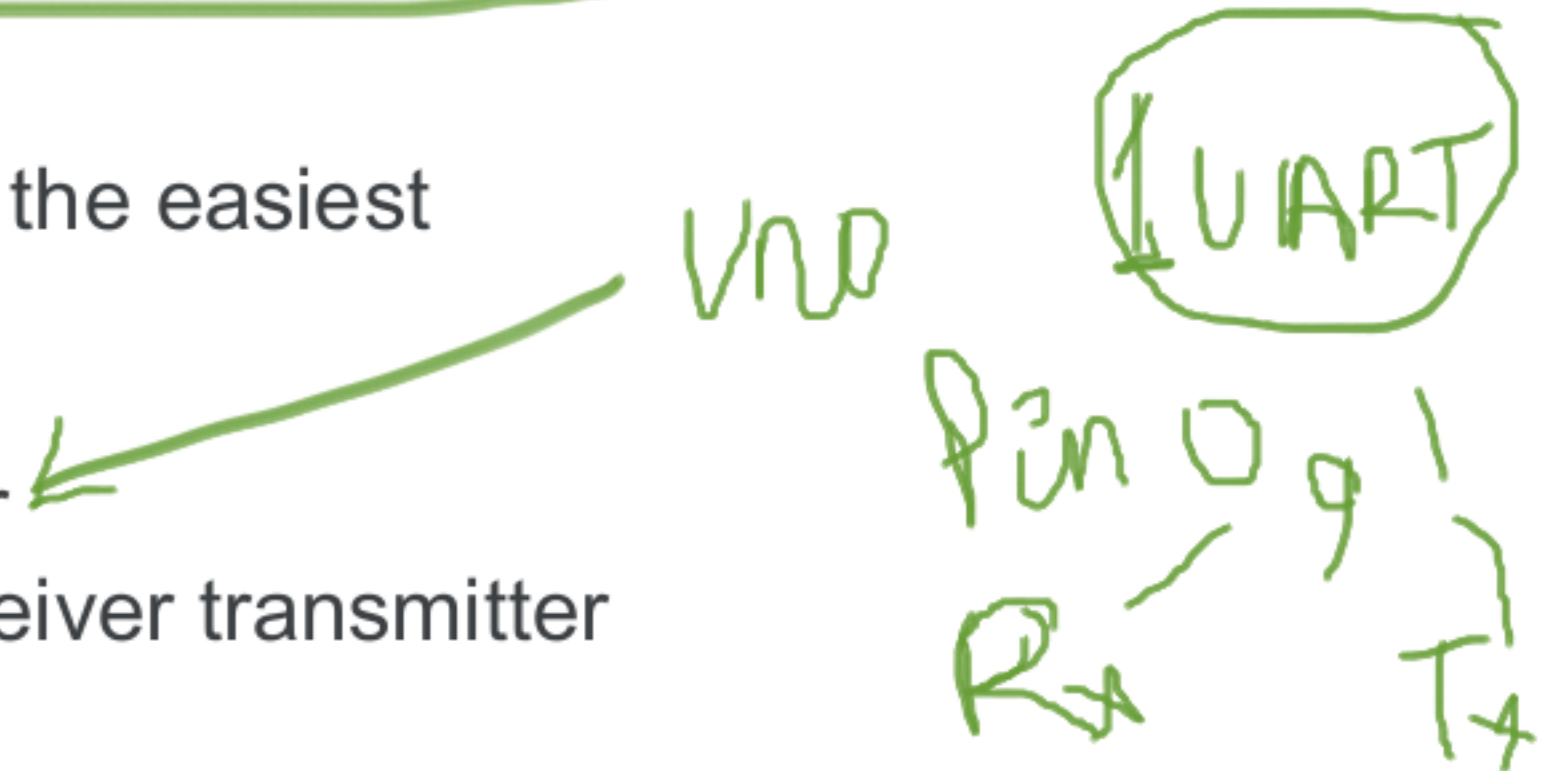


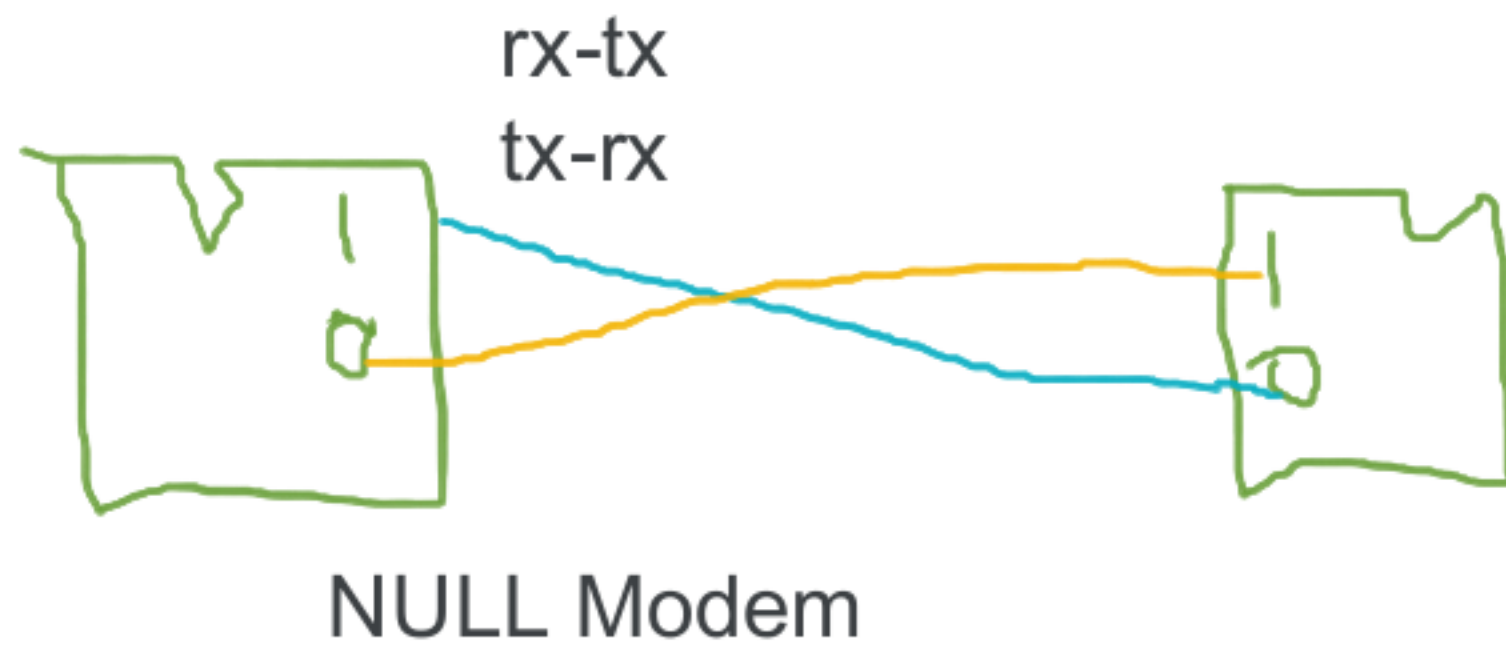
### Serial communication

when two devices want to talk to each other, one of the easiest  
way is serial communication  
generally using

UART - universal asynchronous receiver transmitter

USART - universal synchronous asynchronous receiver transmitter





arduino uno can push the value read from sensor to PC using Serial communication at specific baud rate

\* Another library in ARduino platform makes any pin to be used as RX, TX  
SoftwareSerial

this makes serial communication through software/programming

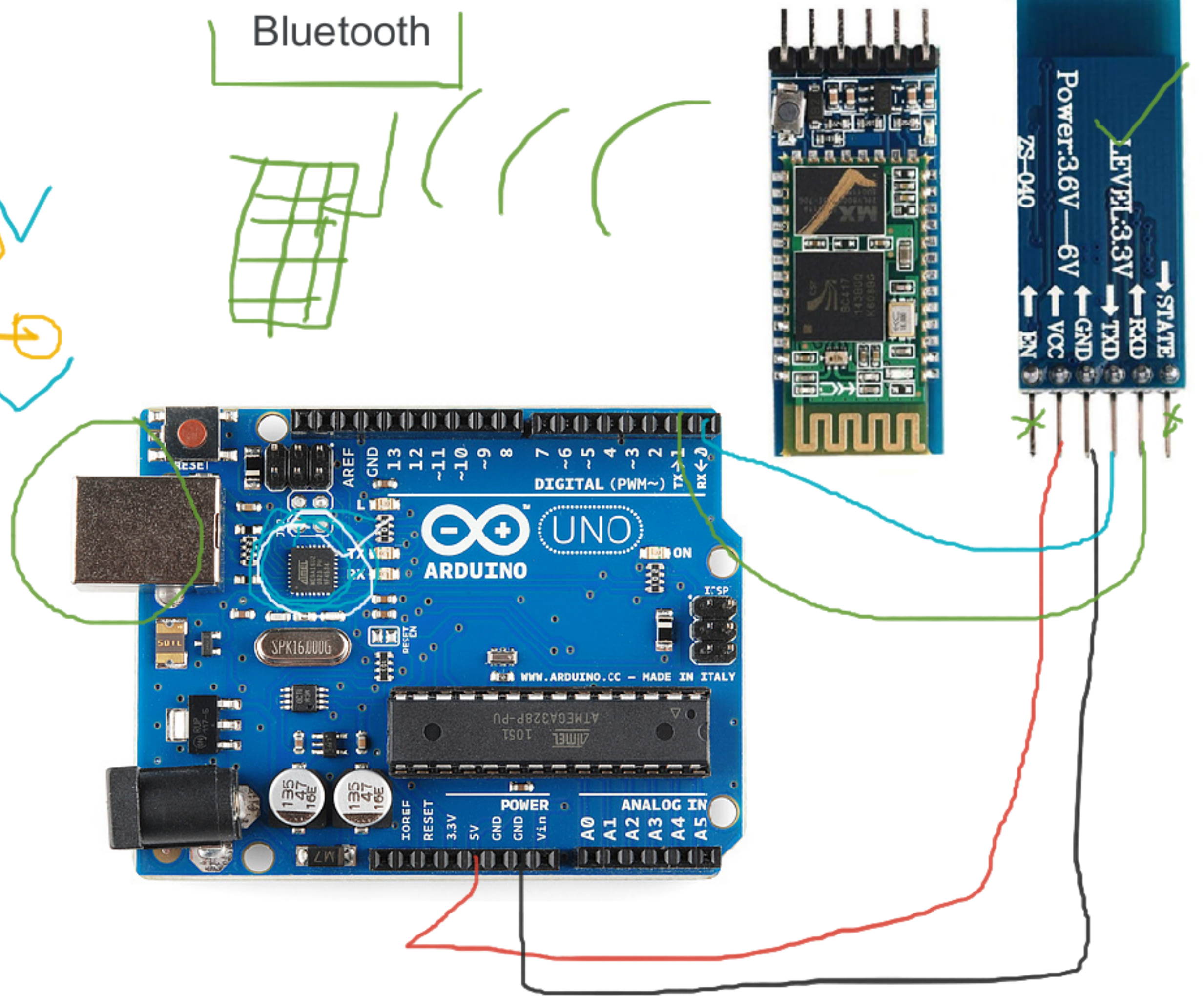
baud rate is signalling rate which is different from data rate



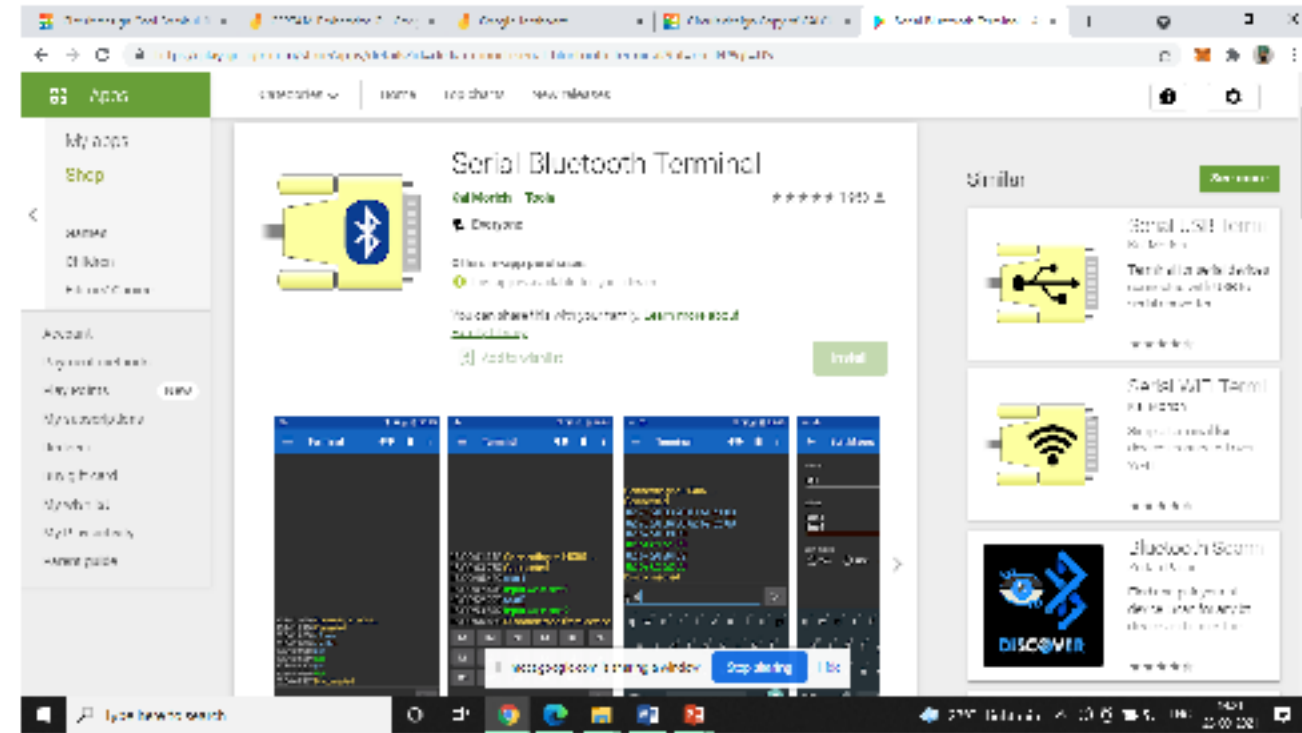
## Bluetooth

bluetooth  
HC05

## Modem AT commands



# mobile app



# arduino Program

```
int LED=13;
void setup()
{
  pinMode(LED , OUTPUT);
  Serial.begin(9600);
}
char v1=' ';
void loop()
{ // Serial.read () return the ASCII code
  if (Serial.available()){
    v1= (char) Serial.read();
    Serial.println(v1);
    delay(500);
  }
  if (v1=='1') { digitalWrite(LED , HIGH); }
  if (v1=='2') { digitalWrite(LED , LOW); }
}
```



