

Trabajo Práctico N°1: Manejo de Puertos

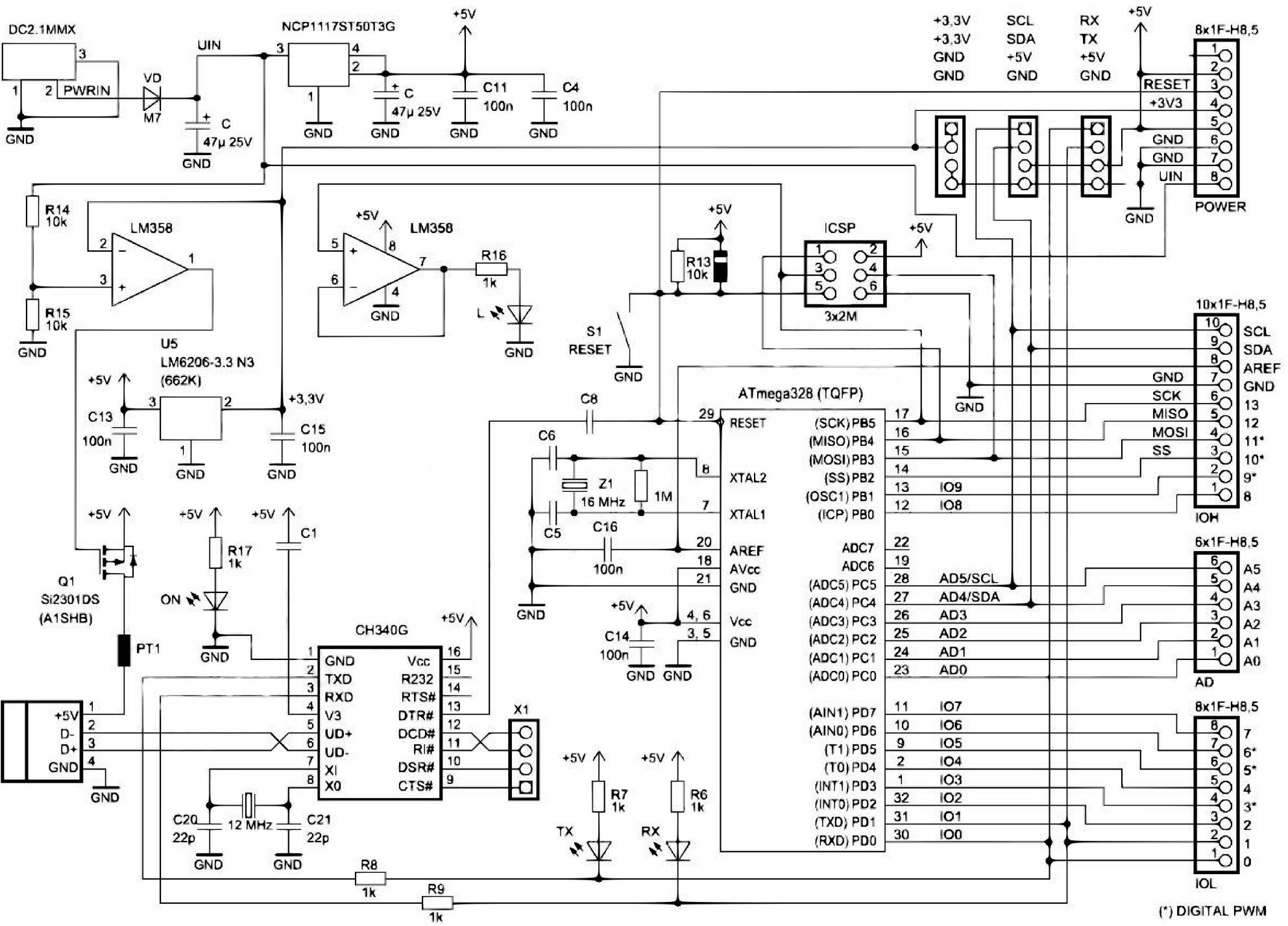
Verónica Romina Bustamante -988640- verobustamante95@gmail.com - vbustamante@fi.uba.ar

Ejercicio 1

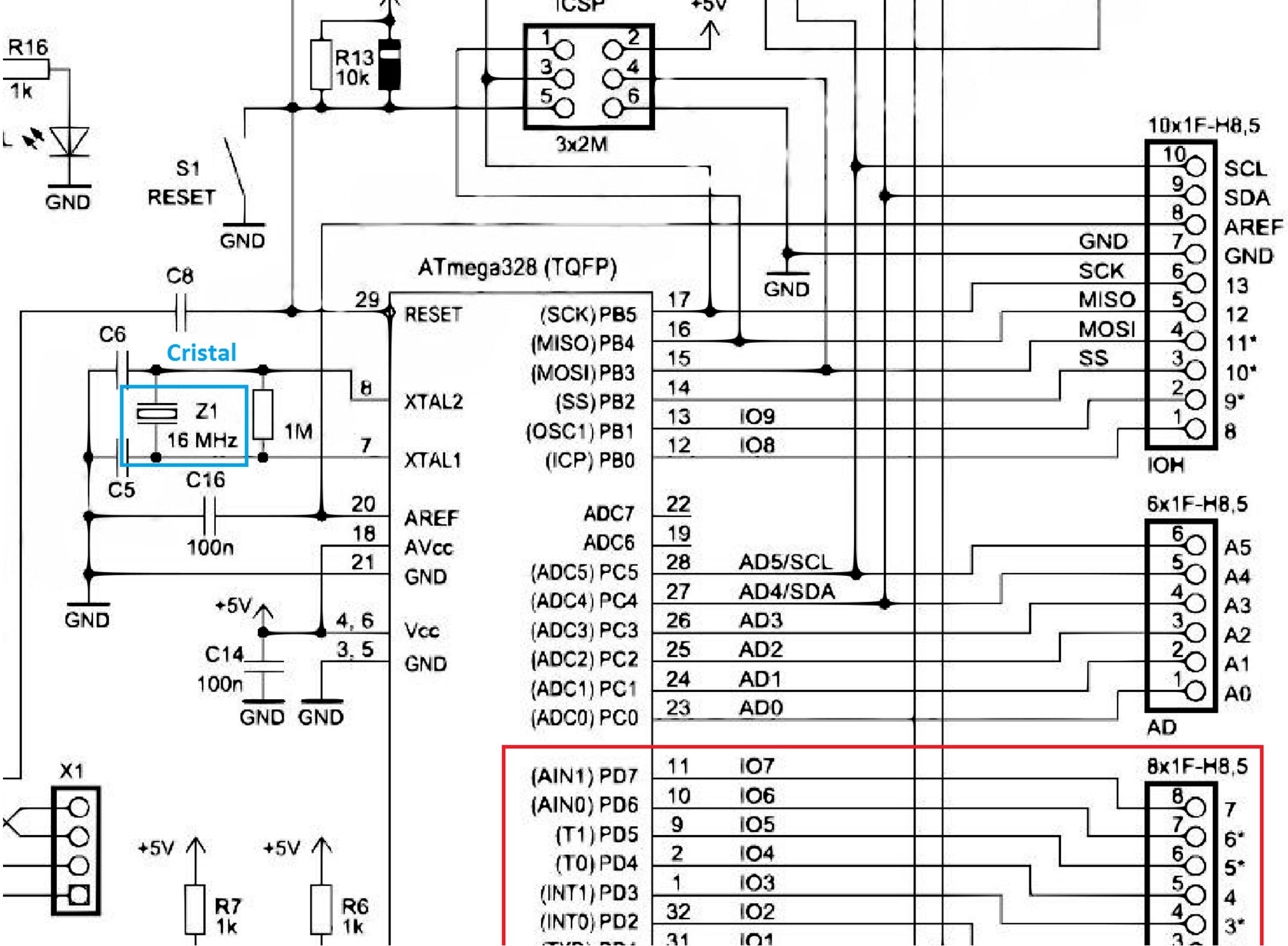
Hacer un programa que haga parpadear un LED conectado en el PIN 2 (Usar la rutina de retardo Dada)

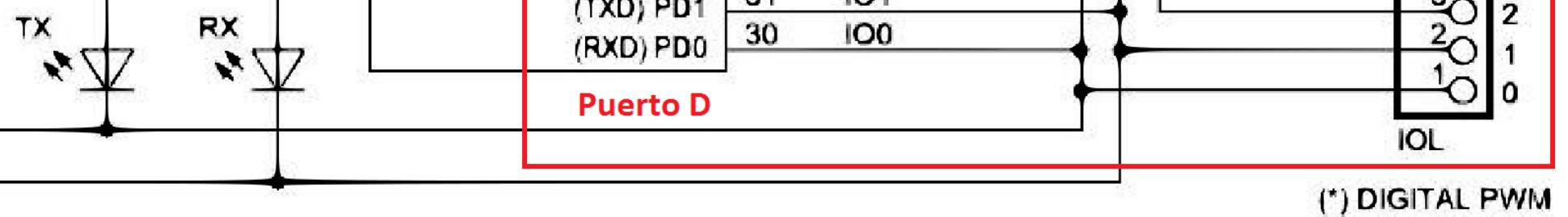
Resolución:

Para el ejercicio nos pide que utilicemos un LED conectado en el PIN 2 del Arduino, el Arduino cuenta con el siguiente esquemático:



En el esquemático puede observarse el ATMega38p:





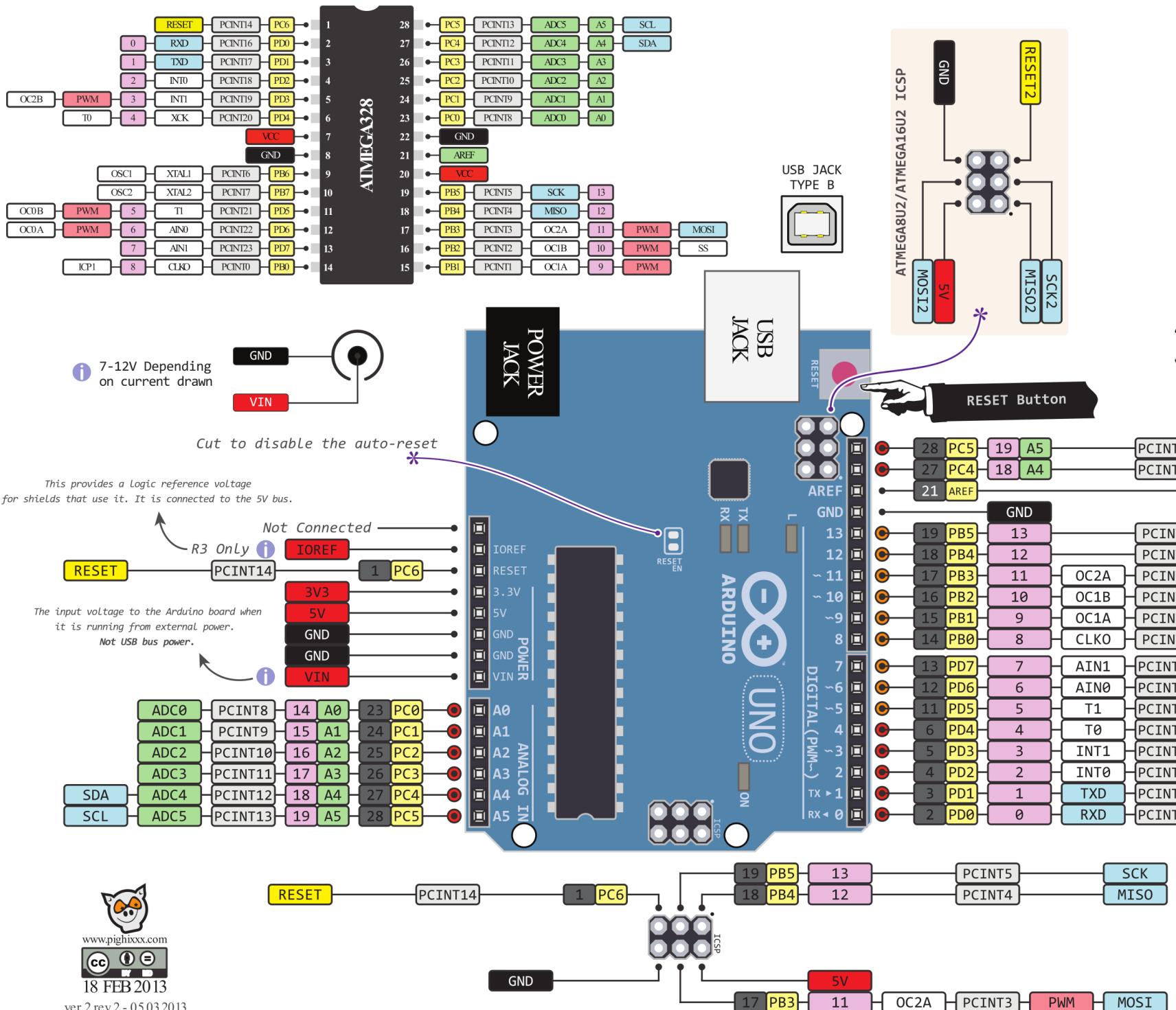
El enunciado pide en el pin 2, ahora, esto puede corresponder al pin 2 del puerto o al pin 2 físico como puede observarse en la siguiente imagen:

THE
DEFINITIVE

ARDUINO

UNO

PINOUT DIAGRAM



www.pighixxx.com



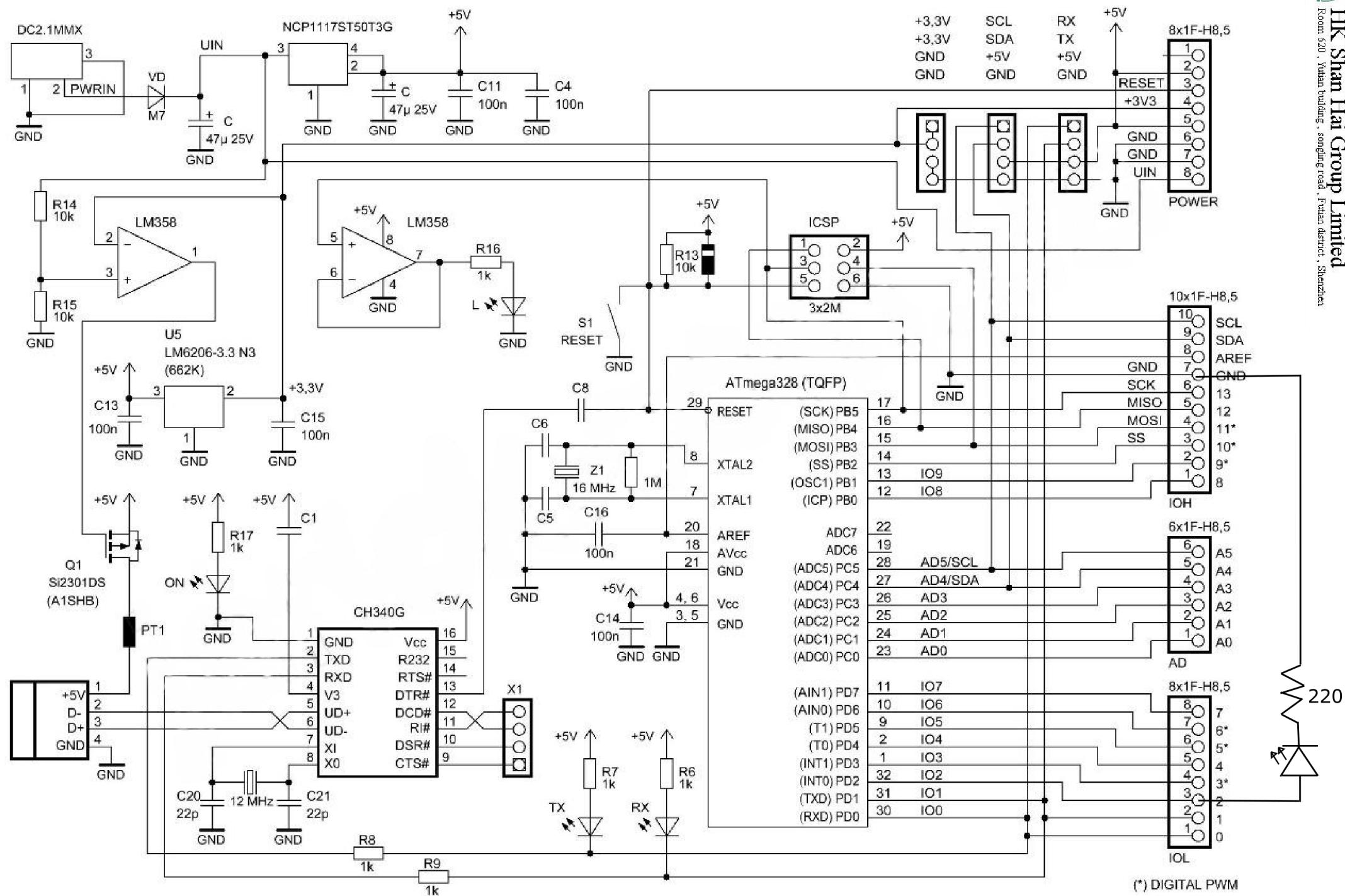
18 FEB 2013

ver 2 rev 2 05.03.2013

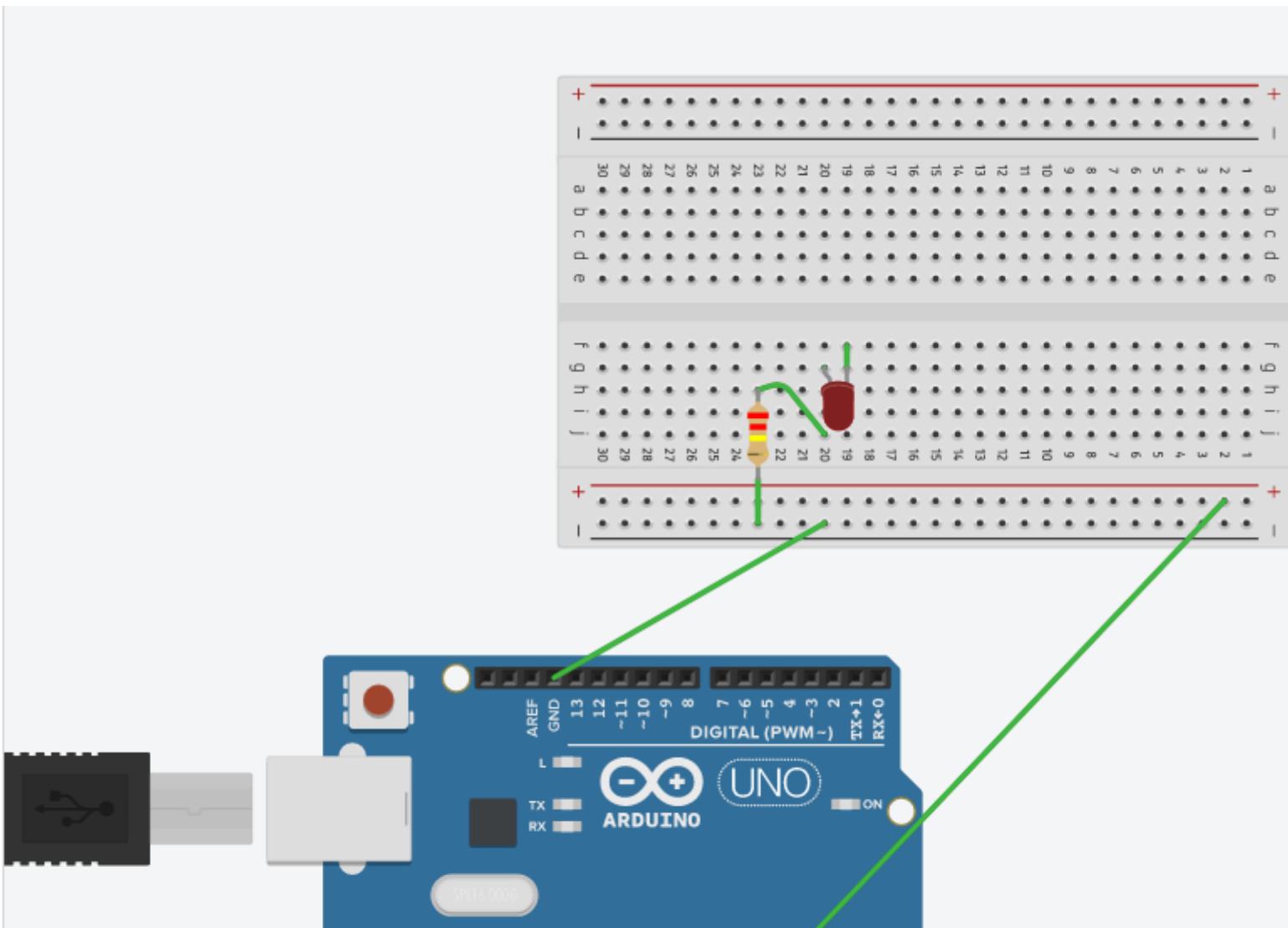
Interpreto que debe referirse al PIN2 del puerto dado que el otro pin corresponde al Rx y aunque es utilizable supone desconectarlo y volverlo a conectar para que el programa funcione, lo que para el enunciado no resulta de complejidad pero si se trabaja con circuitos más complicados puede representar un problema dado que hay que tener cuidado con el resto del circuito.

El circuito para esta práctica fue realizado en protoboard. El ensamblador usado es el del AVRStudio 5. El Arduino posee solo un micro, el ATMega328P, el otro es un integrado CH340G.

El esquemático para el circuito es el siguiente:



Podemos ver un diagrama a continuación:



```
/* * titilarled.asm * * Created: 09/05/2021 08:24:17 p.m. * Author: Vero */ .INCLUDE "M32DEF.INC" .equ BITOUT=2 ;Configuro PD0 como salida LDI R21, 0xFF OUT DDRD, R21 ;Titilar MAIN: SBI PORTD,BITOUT ;set bit PD0 CALL DELAY CBI PORTD,BITOUT CALL DELAY CALL MAIN ;Delay de 1s (modificado) DELAY: LDI R18, 80 LDI R19, 255 LDI R20, 255 LOOP: dec R20 brne LOOP dec R19 brne LOOP dec R18 brne LOOP RET
```

Puede observarse el funcionamiento del programa en los siguientes videos:

- https://youtu.be/l4ITi_VaJGY (https://youtu.be/l4ITi_VaJGY)
- https://youtu.be/RltYo_ZMgg (https://youtu.be/RltYo_ZMgg)

```
In [15]: from IPython.display import HTML
```

```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/l4ITi_VaJGY" frameborder="0" allowfullscreen></iframe>')
```

Out[15]:

Titilar led



```
In [17]: from IPython.display import HTML
```

```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/RltYo_ZMsgg" frameborder="0" allowfullscreen></iframe>')
```

Out[17]:

led parpadea a 1 segundo



Analicemos la cantidad de operaciones:

$$(255 * 3 + 1) * 255 + 1) * 82 + 1 = 16017061$$

Y si el cristal es de 16MHZ y todos los fuses están en 0, el micro estará funcionando a esa frecuencia. Las operaciones para la rutina serán realizadas en:

$$\text{operaciones por segundo} = \frac{16017061 \text{ op}}{16 \text{ MHz}} = 1,001 \text{ op/s}$$

En la práctica esto no ocurrió, se tardaba dos segundos, por lo tanto entendemos que el ATMega328P está funcionando a 8MHz y para que fuese consistente en el código debía de cambiarse LDI R18, 80 por LDI R18, 41 para reducir la frecuencia ahora que se sabía que el microprocesador no trabaja a 16 MHz:

$$\begin{aligned}\text{operaciones} &= (255 * 3 + 1) * 255 + 1) * 40 + 1 = 16017061 \text{ op} \\ \text{operaciones por segundo} &= \frac{16017061 \text{ op}}{16 \text{ MHz}} = 1,001 \text{ op/s}\end{aligned}$$

En el enunciado original teníamos la siguiente rutina de retardo:

```
In [ ]: ; Assembly code
; Delay 8 000 000 cycles
; 1s at 8.0 MHz
ldi r18, 41
ldi r19, 150
ldi r20, 128
L1: dec r20
brne L1
dec r19
brne L1
dec r18
brne L1
```

Al estar R18,R19 y R20 definidas fuera del loop pero no redefinidas dentro del loop se puede obtener como conclusión que van a decrecer la primera vez en 150 y 128 pero luego decrecerán en 255. Esto se puede comprobar realizando un debugging paso a paso en el AVRStudio. Se puede ver estos videos donde se realizó dicha experiencia:

- https://youtu.be/_Asz7_CWNpM (https://youtu.be/_Asz7_CWNpM).
- <https://youtu.be/iTOoSrvxweE> (<https://youtu.be/iTOoSrvxweE>)

```
In [22]: from IPython.display import HTML
```

```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/_Asz7_CwNpM" frameborder="0" allowfullscreen></iframe>')
```

Out[22]:

Funcionamiento retardo de 1s



In [23]: `from IPython.display import HTML`

```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/iT0oSrvxweE" frameborder="0" allowfullscreen></iframe>')
```

Out[23]:

Funcionamiento Loops anidados



Ejercicio 2

Modificar programa para que prenda un LED cuando se presiona el pulsador 1 y quede parpadeando hasta que se apague cuando se presiona el botón 2. El LED está conectado a un pin del microcontrolador y los pulsadores a otros dos pines.

Ejercicio 3

Cómo modifica el circuito y el programa, para usar la resistencia de pullup interna de los ports al conectar el pulsador. ¿Se ahorra algún componente?

Resolución de Ejercicios 2 y 3:

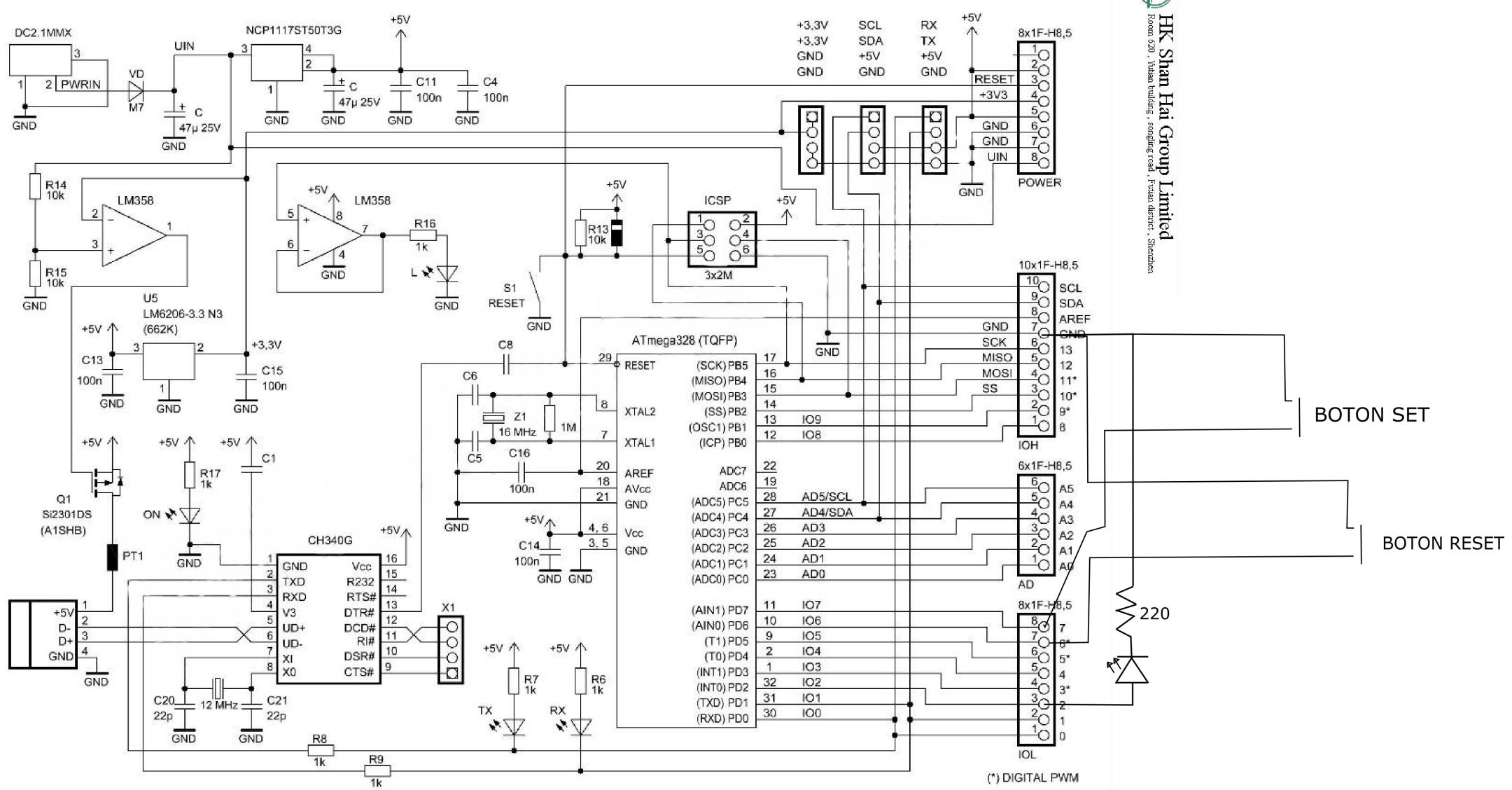
Analicemos la cantidad de operaciones:

$$(255 * 3 + 2) * 255 + 2) * 80 + 1 = 15646961$$

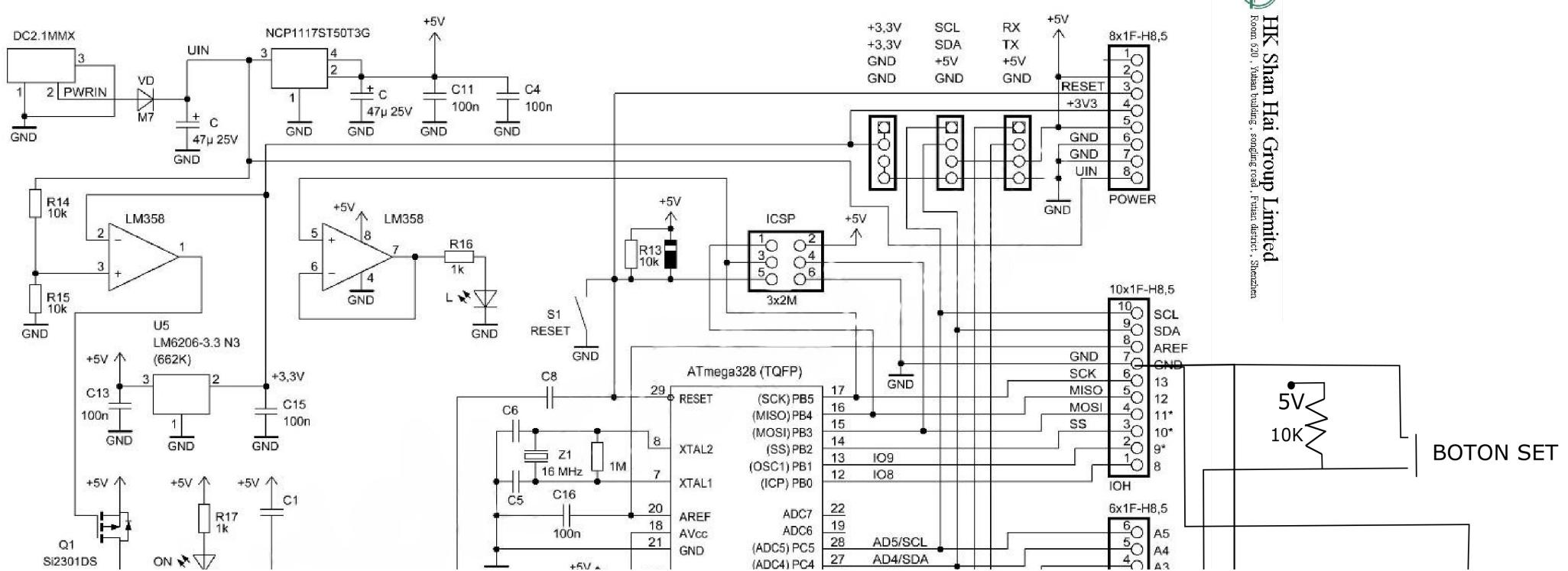
Y si el cristal es de 16MHZ y todos los fuses están en 0, el micro estará funcionando a esa frecuencia. Las operaciones para la rutina serán realizadas en:

$$\text{operaciones por segundo} = \frac{15646961 \text{ op}}{16 \text{ MHz}} = 1,002 \text{ op/s}$$

El esquemático con las resistencias Pull up activadas es el siguiente:



El esquemático sin las resistencias activadas es el siguiente:



.equ LEDPIN=2 .equ SETPIN=7 .equ RESETPIN=6 .equ USE_PULLUP=1 ;1 es para usar resistores internos ;Configuro el puerto donde esta el led como salida LDI R21, (1<

Videos de la experiencia realizada en los puntos 2 y 3

```
In [19]: from IPython.display import HTML
```

```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/NyT-s4BfZp4" frameborder="0" allowfullscreen></iframe>')
```

Out[19]:

Titilar Led con botones, con Rpull up y sin Rpull up



```
In [20]: from IPython.display import HTML
```

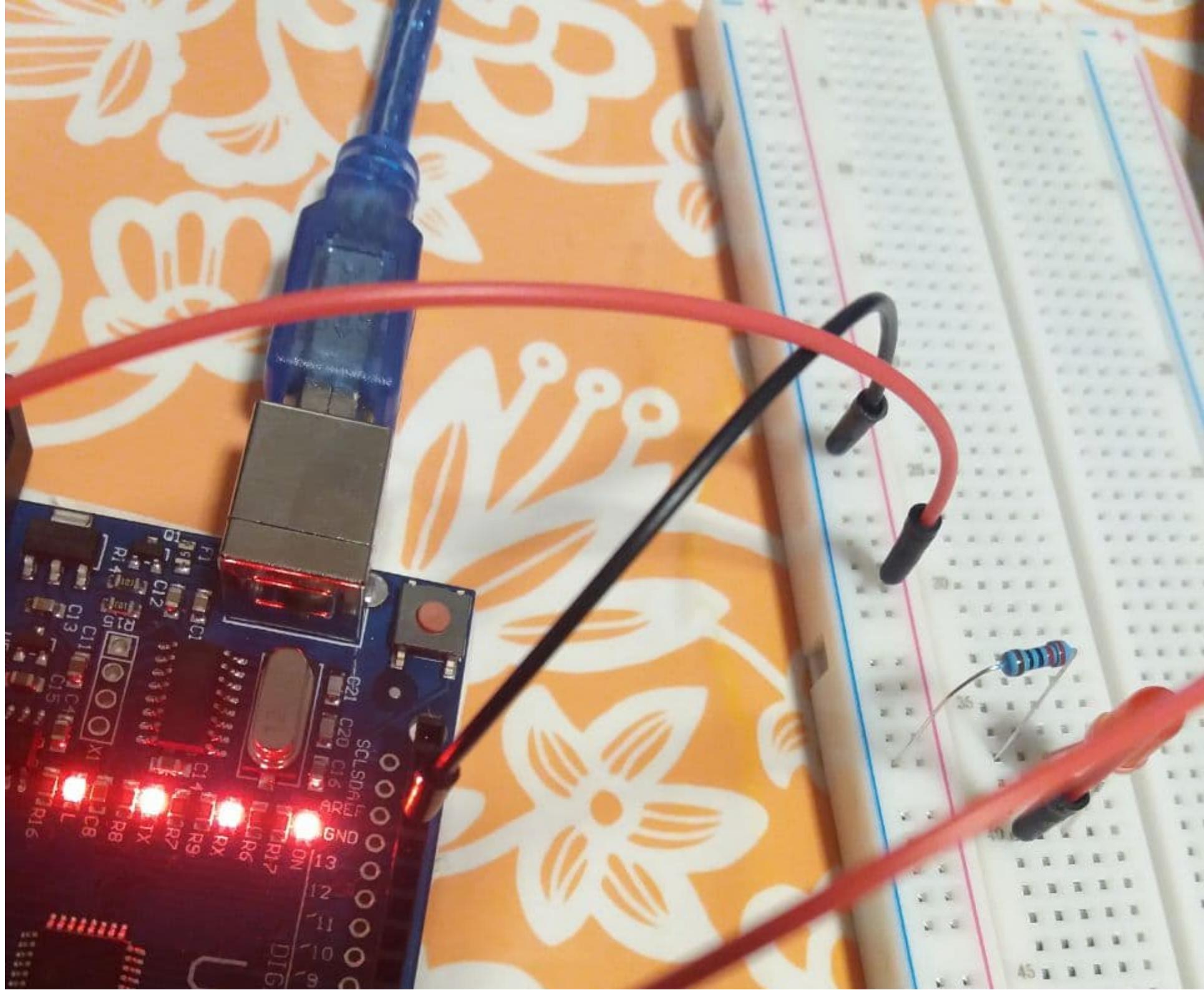
```
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/jcdR-Bd0S04" frameborder="0" allowfullscreen></iframe>')
```

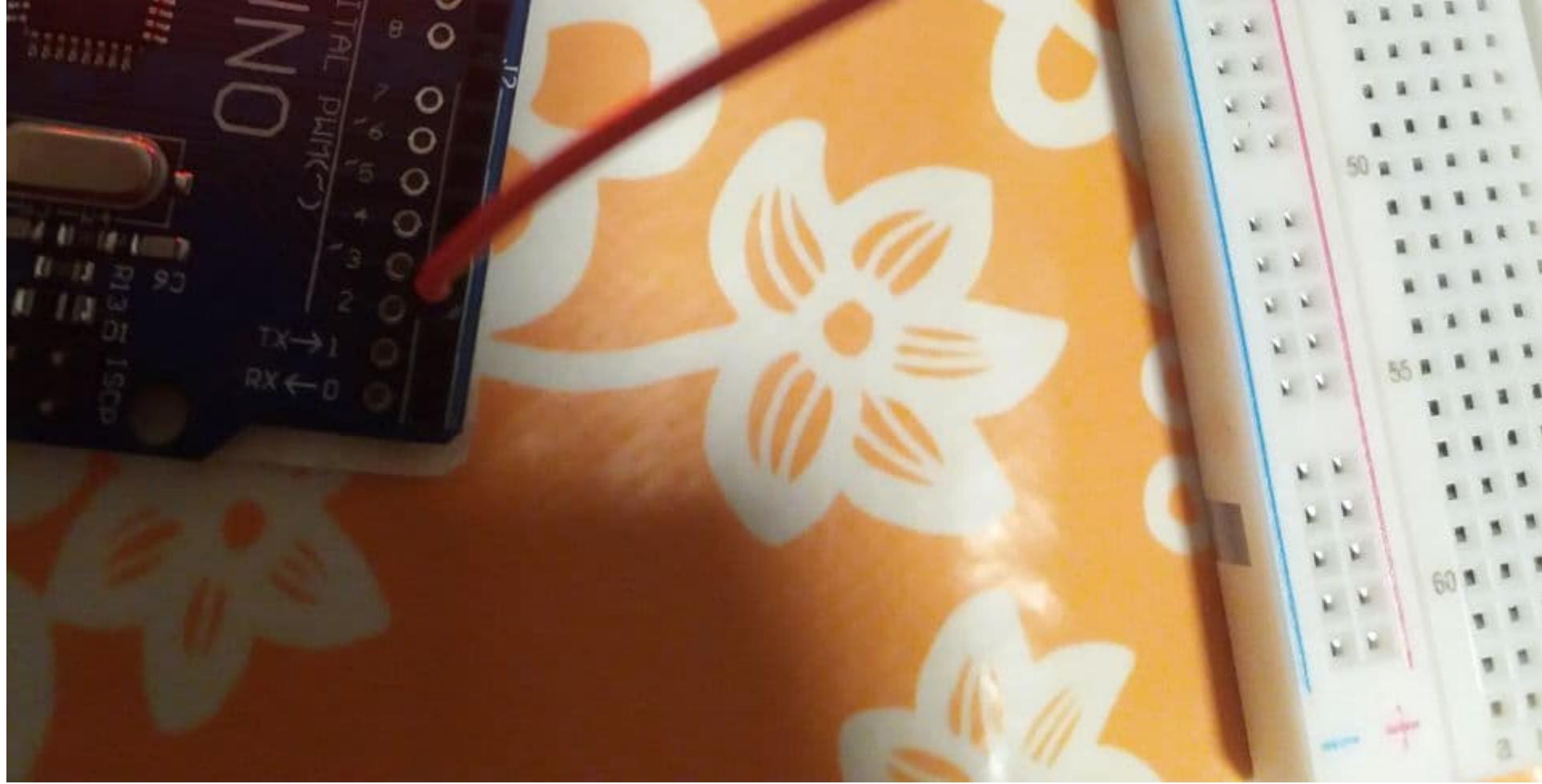
Out[20]:

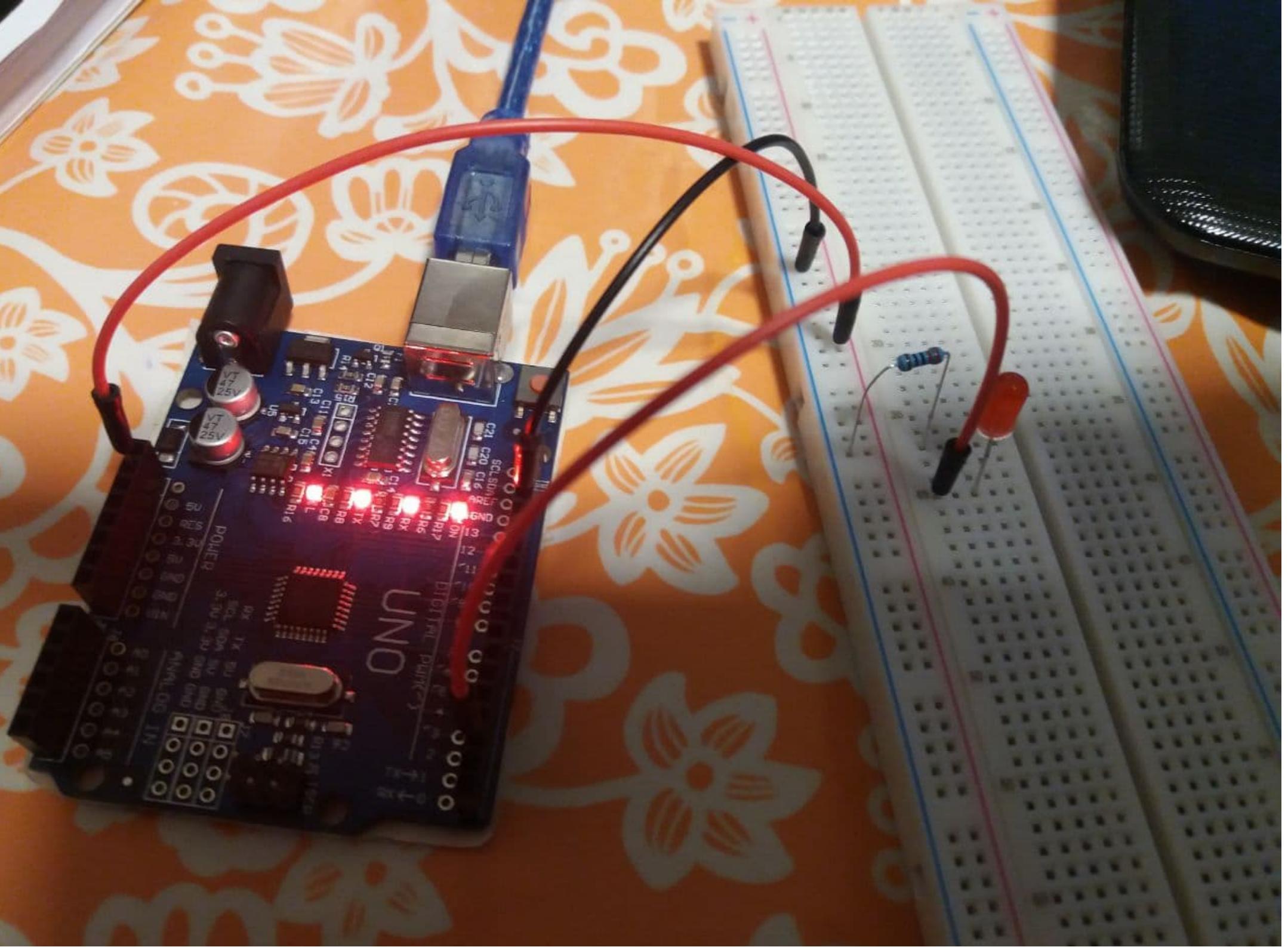
Led parpadeando para caso 3 con 1 seg de delay

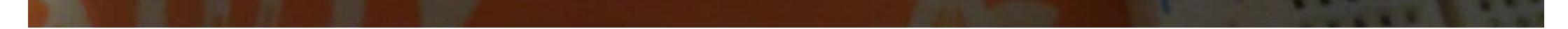


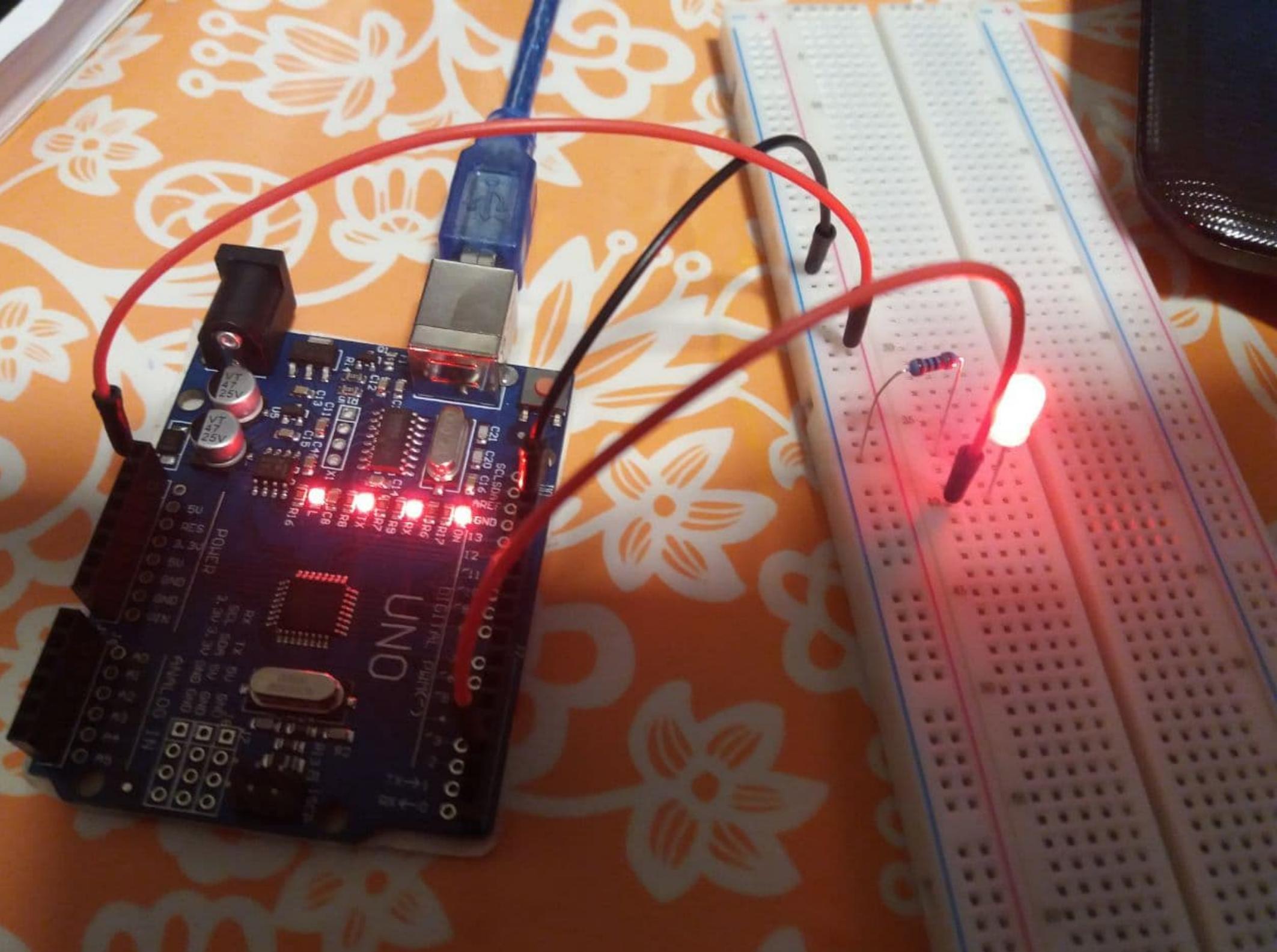
Algunas fotografías de las experiencias realizadas:



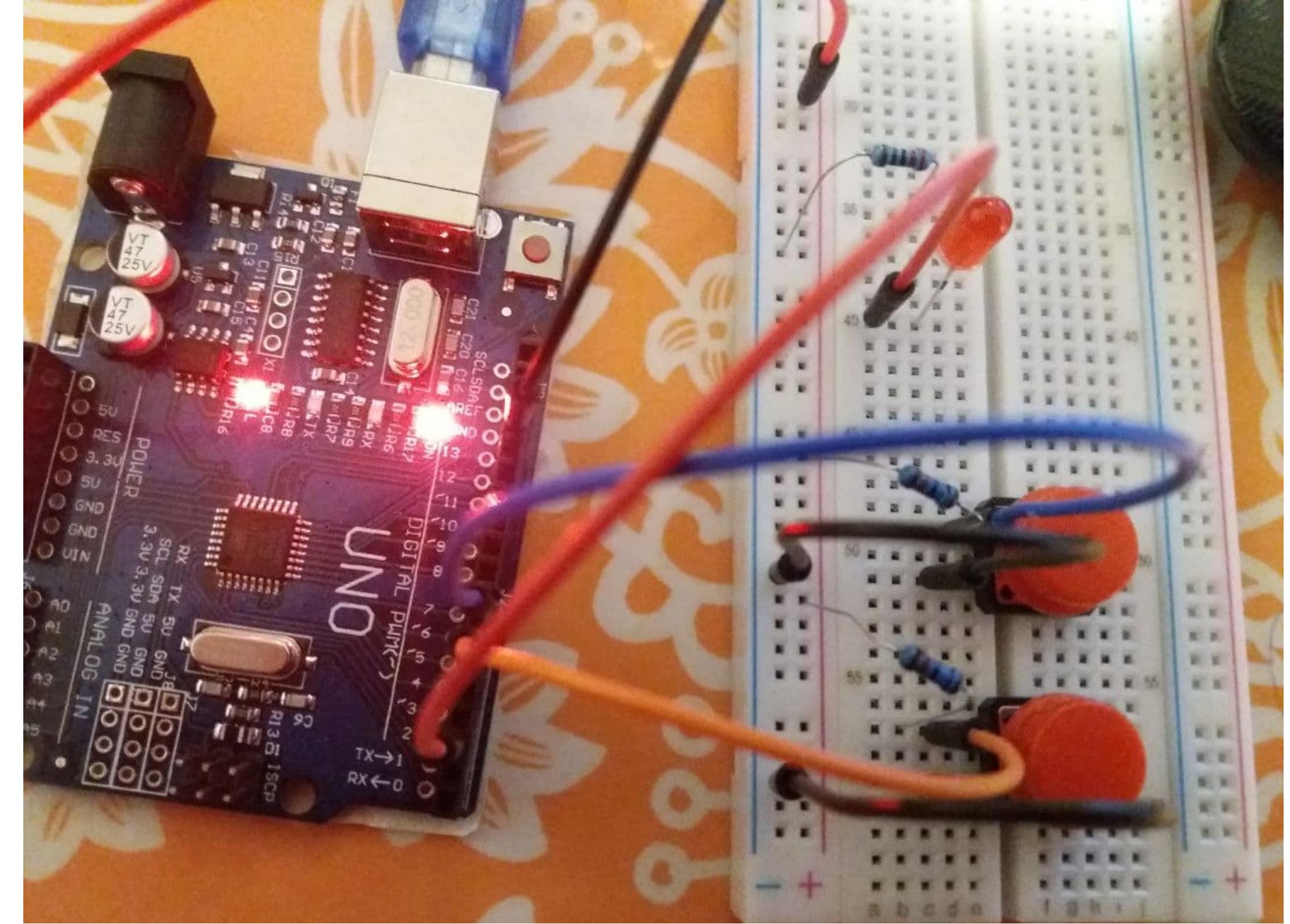




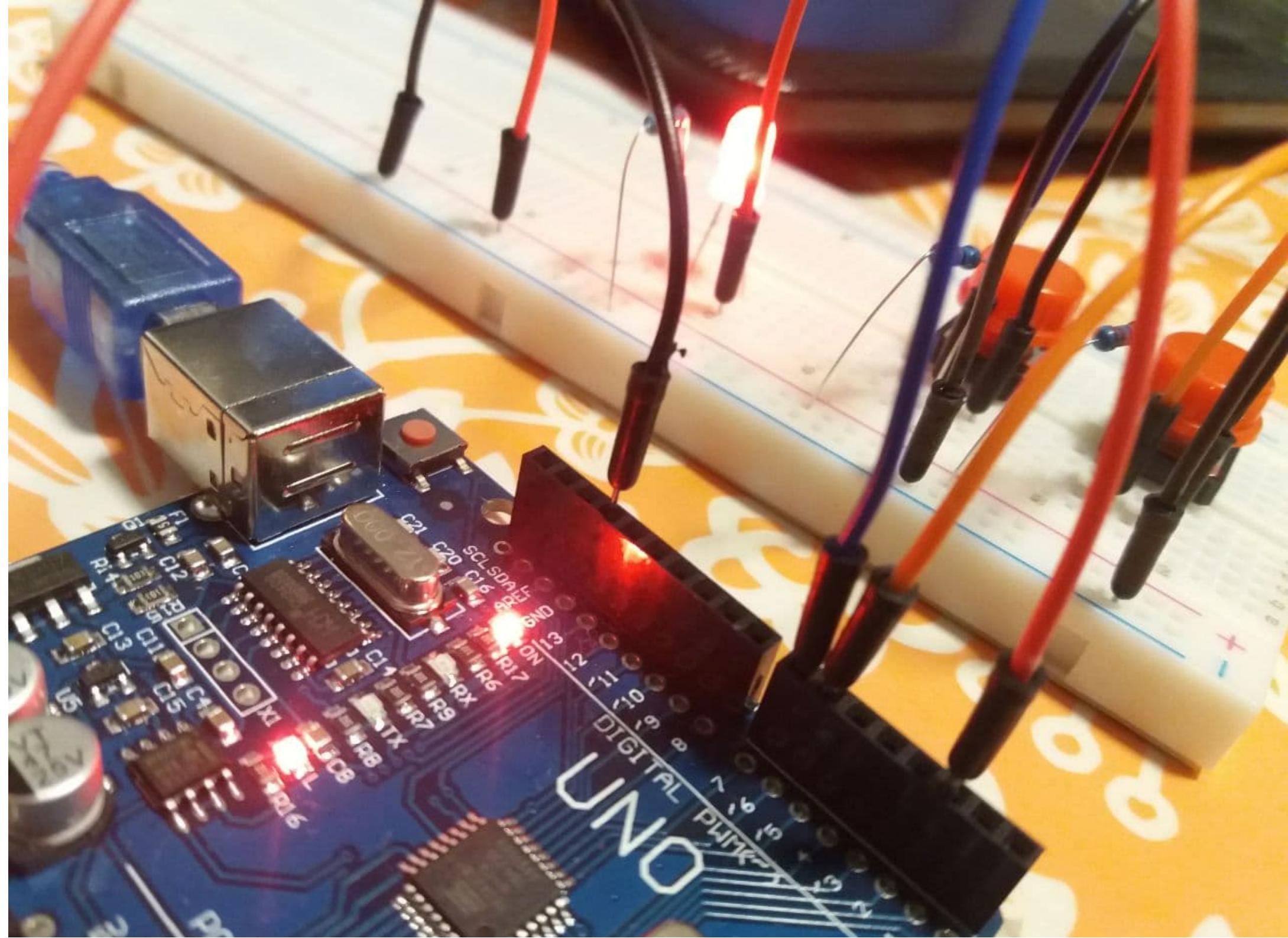




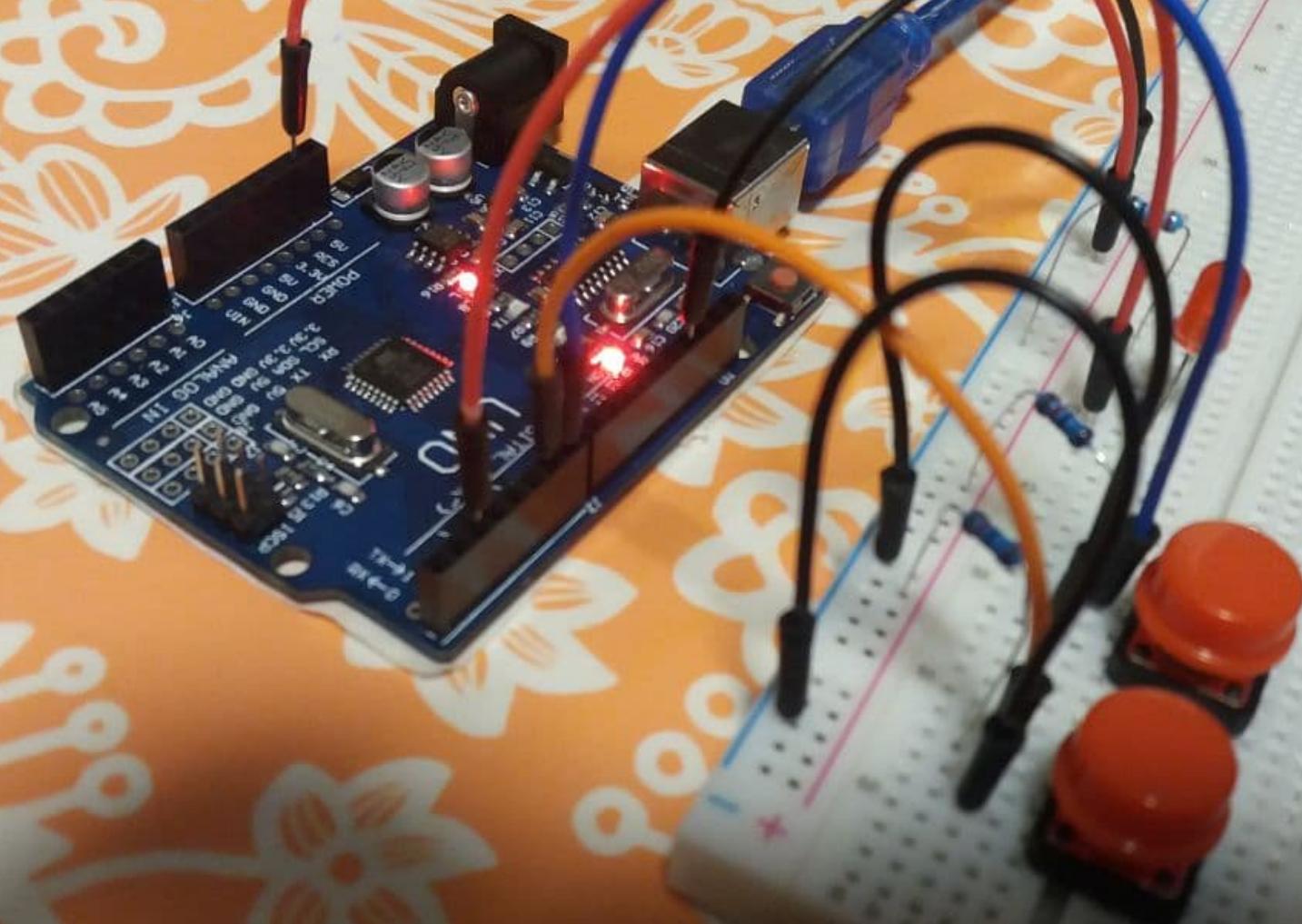


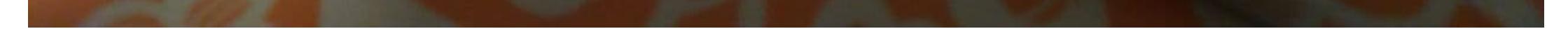


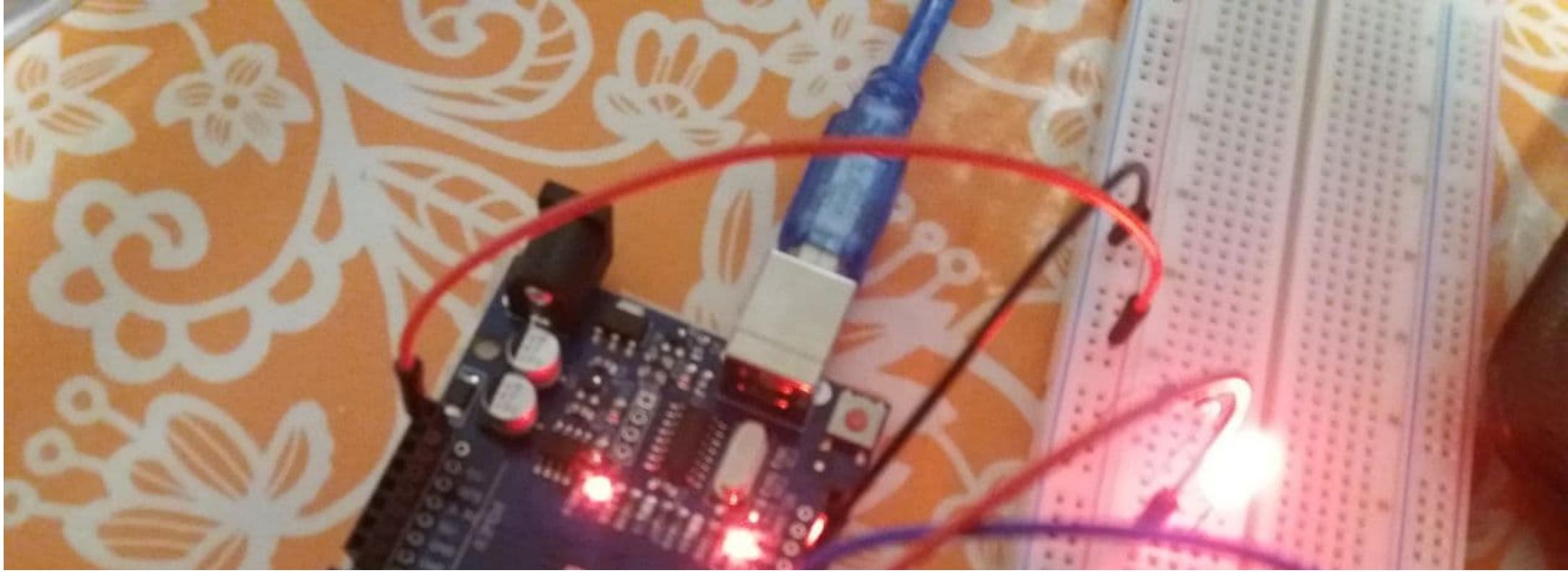












In []: