

# PROYECTO FINAL

## Electrónica Digital II

---

Facultad de Ingeniería y Ciencias Exactas y Naturales, Universidad Favaloro

CABA, Buenos Aires, Argentina

**Grupo 11.** Juana Kallis, Emma Fiorini, María Agustina Vidaurreta y Santiago Simaro

# Objetivos del proyecto

El sistema deberá permitir la carga de datos mediante un teclado matricial y transmitirlos a través de un módulo de radiofrecuencia RF 433MHz. Los datos también se visualizarán en un display de 7 segmentos con multiplexor.

# DISPLAY 7 SEGMENTOS

FUNCTION TABLE

INPUTS					OUTPUTS									
EL	BI	LT	D <sub>D</sub>	D <sub>C</sub>	D <sub>B</sub>	D <sub>A</sub>	O <sub>a</sub>	O <sub>b</sub>	O <sub>c</sub>	O <sub>d</sub>	O <sub>e</sub>	O <sub>f</sub>	O <sub>g</sub>	DISPLAY
X	X	L	X	X	X	X	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	blank
L	H	H	L	L	L	L	H	H	H	H	H	H	L	0
L	H	H	L	L	L	H	L	H	H	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	H	2
L	H	H	L	L	H	H	H	H	H	H	L	L	H	3
L	H	H	L	H	L	L	L	H	H	L	L	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	5
L	H	H	L	H	H	L	L	L	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	H	L	L	H	H	9
L	H	H	H	L	H	L	L	L	L	L	L	L	L	blank
L	H	H	H	L	H	H	L	L	L	L	L	L	L	blank
L	H	H	H	H	L	L	L	L	L	L	L	L	L	blank
L	H	H	H	H	H	L	L	L	L	L	L	L	L	blank
H	H	H	X	X	X	X		*					*	*

EL: Latch Enable

BI: Blanking Input.

LT: Lamp Test.

D<sub>D</sub>, D<sub>C</sub>, D<sub>B</sub>, D<sub>A</sub> : Binary Coded Decimal. Números representados en binario tomando las entradas como bits.

# CONTROL DE DISPLAYS CON HEF4511BP

**Table 2. Pin description**

<b>Symbol</b>	<b>Pin</b>	<b>Description</b>
$\overline{LT}$	3	lamp test input (active LOW)
$\overline{BL}$	4	ripple blanking input (active LOW)
LE	5	latch enable input (active HIGH)
D0 to D3	7, 1, 2, 6	address (data) input
$V_{SS}$	8	ground supply voltage
Qa to Qg	13, 12, 11, 10, 9, 15, 14	segment output
$V_{DD}$	16	supply voltage

# CÓDIGO

```

void decharabin(){//funcion que cambia de char a binario

    if(tecla == '0'){
        output_low(PIN_A6); // LE
        output_high(PIN_A7); // BI =
        output_high(PIN_A4); // LT =
        output_low(PIN_A3); // D
        output_low(PIN_A2); // C
        output_low(PIN_A1); // B
        output_low(PIN_A0); // A
    }
    if (tecla == '1'){
        output_low(PIN_A6); // LE
        output_high(PIN_A7); // BI
        output_high(PIN_A4); // LT
        output_low(PIN_A3); // D
        output_low(PIN_A2); // C
        output_low(PIN_A1); // B
        output_high(PIN_A0); // A
    }
    if (tecla == '2') {
        output_low(PIN_A6); // LE
        output_high(PIN_A7); // BI =
        output_high(PIN_A4); // LT =
        output_low(PIN_A3); // D
        output_low(PIN_A2); // C
        output_high(PIN_A1); // B
        output_low(PIN_A0); // A
    }
}

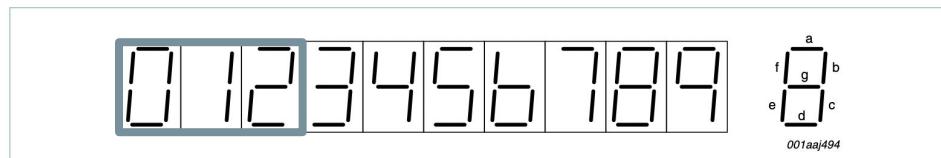
```

## 7. Functional description

Table 3. Function table<sup>[1]</sup>

Inputs										Outputs						Display
LE	BL	LT	D3	D2	D1	D0	Qa	Qb	Qc	Qd	Qe	Qf	Qg			
X	X	L	X	X	X	X	H	H	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	L	L	blank
L	H	H	L	L	L	L	H	H	H	H	H	H	H	H	L	0
L	H	H	L	L	L	H	L	H	H	L	L	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	H	L	H	2
L	H	H	L	L	H	H	H	H	H	H	H	L	L	H	H	3
L	H	H	L	H	L	L	L	H	H	L	L	H	H	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	H	H	5
L	H	H	L	H	H	L	L	H	H	H	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	L	L	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	H	L	L	H	H	H	H	9
L	H	H	H	L	H	X	L	L	L	L	L	L	L	L	L	blank
L	H	H	H	H	X	X	L	L	L	L	L	L	L	L	L	blank
H	H	H	X	X	X	X	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	

[1] H = HIGH voltage level; L = LOW voltage level; X = don't care; N.C. = no change.



# CÓDIGO

```

if (tecla == '3') {
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_low(PIN_A3); // D
    output_low(PIN_A2); // C
    output_high(PIN_A1); // B
    output_high(PIN_A0); // A
}
if (tecla == '4'){
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_low(PIN_A3); // D
    output_high(PIN_A2); // C
    output_low(PIN_A1); // B
    output_low(PIN_A0); // A
}

if (tecla == '5'){
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_low(PIN_A3); // D
    output_high(PIN_A2); // C
    output_low(PIN_A1); // B
    output_high(PIN_A0); // A
}

```

## 7. Functional description

Table 3. Function table<sup>[1]</sup>

Inputs										Outputs							Display
LE	BL	LT	D3	D2	D1	D0	Qa	Qb	Qc	Qd	Qe	Qf	Qg				
X	X	L	X	X	X	X	H	H	H	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	L	L	L	blank
L	H	H	L	L	L	L	H	H	H	H	H	H	H	H	H	L	0
L	H	H	L	L	L	H	L	H	H	H	L	L	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	H	L	H	H	2
L	H	H	L	L	H	H	H	H	H	H	L	L	H	L	H	H	3
L	H	H	L	H	L	L	L	H	H	H	L	L	H	H	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	L	H	H	5
L	H	H	L	H	H	L	L	L	H	H	H	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	H	L	L	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	H	H	L	L	H	L	H	H	9
L	H	H	H	H	X	X	L	L	L	L	L	L	L	L	L	L	blank
L	H	H	H	H	X	X	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	blank
H	H	H	X	X	X	X	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.

[1] H = HIGH voltage level; L = LOW voltage level; X = don't care; N.C. = no change.



# CÓDIGO

```

if (tecla == '7'){
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_low(PIN_A3); // D
    output_high(PIN_A2); // C
    output_high(PIN_A1); // B
    output_high(PIN_A0); // A
}

if (tecla == '8'){
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_high(PIN_A3); // D
    output_low(PIN_A2); // C
    output_low(PIN_A1); // B
    output_low(PIN_A0); // A
}

if (tecla == '9'){
    output_low(PIN_A6); // LE
    output_high(PIN_A7); // BI
    output_high(PIN_A4); // LT
    output_high(PIN_A3); // D
    output_low(PIN_A2); // C
    output_low(PIN_A1); // B
    output_high(PIN_A0); // A
}

```

## 7. Functional description

Table 3. Function table<sup>[1]</sup>

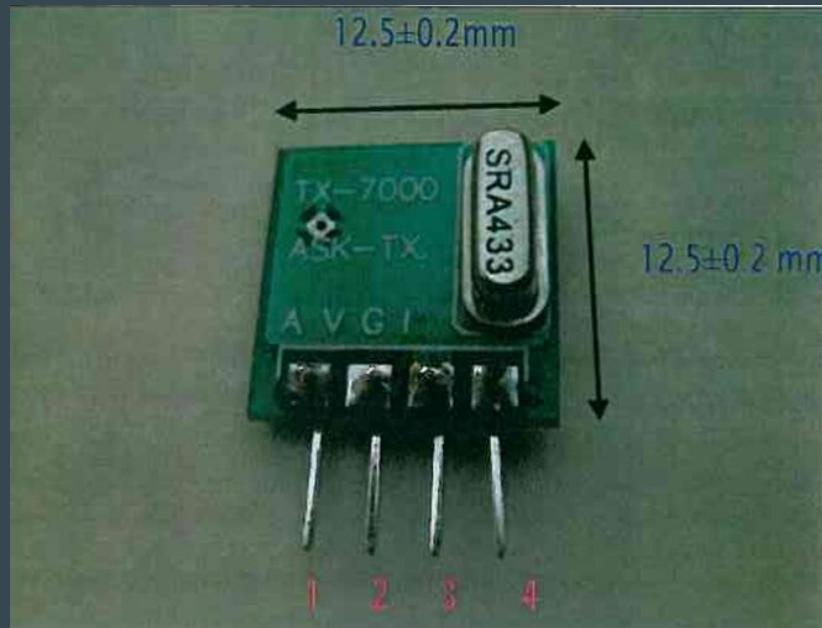
Inputs										Outputs							Display
LE	BL	LT	D3	D2	D1	D0	Qa	Qb	Qc	Qd	Qe	Qf	Qg				
X	X	L	X	X	X	X	H	H	H	H	H	H	H	H	H	H	8
X	L	H	X	X	X	X	L	L	L	L	L	L	L	L	L	L	blank
L	H	H	L	L	L	L	H	H	H	H	H	H	H	H	H	L	0
L	H	H	L	L	L	H	L	H	H	H	L	L	L	L	L	L	1
L	H	H	L	L	H	L	H	H	L	H	H	L	H	L	L	H	2
L	H	H	L	L	H	H	H	H	H	H	L	L	L	H	H	H	3
L	H	H	L	H	L	L	L	H	H	H	L	L	H	H	H	H	4
L	H	H	L	H	L	H	H	L	H	H	L	H	H	L	H	H	5
L	H	H	L	H	H	L	L	L	H	H	H	H	H	H	H	H	6
L	H	H	L	H	H	H	H	H	H	H	L	L	L	L	L	L	7
L	H	H	H	L	L	L	H	H	H	H	H	H	H	H	H	H	8
L	H	H	H	L	L	H	H	H	H	H	L	L	H	L	H	H	9
L	H	H	H	L	H	X	L	L	L	L	L	L	L	L	L	L	blank
L	H	H	H	H	X	X	L	L	L	L	L	L	L	L	L	L	blank
H	H	H	X	X	X	X	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.	N.C.

[1] H = HIGH voltage level; L = LOW voltage level; X = don't care; N.C. = no change.

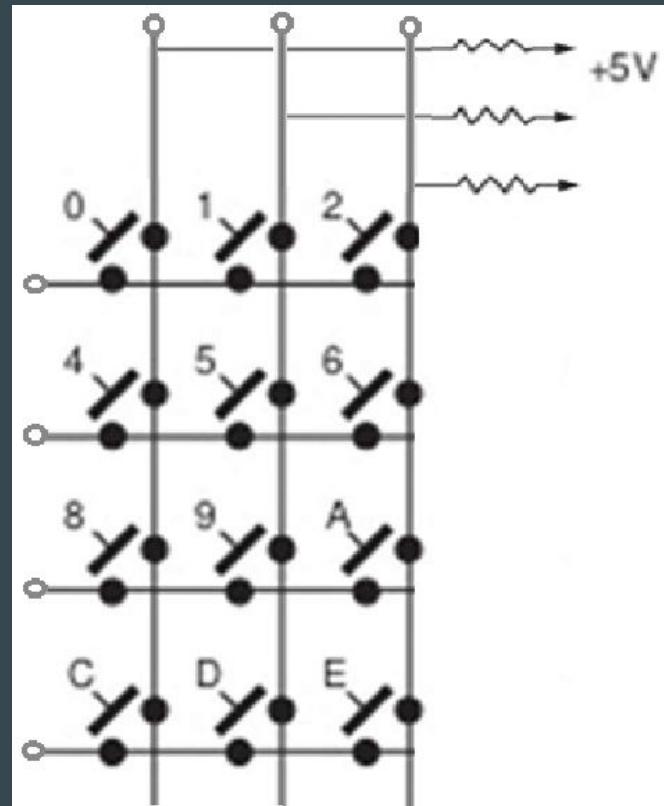


## RADIOFRECUENCIA (433MHz)

- PIN 1 : ANT ( antena)
- PIN 2; Vcc ( 3-12v)
- PIN 3 GND
- PIN 4 (DATA IN – PIN B5)



# Teclado Matricial 4x3



# Interrupciones utilizadas

#INT\_TIMER0

#INT\_IOC



Debounce del teclado matricial. Timer0 lleva la cuenta de 2 tiempos, 150 ms y 1s. En IOC una vez detectada la variación en el estado de los pines, esperamos hasta habilitar.

# Interrupción del Timer0

```
#INT_TIMER0 // ACA ESCRIBO QUÉ DEBO HACER EN CADA INTERRUPCIÓN
void Timer0_ISR() {

    set_timer0(61);

    if(contador_ms_1 >= 3) { //Pasaron 150ms
        contador_ms_1 = 0;
        flag_segundo = 1; // aviso que ya pasó el tiempo deseado
    }

    if(contador_ms_2 >= 20) { //Pasaron 1s
        contador_ms_2 = 0;
        flag_segundo2 = 1; // aviso que ya pasó el tiempo deseado
    }

    contador_ms_1++;
    contador_ms_2++;

}
```

# Interrupción IOC

```
#INT_IOC
void IOC_ISR(){
    if(flag_segundo == 1) { // Hubo variación de estado en los pines y pasó el tiempo de espera
        flag_debounce = 1;
        flag_segundo = 0;
    }
}
```

## PROTOCOLO DE COMUNICACIÓN RS232

- Velocidad de 9600 BAUD
- Número de bits de la comunicación 8
- Paridad: N
- Bit de stop = 1
- Pin de transmisión Pin B5
- Protocolo acordado ‘<’ número ‘>’ de codificación

# Código

```
void main()
{
    Init_Keypad();
    InitTimer0();
    Init_GPIO();

    while(TRUE)
    {

        tecla = getKey();

        if(tecla != 'f') {

            if(flag_segundo2 == 1) {
                printf("<%c>", tecla);
                flag_segundo2 = 0;
            }
            decharabin();
        }
    }
}
```

```
void Init_GPIO(void){

    set_tris_a(0b00000000);

    // RB0, RB1, RB2 y RB3 filas del teclado, entrada
    // RB4, RB6 RB7 columnas, salidas
    set_tris_b(0b11010000);

    enable_interrupts(INT_IOC_B4);
    enable_interrupts(INT_IOC_B6);
    enable_interrupts(INT_IOC_B7);

    enable_interrupts(GLOBAL);
}
```

# Problemáticas

Problemática 1

- Multiplexor (Números 5 y 3)

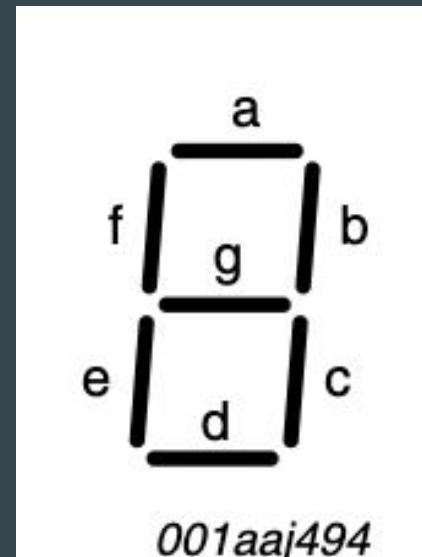
Problemática 2

- Ruido del teclado

Problemática 3

- Radio Frecuencia

# Número 5 y 3



# Número 5 y 3

Inputs							Outputs							Display
LE	BL	LT	D3	D2	D1	D0	Qa	Qb	Qc	Qd	Qe	Qf	Qg	
L	H	H	L	L	H	H	H	H	H	H	L	L	H	3
L	H	H	L	H	L	H	H	L	H	H	L	H	H	5

```
if (tecla == '3') {  
    output_low(PIN_A6); // LE  
    output_high(PIN_A7); // BI  
    output_high(PIN_A4); // LT  
    output_low(PIN_A3); // D  
    output_low(PIN_A2); // C  
    output_high(PIN_A1); // B  
    output_high(PIN_A0); // A  
}
```

```
if (tecla == '5'){  
    output_low(PIN_A6); // LE  
    output_high(PIN_A7); // BI  
    output_high(PIN_A4); // LT  
    output_low(PIN_A3); // D  
    output_high(PIN_A2); // C  
    output_low(PIN_A1); // B  
    output_high(PIN_A0); // A  
}
```

# Ruido del teclado

Resistencias de Pull- Up

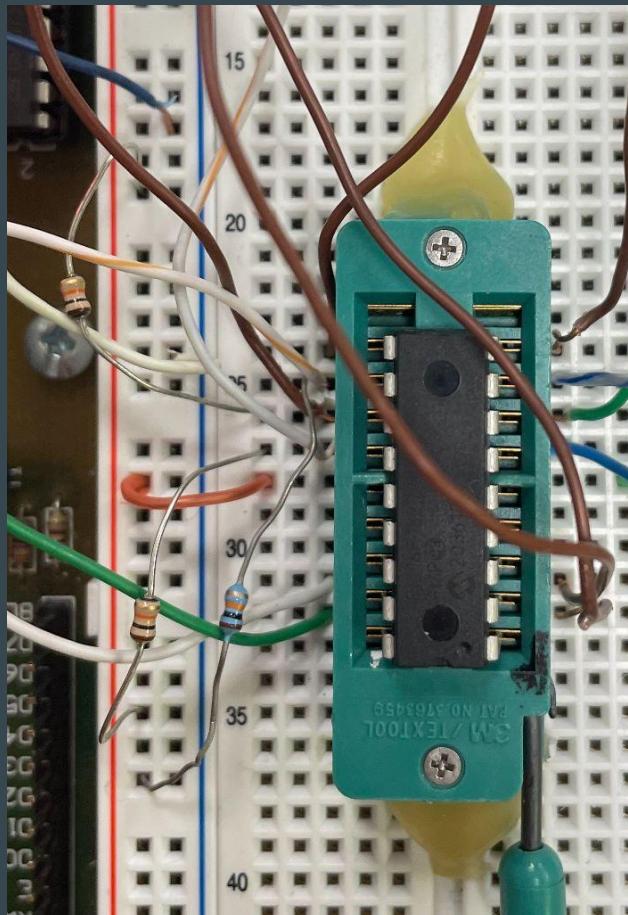
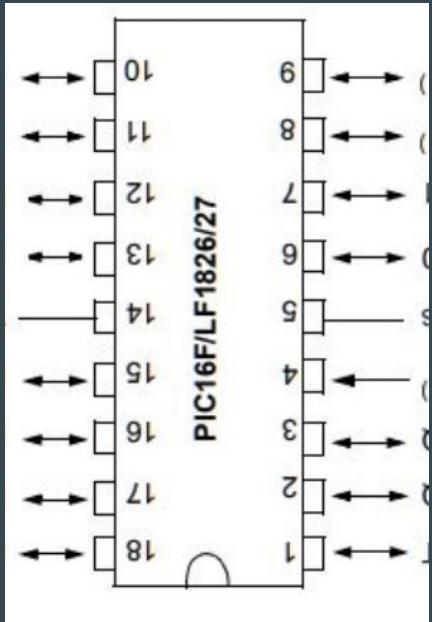
$$R = 10k\Omega$$

Columnas

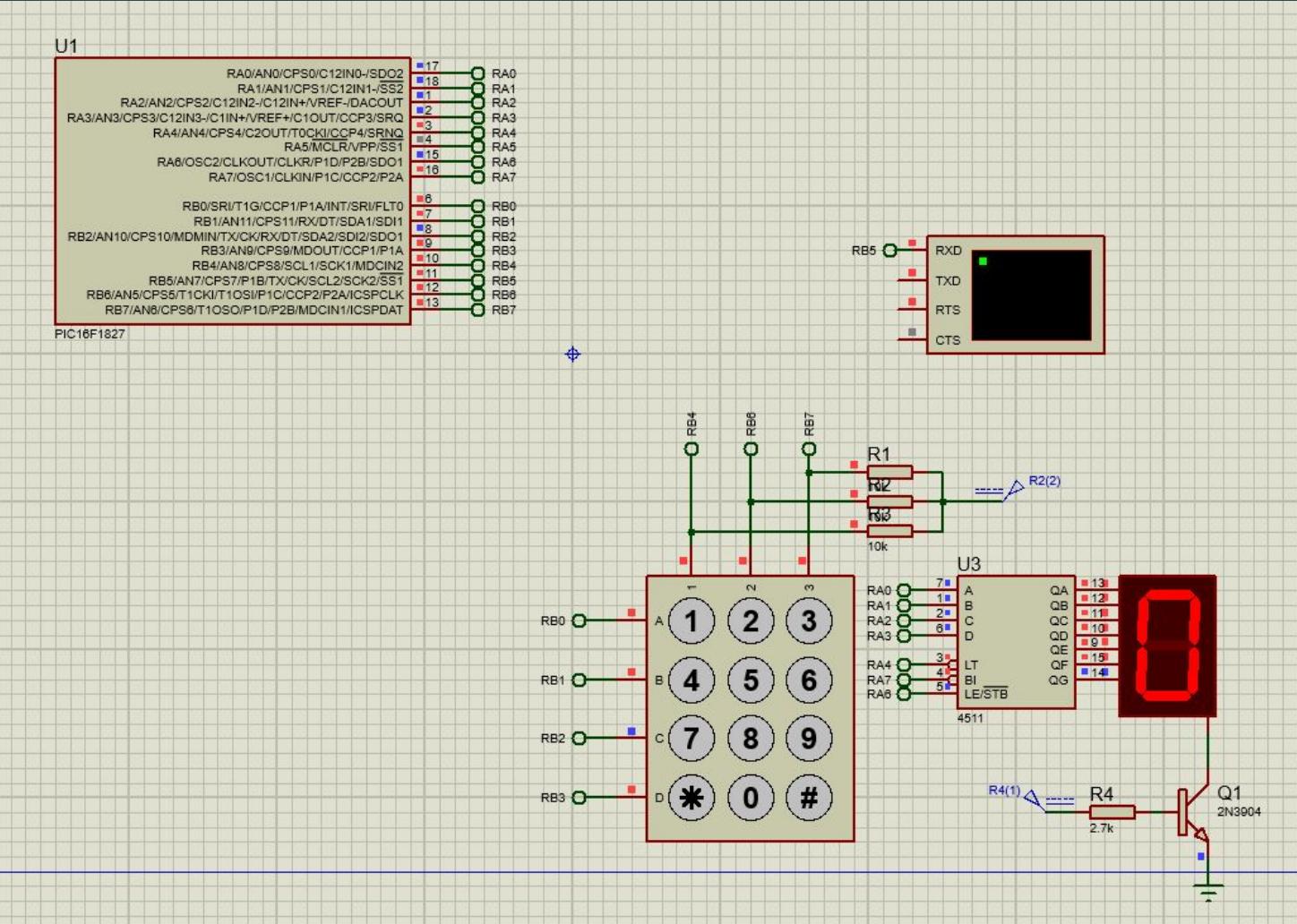
**R B4**

**R B6**

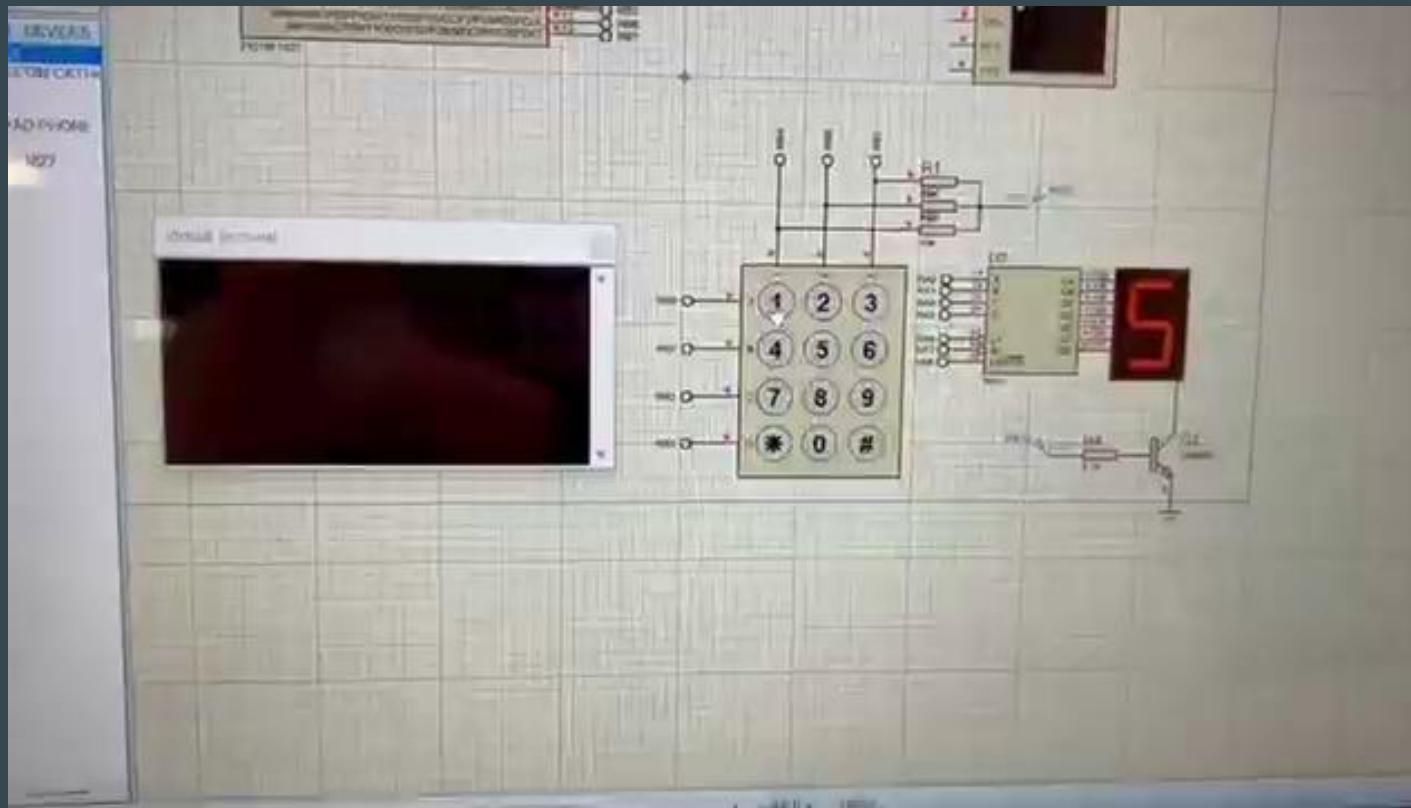
**R B7**



# Proteus



# Video



# Video



# Gracias!

**Grupo 11.** Juana Kallis, Emma Fiorini, Maria Agustina Vidaurreta y Santiago Simaro