

# Segundo examen práctico

**Alumno:** Azpeitia Hernández Vladimir

**Grupo:** 3CM17

**Fecha:** 6 de octubre de 2021

## Planteamiento del problema

Desarrollar un programa en ensamblador que funcione como un contador, haciendo uso de interrupciones y un botón para contar el número de pulsaciones que el usuario ha realizado. El número se deberá mostrar en una matriz de leds de 8x8. El contador solo llegará hasta el número 9, una vez llegué el 9 el contador se reiniciará en 0.

## Código de la solución

---

```
1 .include "m8535def.inc"
2     .def adl = r17
3     .def ctados = r23
4     .def col = r18
5     .def aux = r19
6     .def cta = r20
7     .def col2 = r21
8     .def mcount = r22
9 ;-----
10 .macro num
11     push zh
12     push zl
13     ldi ZH, high(@0<<1)           ; Initialize Z pointer
14     ldi ZL, low(@0<<1)
15     lpm r0, Z+
16     lpm r1, Z+
17     lpm r2, Z+
18     lpm r3, Z+
19     lpm r4, Z+
20     lpm r5, Z+
21     lpm r6, Z+
22     lpm r7, Z
23     pop zl
24     pop zh
```

```

25 .endm
26 ;-----
27         rjmp  Start
28         rjmp  cuenta1
29         rjmp  cuenta2
30 .org $008
31     nop
32     rjmp  muestra
33
34 ;-----
35 Start:
36         LDI  R16,LOW(RAMEND)
37     OUT  SPL,R16
38     LDI  R16,HIGH(RAMEND)
39     OUT  SPH,R16
40     ldi  aux, 0b11000000
41     out  gicr,aux
42     ldi  aux,15
43     out  mcucr,aux
44
45     SER  R16
46     OUT  portd,R16
47     OUT  DDRB,R16
48     OUT  DDRC,R16
49     OUT  DDRA,R16
50     LDI  R16,$ED ; 1110 1101
51     ldi  aux,2
52     out  tccr0,aux
53     ldi  aux,1
54     out  tmsk,aux
55     SEI
56         ldi  cta,-1
57         ldi  ctados,-1
58         ldi  mcount,0
59     ldi  col,1
60     ldi  col2,1
61     clr  zh
62     ldi  z1,0
63 ;-----
64 Loop:
65     nop
66     rjmp  Loop
67
68 ;-----
69 muestra:
70     cpi  mcount,0
71     breq  barre
72     rjmp  barre2

```

```

73 barre:
74     mov aux,cta
75     rcall deco
76     ldi aux,0
77     out portb,aux
78         out porta,zh
79         ld aux,z+
80         lsl col
81
82         brcs nbarre
83
84 sss:
85         com col
86         out portc,col
87         com col
88         out porta,aux
89         reti
90 nbarre:
91         ldi col,1
92         ldi z1,0
93         ld aux,z+
94
95         ldi mcount,1
96         rjmp sss
97     ; reti
98
99 barre2:
100    mov aux,ctados
101    ;ldi aux,3
102    rcall deco
103    ldi aux,0
104    out porta,aux
105        out portb,zh
106        ld aux,z+
107        lsl col2
108
109        brcs nbarre2
110
111 sss2:
112        com col2
113        out portc,col2
114        com col2
115        out portb,aux
116        reti
117 nbarre2:
118        ldi col2,1
119        ldi z1,0
120        ld aux,z+

```

```

121
122     ldi mcount,0
123     rjmp sss2
124     ;reti
125
126
127 ;-----
128 cuenta1:
129     inc cta
130     cpi cta,10
131     brne rcta1
132     ldi cta,-1
133 rcta1:
134     reti
135 cuenta2:
136     inc ctados
137     cpi ctados,10
138     brne rcta2
139     ldi ctados,-1
140 rcta2:
141     reti
142
143 deco:
144     cpi aux,0
145     breq cta0
146     cpi aux,1
147     breq cta1
148     cpi aux,2
149     breq cta2
150     cpi aux,3
151     breq cta3
152     cpi aux,4
153     breq cta4
154     cpi aux,5
155     breq cta5
156     cpi aux,6
157     breq cta6
158     cpi aux,7
159     breq cta7
160     cpi aux,8
161     breq cta8
162     cpi aux,9
163     breq cta9
164 ncta:
165     ret
166 cta3:
167     rjmp cta31
168 cta4:

```

```

169      rjmp cta41
170 cta5:
171      rjmp cta51
172 cta6:
173      rjmp cta61
174 cta7:
175      rjmp cta71
176 cta8:
177      rjmp cta81
178 cta9:
179      rjmp cta91
180
181 cta0:
182      num cero
183      rjmp ncta
184 cta1:
185      num uno
186      rjmp ncta
187 cta2:
188      num dos
189      rjmp ncta
190 cta31:
191      num tres
192      rjmp ncta
193 cta41:
194      num cuatro
195      rjmp ncta
196 cta51:
197      num cinco
198      rjmp ncta
199 cta61:
200      num seis
201      rjmp ncta
202 cta71:
203      num siete
204      rjmp ncta
205 cta81:
206      num ocho
207      rjmp ncta
208 cta91:
209      num nueve
210      rjmp ncta
211 ; -----
212 cero:
213      .db $00,$7C,$82,$82,$82,$7C,$00,$02
214 uno:
215      .db $00,$22,$42,$FE,$02,$02,$00,$04
216 dos:

```

```

217      .db $00,$42,$86,$8A,$92,$62,$00,$08
218 tres:
219      .db $00,$44,$82,$92,$92,$6C,$00,$10
220 cuatro:
221      .db $00,$08,$18,$28,$48,$FE,$00,$20
222 cinco:
223      .db $00,$F4,$92,$92,$92,$8C,$00,$40
224 seis:
225      .db $00,$7C,$92,$92,$92,$4C,$00,$40
226 siete:
227      .db $00,$80,$80,$80,$80,$FE,$00,$80
228 ocho:
229      .db $00,$6C,$92,$92,$92,$6C,$00,$08
230 nueve:
231      .db $00,$64,$92,$92,$92,$7C,$00,$08

```

---

## Simulación del circuito en Proteus 8 Professional

A continuación se muestran capturas de pantalla de la simulación de nuestro circuito en Proteus 8 Professional, en este caso utilizamos una frecuencia de 1Mhz para el reloj del microcontrolador ATmega 8535.

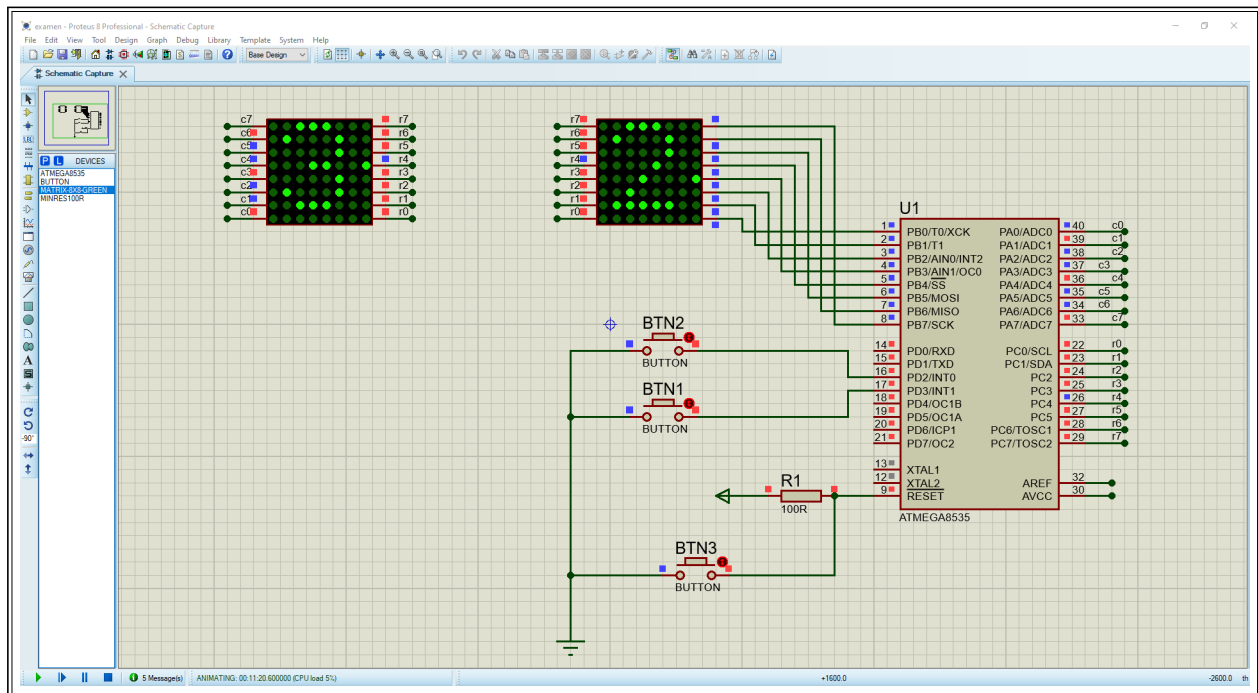


Figura 1: Simulación en Proteus 8 Professional Captura 1

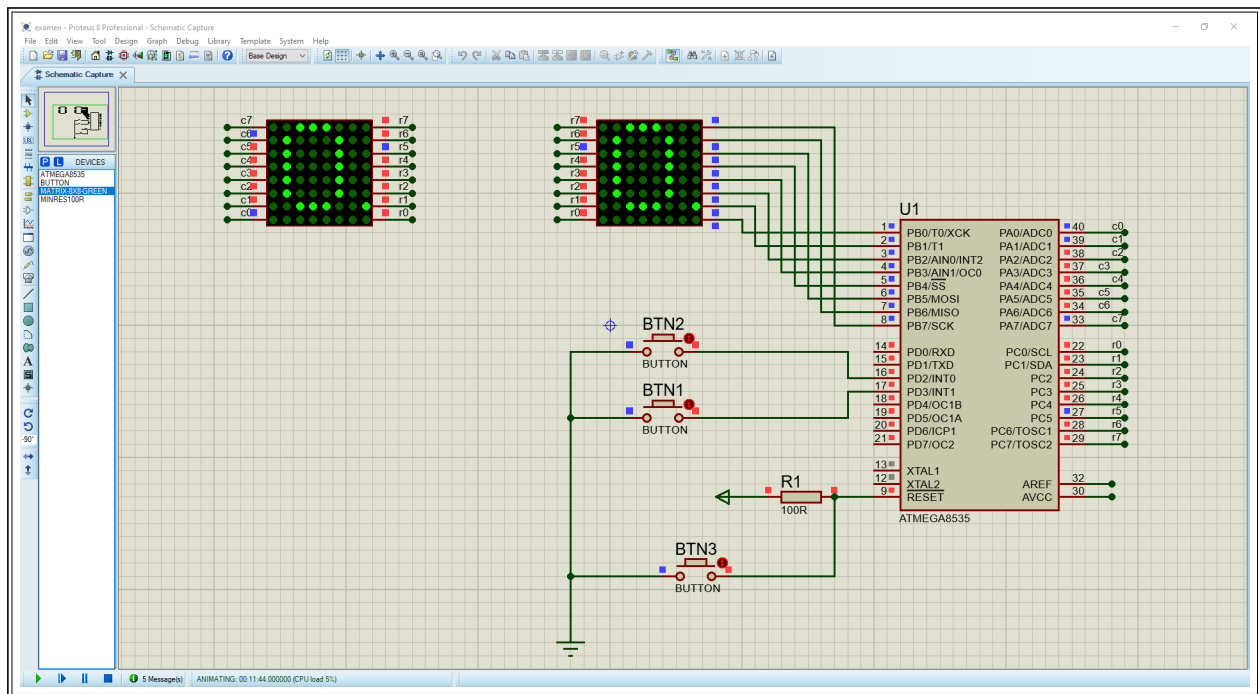


Figura 2: Simulación en Proteus 8 Professional Captura 2

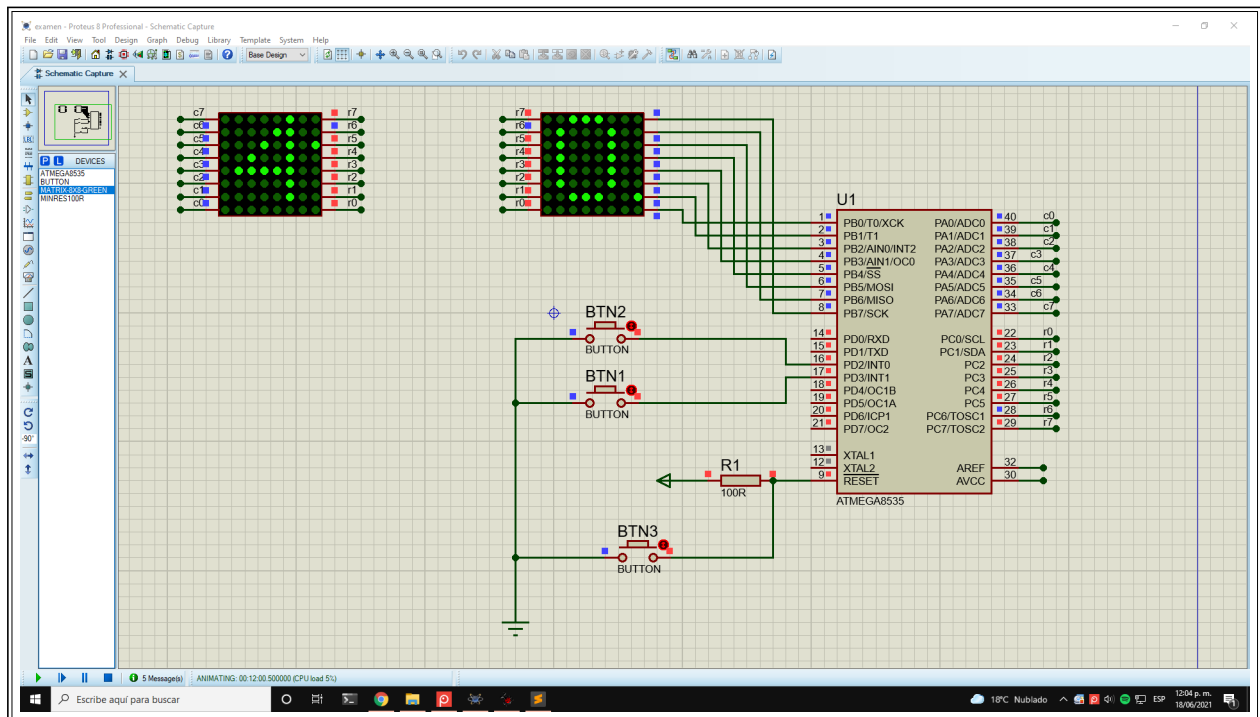


Figura 3: Simulación en Proteus 8 Professional Captura 3

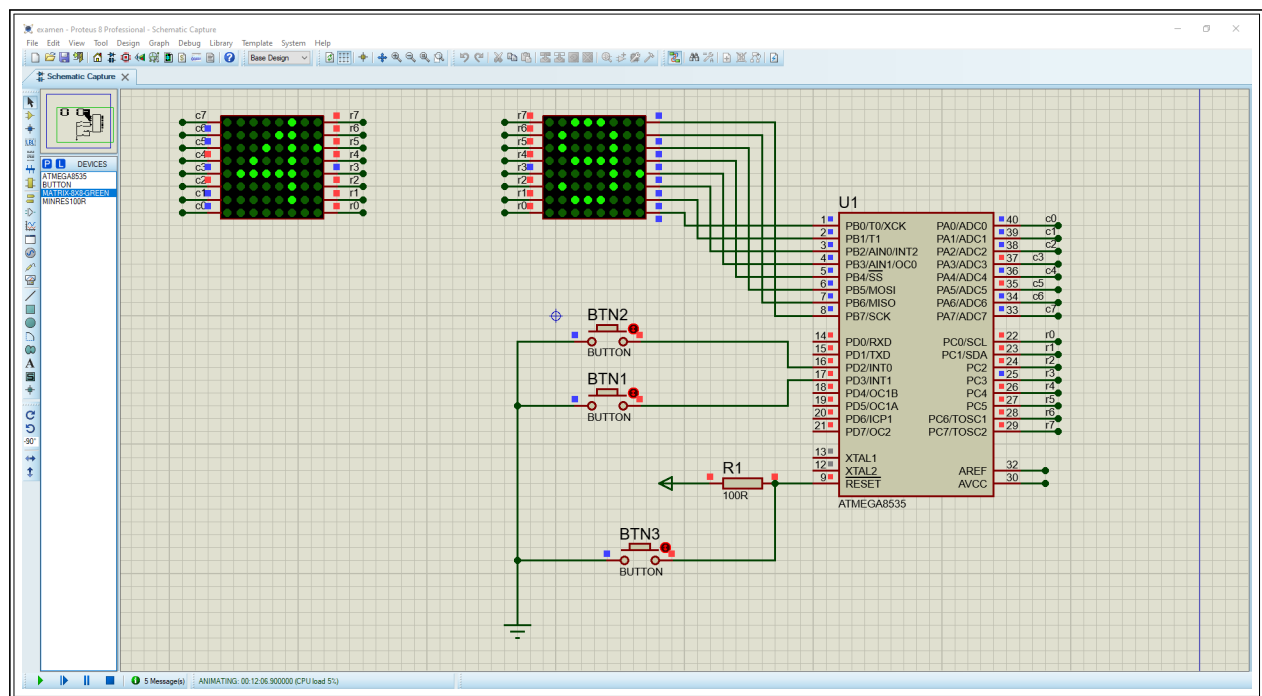


Figura 4: Simulación en Proteus 8 Professional Captura 4