



Instituto Politécnico Nacional



Escuela Superior de Computo

Materia:

Introducción a los microcontroladores.

Profesor:

Sanchez Aguilar Fernando

Alumnos:

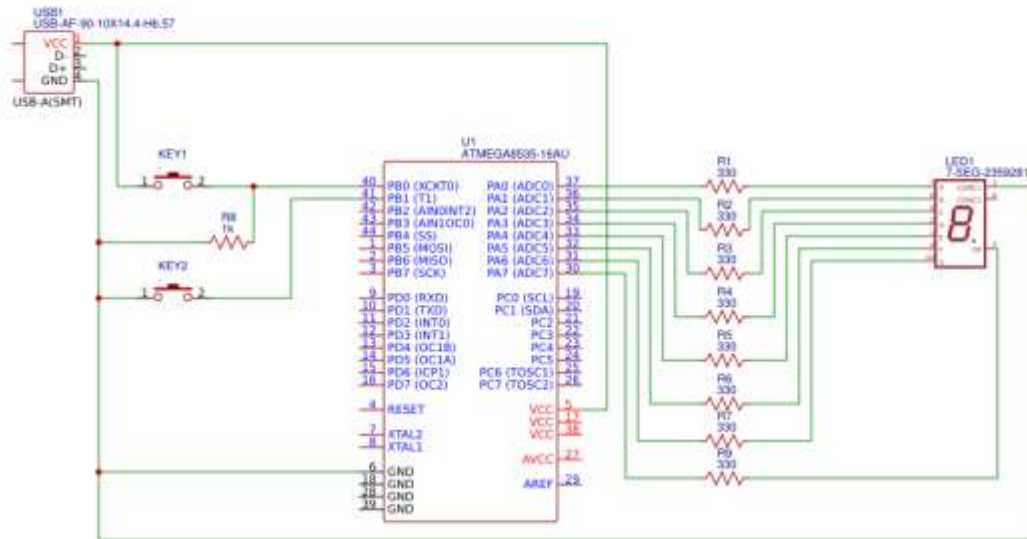
Aldavera Gallaga Iván

Lara Soto Rubén Jair

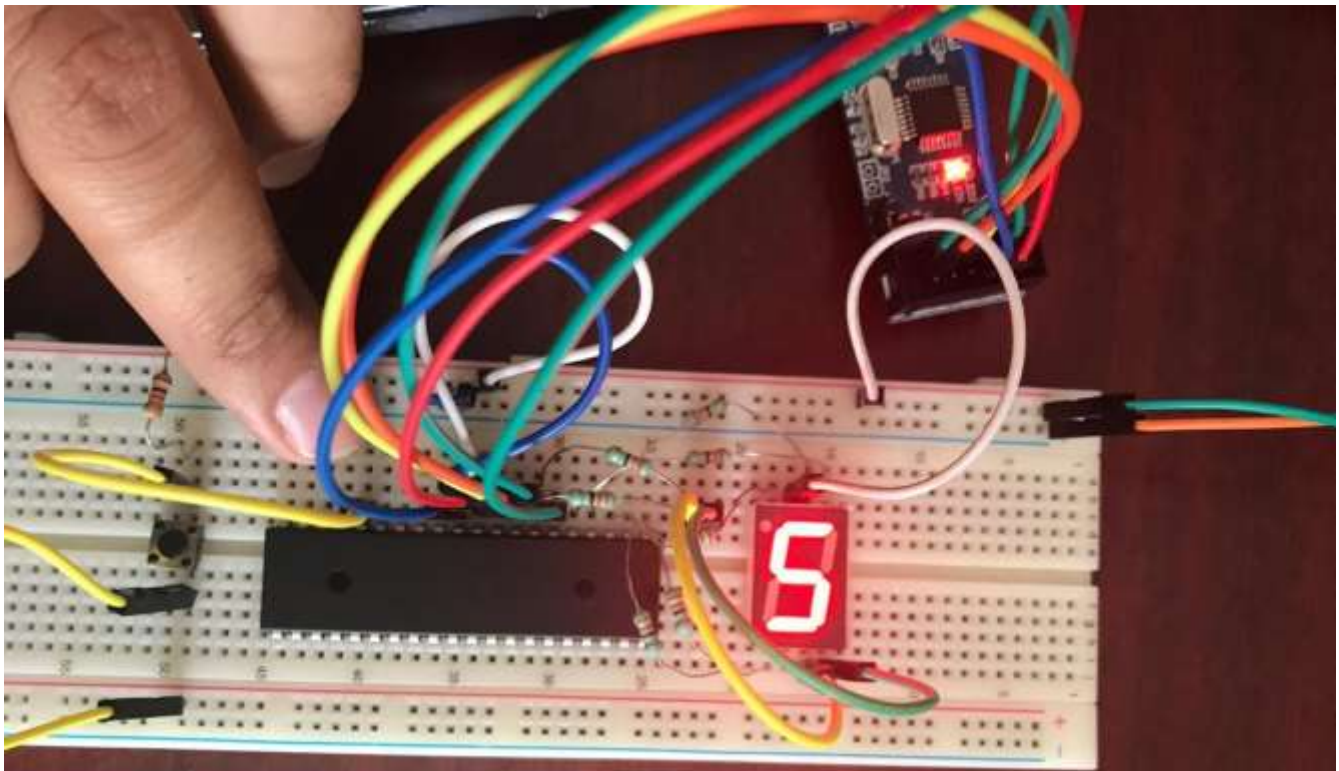
Morales Castellanos Adolfo Erik

Practica N°19

Interrupciones



TITLE: Practica Interrupciones		REV: 1.0
Company: ESCOM		Sheet: 1/1
Date: 2019-05-13		Drawn By: Equipo 4



```

1.  /*****
2.  This program was created by the CodeWizardAVR V3.35
3.  Automatic Program Generator
4.  © Copyright 1998-2019 Pavel Haiduc, HP InfoTech s.r.l.
5.  http://www.hpinfotech.com
6.
7.  Project :
8.  Version :
9.  Date    : 4/4/2019
10. Author : Equipo 4
11. Company :
12. Comments:
13.
14.
15. Chip type           : ATmega8535L
16. Program type        : Application
17. AVR Core Clock frequency: 1.000000 MHz
18. Memory model        : Small
19. External RAM size    : 0
20. Data Stack size     : 128
21. *****/
22.
23. #include <mega8535.h>
24. #include <delay.h>
25. char var;
26. const char tabla7segmentos[10]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7c,0x07,0x7f,0x6f}
    ;
27. // Declare your global variables here
28.
29. // External Interrupt 0 service routine
30. interrupt [EXT_INT0] void ext_int0_isr(void)
31. {
32. // Place your code here
33. if(var<9)
34. var++;
35. delay_ms(50);
36.
37. }
38.
39. // External Interrupt 1 service routine
40. interrupt [EXT_INT1] void ext_int1_isr(void)
41. {
42. // Place your code here
43. if(var>0)
44. var--;
45. delay_ms(200);
46.
47. }
48.
49. void main(void)
50. {
51. // Declare your local variables here
52.
53. // Input/Output Ports initialization
54. // Port A initialization
55. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
56. DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) | (0<<DDA0);
57. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
58. PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) | (0<<PORTA1) | (0<<PORTA0);

```

```

59.
60. // Port B initialization
61. // Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=Out
62. DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0);
63. // State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
64. PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<PORTB0);
65.
66. // Port C initialization
67. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
68. DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) | (0<<DDC0);
69. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
70. PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) | (0<<PORTC0);
71.
72. // Port D initialization
73. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
74. DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) | (0<<DDD0);
75. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=P Bit2=T Bit1=T Bit0=T
76. PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (1<<PORTD3) | (0<<PORTD2) | (0<<PORTD1) | (0<<PORTD0);
77.
78. // Timer/Counter 0 initialization
79. // Clock source: System Clock
80. // Clock value: Timer 0 Stopped
81. // Mode: Normal top=0xFF
82. // OC0 output: Disconnected
83. TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (0<<CS01) | (0<<CS00);
84. TCNT0=0x00;
85. OCR0=0x00;
86.
87. // Timer/Counter 1 initialization
88. // Clock source: System Clock
89. // Clock value: Timer1 Stopped
90. // Mode: Normal top=0xFFFF
91. // OC1A output: Disconnected
92. // OC1B output: Disconnected
93. // Noise Canceler: Off
94. // Input Capture on Falling Edge
95. // Timer1 Overflow Interrupt: Off
96. // Input Capture Interrupt: Off
97. // Compare A Match Interrupt: Off
98. // Compare B Match Interrupt: Off
99. TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WGM10);
100. TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) | (0<<CS10);
101. TCNT1H=0x00;
102. TCNT1L=0x00;
103. ICR1H=0x00;
104. ICR1L=0x00;
105. OCR1AH=0x00;
106. OCR1AL=0x00;
107. OCR1BH=0x00;
108. OCR1BL=0x00;
109.

```

```

110.    // Timer/Counter 2 initialization
111.    // Clock source: System Clock
112.    // Clock value: Timer2 Stopped
113.    // Mode: Normal top=0xFF
114.    // OC2 output: Disconnected
115.    ASSR=0<<AS2;
116.    TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<
CS21) | (0<<CS20);
117.    TCNT2=0x00;
118.    OCR2=0x00;
119.
120.    // Timer(s)/Counter(s) Interrupt(s) initialization
121.    TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) |
(0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);
122.
123.    // External Interrupt(s) initialization
124.    // INT0: On
125.    // INT0 Mode: Rising Edge
126.    // INT1: On
127.    // INT1 Mode: Low level
128.    // INT2: Off
129.    GICR|=(1<<INT1) | (1<<INT0) | (0<<INT2);
130.    MCUCR=(0<<ISC11) | (0<<ISC10) | (1<<ISC01) | (1<<ISC00);
131.    MCUCSR=(0<<ISC2);
132.    GIFR=(1<<INTF1) | (1<<INTF0) | (0<<INTF2);
133.
134.    // USART initialization
135.    // USART disabled
136.    UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) | (0<<U
CS22) | (0<<RXB8) | (0<<TXB8);
137.
138.    // Analog Comparator initialization
139.    // Analog Comparator: Off
140.    // The Analog Comparator's positive input is
141.    // connected to the AIN0 pin
142.    // The Analog Comparator's negative input is
143.    // connected to the AIN1 pin
144.    ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) |
(0<<ACIS1) | (0<<ACIS0);
145.    SFIOR=(0<<ACME);
146.
147.    // ADC initialization
148.    // ADC disabled
149.    ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<AD
PS2) | (0<<ADPS1) | (0<<ADPS0);
150.
151.    // SPI initialization
152.    // SPI disabled
153.    SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA)
| (0<<SPR1) | (0<<SPR0);
154.
155.    // TWI initialization
156.    // TWI disabled
157.    TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);
158.
159.    // Globally enable interrupts
160.    #asm("sei")
161.    var=2;
162.    while (1)
163.    {
164.        // Place your code here

```

```
165.          PORTB=~tabla7segmentos[var];  
166.      }  
167. }
```