Práctica #2 Contador
Introducción a los Microcontroladores
Fernando Aguilar Sánchez
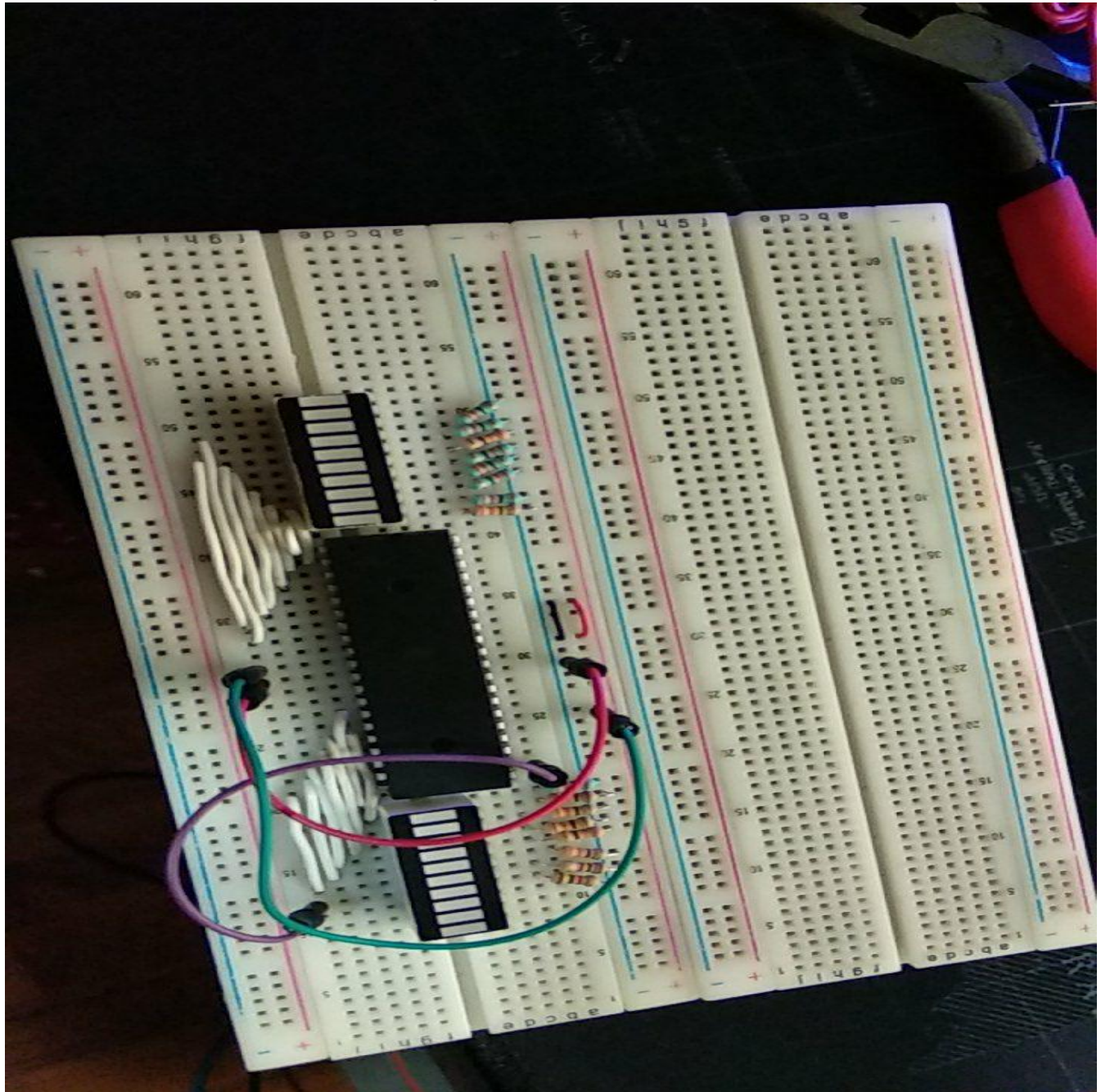
Jorge Gómez Reus

Índice

1. Protoboard

Figura 1: ProtoBoard

2. Código

```
1  #include <mega8535.h>
2  #include <delay.h>
3
4  void main(void)
5  {
6  // Input/Output Ports initialization
7  // Port A initialization
8  // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
9  DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) | (0<<DDA0);
10 // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
11 PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) | (0<<PORTA1) | (0<<
       PORTA0);
12
13 // Port B initialization
14 // Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=Out
15 DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0);
16 // State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
17 PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<
       PORTB0);
18
19 // Port C initialization
20 // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
21 DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) | (0<<DDC0);
22 // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
23 PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) | (0<<
       PORTC0);
24
25 // Port D initialization
26 // Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=Out
27 DDRD=(1<<DDD7) | (1<<DDD6) | (1<<DDD5) | (1<<DDD4) | (1<<DDD3) | (1<<DDD2) | (1<<DDD1) | (1<<DDD0);
28 // State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
29 PORTD=(1<<PORTD7) | (1<<PORTD6) | (1<<PORTD5) | (1<<PORTD4) | (1<<PORTD3) | (1<<PORTD2) | (1<<PORTD1) | (1<<
       PORTD0);
30
31 // Timer/Counter 0 initialization
32 // Clock source: System Clock
33 // Clock value: Timer 0 Stopped
34 // Mode: Normal top=0xFF
35 // OC0 output: Disconnected
36 TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (0<<CS01) | (0<<CS00);
37 TCNT0=0x00;
38 OCR0=0x00;
39
40 // Timer/Counter 1 initialization
41 // Clock source: System Clock
42 // Clock value: Timer1 Stopped
43 // Mode: Normal top=0xFFFF
44 // OC1A output: Disconnected
45 // OC1B output: Disconnected
46 // Noise Canceler: Off
47 // Input Capture on Falling Edge
48 // Timer1 Overflow Interrupt: Off
49 // Input Capture Interrupt: Off
50 // Compare A Match Interrupt: Off
51 // Compare B Match Interrupt: Off
52 TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WGM10);
53 TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) | (0<<CS10);
54 TCNT1H=0x00;
55 TCNT1L=0x00;
```

```
56  ICR1H=0x00;
57  ICR1L=0x00;
58  OCR1AH=0x00;
59  OCR1AL=0x00;
60  OCR1BH=0x00;
61  OCR1BL=0x00;
62
63  // Timer/Counter 2 initialization
64  // Clock source: System Clock
65  // Clock value: Timer2 Stopped
66  // Mode: Normal top=0xFF
67  // OC2 output: Disconnected
68  ASSR=0<<AS2;
69  TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<CS21) | (0<<CS20);
70  TCNT2=0x00;
71  OCR2=0x00;
72
73  // Timer(s)/Counter(s) Interrupt(s) initialization
74  TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (0<<
        TOIE0);
75
76  // External Interrupt(s) initialization
77  // INT0: Off
78  // INT1: Off
79  // INT2: Off
80  MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
81  MCUCSR=(0<<ISC2);
82
83  // USART initialization
84  // USART disabled
85  UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) | (0<<UCSZ2) | (0<<RXB8) | (0<<TXB8);
86
87  // Analog Comparator initialization
88  // Analog Comparator: Off
89  // The Analog Comparator's positive input is
90  // connected to the AIN0 pin
91  // The Analog Comparator's negative input is
92  // connected to the AIN1 pin
93  ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);
94  SFIOR=(0<<ACME);
95
96  // ADC initialization
97  // ADC disabled
98  ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) | (0<<ADPS0);
99
100 // SPI initialization
101 // SPI disabled
102 SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) | (0<<SPR0);
103
104 // TWI initialization
105 // TWI disabled
106 TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);
107
108 while (1)
109        {
110        // Place your code here
111        PORTB++; //incrementa valor de puerto
112        PORTD--; //incrementa valor de puerto
113        delay_ms(250); //retardo que se genera para visualizar info
114        }
115 }
```

3. Circuito

Figura 2: Simulación