Instituto Politécnico Nacional

Escuela Superior de Computo

Materia:

Introducción a los microcontroladores.

Profesor:

Sanchez Aguilar Fernando

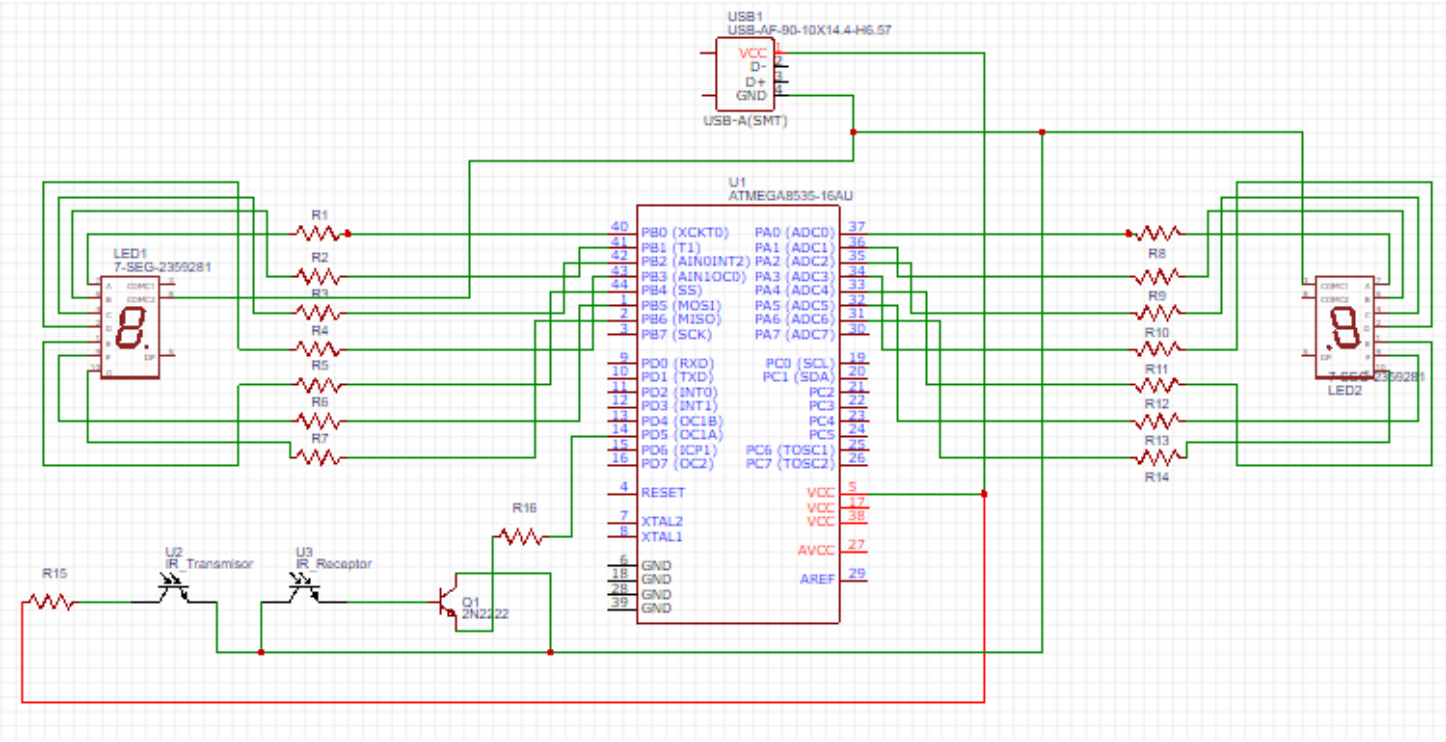
Alumnos:

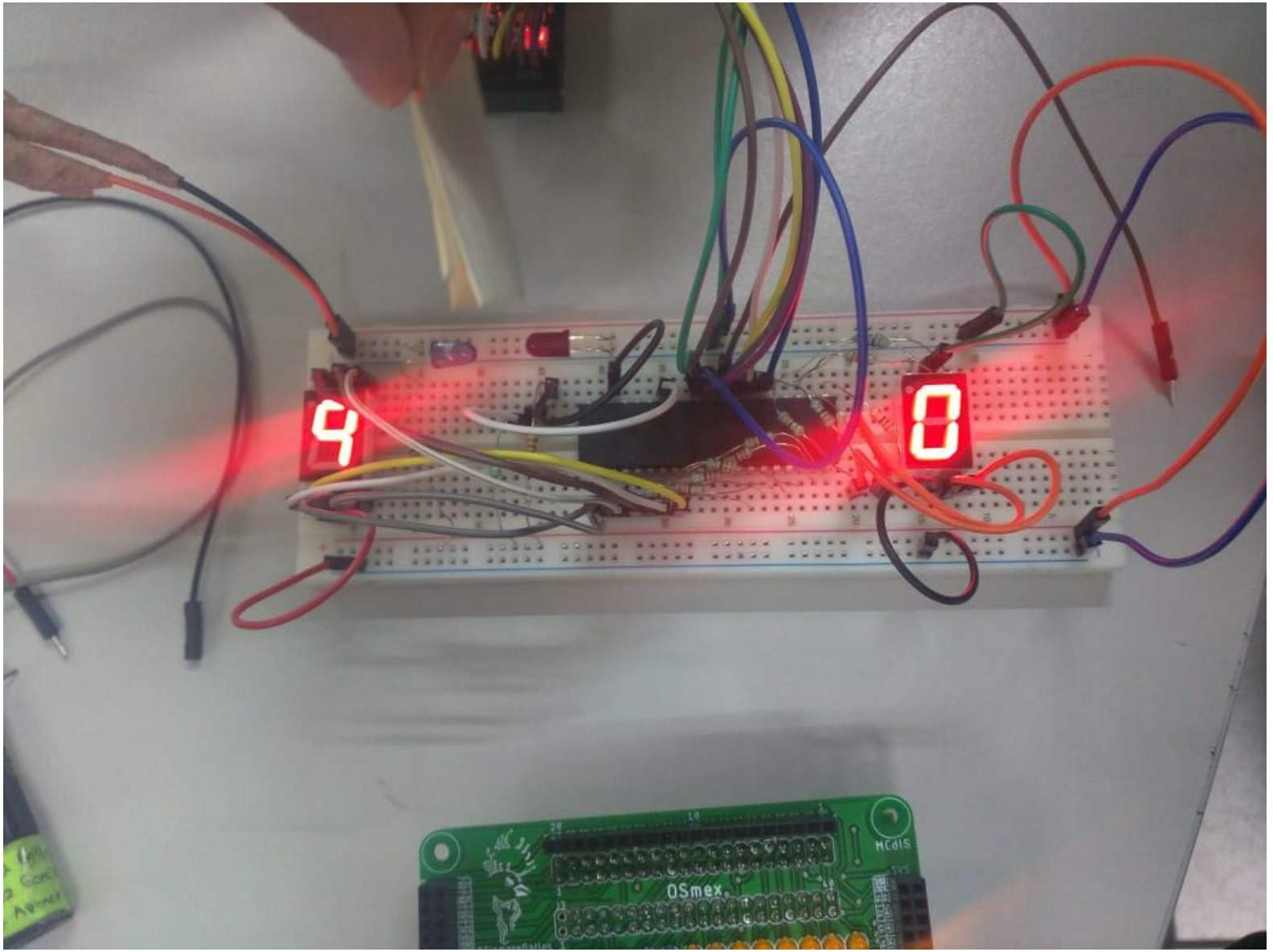
Aldavera Gallaga Iván

Lara Soto Rubén Jair

Morales Castellanos Adolfo Erik

Practica N°7


USB1
USB-AF-90-10X14.4-H6.57
VCC
D-
D+
GND
USB-A(SMT)
U1
ATMEGA8535-16AU
PB0 (XCKT0)
PB1 (T1)
PB2 (AIN0INT2)
PB3 (AIN1OC0)
PB4 (SS)
PB5 (MOSI)
PB6 (MISO)
PB7 (SCK)
PD0 (RXD)
PD1 (TXD)
PD2 (INT0)
PD3 (INT1)
PD4 (OC1B)
PD5 (OC1A)
PD6 (ICP1)
PD7 (OC2)
RESET
XTAL2
XTAL1
GND
GND
GND
GND
PA0 (ADC0)
PA1 (ADC1)
PA2 (ADC2)
PA3 (ADC3)
PA4 (ADC4)
PA5 (ADC5)
PA6 (ADC6)
PA7 (ADC7)
PC0 (SCL)
PC1 (SDA)
PC2
PC3
PC4
PC5
PC6 (TOSC1)
PC7 (TOSC2)
VCC
VCC
VCC
AVCC
AREF
LED1
7-SEG-2359281
LED2
7-SEG-2359281
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R15
R16
U2
IR_Transmisor
U3
IR_Receptor
Q1
2N2222


4
0
OSmex

```
/*****************************************************
This program was created by the
CodeWizardAVR V2.60 Evaluation
Automatic Program Generator
© Copyright 1998-2012 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com

Project :
Version :
Date    : 31/01/2019
Author  :
Company :
Comments:


Chip type               : ATmega8535L
Program type            : Application
AVR Core Clock frequency: 1,000000 MHz
Memory model            : Small
External RAM size       : 0
Data Stack size         : 128
*****************************************************/

#include <mega8535.h>
#include <delay.h>
#define boton PIND.0
bit botonp;
bit botona;
unsigned char var=0, var1=0;
const char tabla7segmentos [10]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7c,0x07,0x7f,0x6f
};

// Declare your global variables here

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=O
ut
DDRA=(1<<DDA7) | (1<<DDA6) | (1<<DDA5) | (1<<DDA4) | (1<<DDA3) | (1<<DDA2) | (1<<D
DA1) | (1<<DDA0);
// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PO
RTA2) | (0<<PORTA1) | (0<<PORTA0);

// Port B initialization
// Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=O
ut
DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) | (1<<DDB2) | (1<<D
DB1) | (1<<DDB0);
// State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PO
RTB2) | (0<<PORTB1) | (0<<PORTB0);

// Port C initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<D
DC1) | (0<<DDC0);
```

```
54. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
55. PORTC=(0<<PORTC7) | (0<<PORTC6) | (0<<PORTC5) | (0<<PORTC4) | (0<<PORTC3) | (0<<PO
    RTC2) | (0<<PORTC1) | (0<<PORTC0);
56.
57. // Port D initialization
58. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
59. DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<D
    DD1) | (0<<DDD0);
60. // State: Bit7=P Bit6=P Bit5=P Bit4=P Bit3=P Bit2=P Bit1=P Bit0=P
61. PORTD=(1<<PORTD7) | (1<<PORTD6) | (1<<PORTD5) | (1<<PORTD4) | (1<<PORTD3) | (1<<PO
    RTD2) | (1<<PORTD1) | (1<<PORTD0);
62.
63. // Timer/Counter 0 initialization
64. // Clock source: System Clock
65. // Clock value: Timer 0 Stopped
66. // Mode: Normal top=0xFF
67. // OC0 output: Disconnected
68. TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (0<<CS01) |
    (0<<CS00);
69. TCNT0=0x00;
70. OCR0=0x00;
71.
72. // Timer/Counter 1 initialization
73. // Clock source: System Clock
74. // Clock value: Timer1 Stopped
75. // Mode: Normal top=0xFFFF
76. // OC1A output: Disconnected
77. // OC1B output: Disconnected
78. // Noise Canceler: Off
79. // Input Capture on Falling Edge
80. // Timer1 Overflow Interrupt: Off
81. // Input Capture Interrupt: Off
82. // Compare A Match Interrupt: Off
83. // Compare B Match Interrupt: Off
84. TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WG
    M10);
85. TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) |
     (0<<CS10);
86. TCNT1H=0x00;
87. TCNT1L=0x00;
88. ICR1H=0x00;
89. ICR1L=0x00;
90. OCR1AH=0x00;
91. OCR1AL=0x00;
92. OCR1BH=0x00;
93. OCR1BL=0x00;
94.
95. // Timer/Counter 2 initialization
96. // Clock source: System Clock
97. // Clock value: Timer2 Stopped
98. // Mode: Normal top=0xFF
99. // OC2 output: Disconnected
100.       ASSR=0<<AS2;
101.       TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<
    CS21) | (0<<CS20);
102.       TCNT2=0x00;
103.       OCR2=0x00;
104.
105.       // Timer(s)/Counter(s) Interrupt(s) initialization
106.       TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) |
    (0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);
```

```
107.
108.       // External Interrupt(s) initialization
109.       // INT0: Off
110.       // INT1: Off
111.       // INT2: Off
112.       MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
113.       MCUCSR=(0<<ISC2);
114.
115.       // USART initialization
116.       // USART disabled
117.       UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) | (0<<U
      CSZ2) | (0<<RXB8) | (0<<TXB8);
118.
119.       // Analog Comparator initialization
120.       // Analog Comparator: Off
121.       ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) |
      (0<<ACIS1) | (0<<ACIS0);
122.       SFIOR=(0<<ACME);
123.
124.       // ADC initialization
125.       // ADC disabled
126.       ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<AD
      PS2) | (0<<ADPS1) | (0<<ADPS0);
127.
128.       // SPI initialization
129.       // SPI disabled
130.       SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA)
      | (0<<SPR1) | (0<<SPR0);
131.
132.       // TWI initialization
133.       // TWI disabled
134.       TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);
135.
136.       while (1)
137.             {
138.            if(boton==0)
139.             botona=0;
140.             else
141.             botona=1;
142.             if((botona==0)&&(botonp==1)){
143.             var++;
144.             if(var==10){
145.             var1++;
146.             var=0;
147.             }
148.             if(var1==10){
149.             var1=0;
150.             var=0;
151.             }
152.             delay_ms(40);
153.             }
154.             if((botona==1)&&(botonp==0))
155.             delay_ms(40);
156.             PORTB=tabla7segmentos[var1];
157.             PORTA=tabla7segmentos[var];
158.             botonp=botona;
159.
160.             }
161.       }
```