 Instituto Politécnico Nacional

Escuela Superior de Computo

Materia:

Introducción a los microcontroladores.

Profesor:

Sanchez Aguilar Fernando

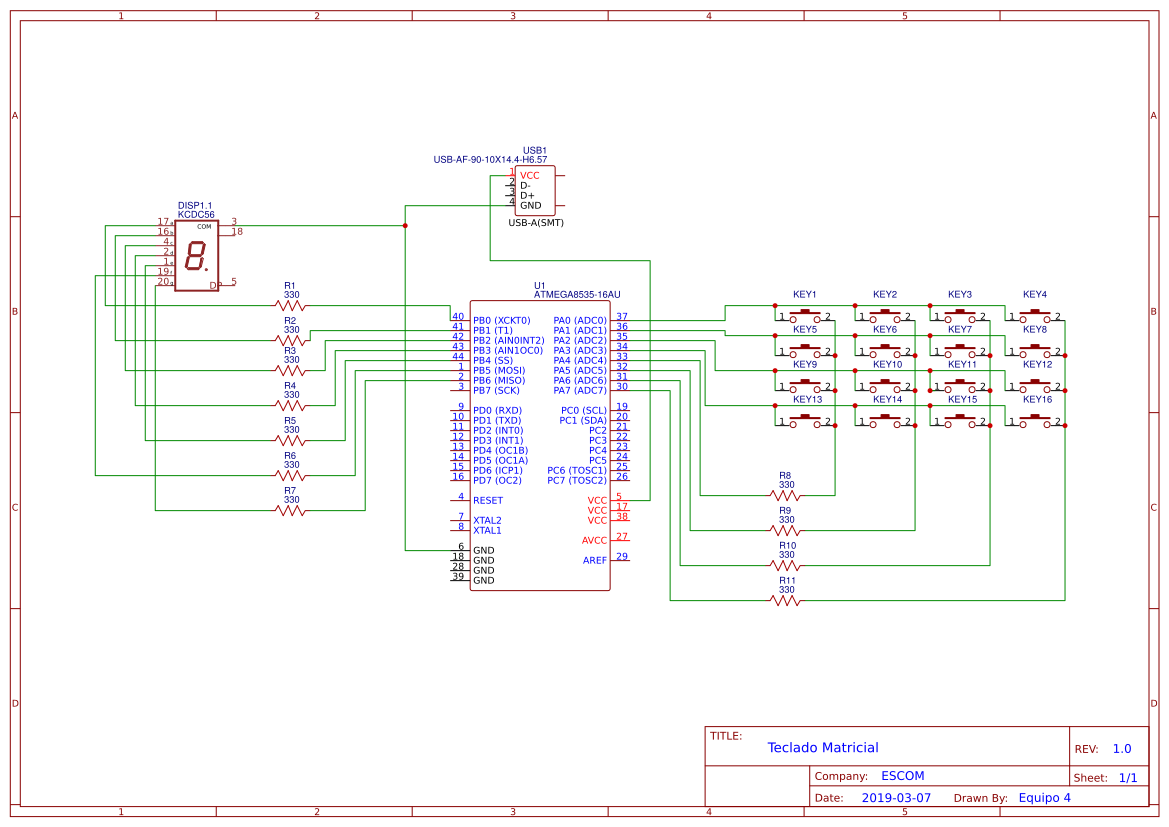
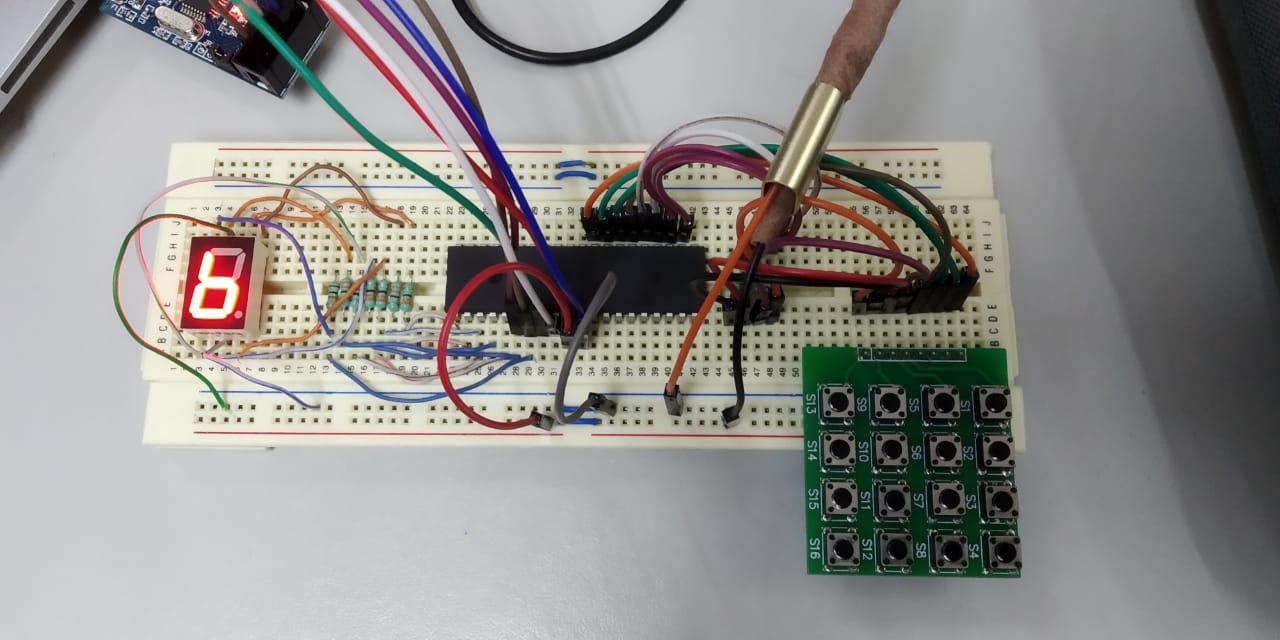
Alumnos:

Aldavera Gallaga Iván

Lara Soto Rubén Jair

Morales Castellanos Adolfo Erik

Practica N°10



Código

1. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
2. This program was created by the
3. CodeWizardAVR V2.60 Evaluation
4. Automatic Program Generator
5. © Copyright 1998-2012 Pavel Haiduc, HP InfoTech s.r.l.
6. http://www.hpinfotech.com
8. Project :
9. Version :
10. Date    : 12/02/2019
11. Author  : Equipo 4
12. Company : ESCOM
13. Comments:

16. Chip type               : ATmega8535L
17. Program type            : Application
18. AVR Core Clock frequency: 1,000000 MHz
19. Memory model            : Small
20. External RAM size       : 0
21. Data Stack size         : 128
22. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
24. #include <mega8535.h>
25. unsigned **char** tecla,lectura;
26. **const** **char** tabla7segmentos[17]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7c,0x07,0x7f,0x6f,0x77,0x7f,0x39,0x3f,0x79,0x71,0x7d};
28. // Declare your global variables here
30. **void** main(**void**)
31. {
32. // Declare your local variables here
34. // Input/Output Ports initialization
35. // Port A initialization
36. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
37. DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) | (0<<DDA0);
38. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
39. PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) | (0<<PORTA1) | (0<<PORTA0);
41. // Port B initialization
42. // Function: Bit7=Out Bit6=Out Bit5=Out Bit4=Out Bit3=Out Bit2=Out Bit1=Out Bit0=Out
43. DDRB=(1<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0);
44. // State: Bit7=0 Bit6=0 Bit5=0 Bit4=0 Bit3=0 Bit2=0 Bit1=0 Bit0=0
45. PORTB=(0<<PORTB7) | (0<<PORTB6) | (0<<PORTB5) | (0<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) | (0<<PORTB1) | (0<<PORTB0);
47. // Port C initialization
48. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=Out Bit2=Out Bit1=Out Bit0=Out
49. DDRC=(0<<DDC7) | (0<<DDC6) | (0<<DDC5) | (0<<DDC4) | (1<<DDC3) | (1<<DDC2) | (1<<DDC1) | (1<<DDC0);
50. // State: Bit7=P Bit6=P Bit5=P Bit4=P Bit3=0 Bit2=0 Bit1=0 Bit0=0
51. PORTC=(1<<PORTC7) | (1<<PORTC6) | (1<<PORTC5) | (1<<PORTC4) | (0<<PORTC3) | (0<<PORTC2) | (0<<PORTC1) | (0<<PORTC0);
53. // Port D initialization
54. // Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
55. DDRD=(0<<DDD7) | (0<<DDD6) | (0<<DDD5) | (0<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) | (0<<DDD0);
56. // State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
57. PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (0<<PORTD3) | (0<<PORTD2) | (0<<PORTD1) | (0<<PORTD0);
59. // Timer/Counter 0 initialization
60. // Clock source: System Clock
61. // Clock value: Timer 0 Stopped
62. // Mode: Normal top=0xFF
63. // OC0 output: Disconnected
64. TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (0<<CS01) | (0<<CS00);
65. TCNT0=0x00;
66. OCR0=0x00;
68. // Timer/Counter 1 initialization
69. // Clock source: System Clock
70. // Clock value: Timer1 Stopped
71. // Mode: Normal top=0xFFFF
72. // OC1A output: Disconnected
73. // OC1B output: Disconnected
74. // Noise Canceler: Off
75. // Input Capture on Falling Edge
76. // Timer1 Overflow Interrupt: Off
77. // Input Capture Interrupt: Off
78. // Compare A Match Interrupt: Off
79. // Compare B Match Interrupt: Off
80. TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WGM10);
81. TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) | (0<<CS10);
82. TCNT1H=0x00;
83. TCNT1L=0x00;
84. ICR1H=0x00;
85. ICR1L=0x00;
86. OCR1AH=0x00;
87. OCR1AL=0x00;
88. OCR1BH=0x00;
89. OCR1BL=0x00;
91. // Timer/Counter 2 initialization
92. // Clock source: System Clock
93. // Clock value: Timer2 Stopped
94. // Mode: Normal top=0xFF
95. // OC2 output: Disconnected
96. ASSR=0<<AS2;
97. TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<CS21) | (0<<CS20);
98. TCNT2=0x00;
99. OCR2=0x00;
101. // Timer(s)/Counter(s) Interrupt(s) initialization
102. TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0) | (0<<TOIE0);
104. // External Interrupt(s) initialization
105. // INT0: Off
106. // INT1: Off
107. // INT2: Off
108. MCUCR=(0<<ISC11) | (0<<ISC10) | (0<<ISC01) | (0<<ISC00);
109. MCUCSR=(0<<ISC2);
111. // USART initialization
112. // USART disabled
113. UCSRB=(0<<RXCIE) | (0<<TXCIE) | (0<<UDRIE) | (0<<RXEN) | (0<<TXEN) | (0<<UCSZ2) | (0<<RXB8) | (0<<TXB8);
115. // Analog Comparator initialization
116. // Analog Comparator: Off
117. ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) | (0<<ACIS0);
118. SFIOR=(0<<ACME);
120. // ADC initialization
121. // ADC disabled
122. ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) | (0<<ADPS0);
124. // SPI initialization
125. // SPI disabled
126. SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) | (0<<SPR0);
128. // TWI initialization
129. // TWI disabled
130. TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);
132. **while** (1)
133. {
134. //se prueba la primera columna se envia 1110
135. PORTC=0b11111110;
137. lectura=PINC&0b11110000;
138. **if**(lectura==0b11100000)
139. tecla=13;
140. **if**(lectura==0b11010000)
141. tecla=14;
142. **if**(lectura==0b10110000)
143. tecla=15;
144. **if**(lectura==0b01110000)
145. tecla=16;
147. //se prueba la segunda columna se envia 1101
148. PORTC=0b11111101;
150. lectura=PINC&0b11110000;
151. **if**(lectura==0b11100000)
152. tecla=9;
153. **if**(lectura==0b11010000)
154. tecla=10;
155. **if**(lectura==0b10110000)
156. tecla=11;
157. **if**(lectura==0b01110000)
158. tecla=12;
160. //se prueba la tercera columna se envia 1011
162. PORTC=0b11111011;
164. lectura=PINC&0b11110000;
165. **if**(lectura==0b11100000)
166. tecla=5;
167. **if**(lectura==0b11010000)
168. tecla=6;
169. **if**(lectura==0b10110000)
170. tecla=7;
171. **if**(lectura==0b01110000)
172. tecla=8;
174. //se prueba la cuarta columna se envia 0111
176. PORTC=0b11110111;
178. lectura=PINC&0b11110000;
179. **if**(lectura==0b11100000)
180. tecla=1;
181. **if**(lectura==0b11010000)
182. tecla=2;
183. **if**(lectura==0b10110000)
184. tecla=3;
185. **if**(lectura==0b01110000)
186. tecla=4;
188. PORTB=tabla7segmentos[tecla];
189. };
190. }