| Resultado de imagen para logo ipn | Instituto Politécnico Nacional  Escuela Superior de Cómputo | Resultado de imagen para logo escom |
| --- | --- | --- |

**Ejercicio 2: Decodificador**

**Microcontroladores**

Grupo: 3CM16

Alumnos:

Cazares Martínez Maximiliano

Lozano Rivera Oscar

Ramos Nieves Adrian

Profesor.

Pérez Pérez José Juan

**Ejercicio**

Realizar un decodificador binario-hexadecimal utilizando el microcontrolador ATMEGA8535, la representación decimal se mostrará en un display de 7 segmentos cátodo común.

| Entrada | Salida Decimal | Salida a Display |
| --- | --- | --- |
| 0000 | 0 | 0111111 ► 3f |
| 0001 | 1 | 0000110 ► 06 |
| 0010 | 2 | 1011011 ► 5b |
| 0011 | 3 | 1001111 ► 4f |
| 0100 | 4 | 1100110 ► 66 |
| 0101 | 5 | 1101101 ► 6d |
| 0110 | 6 | 1111101 ► 7d |
| 0111 | 7 | 0000111 ► 07 |
| 1000 | 8 | 1111111 ► 7f |
| 1001 | 9 | 1101111 ► 6f |
| 1010 | 10 | 1110111 ► 77 |
| 1011 | 11 | 1111100 ► 7C |
| 1100 | 12 | 0111001 ► 39 |
| 1101 | 13 | 1011110 ► 5E |
| 1110 | 14 | 1111001 ► 79 |
| 1111 | 15 | 1110001 ► 71 |

**Código de AVR Studio**

.include "m8535def.inc"

ser r16

out ddra,r16

out portb,r16

ldi r16,$3f

mov r0,r16

ldi r16,6

mov r1,r16

ldi r16,$5b

mov r2,r16

ldi r16,$4f

mov r3,r16

ldi r16,$66

mov r4,r16

ldi r16,$6d

mov r5,r16

ldi r16,$7d

mov r6,r16

ldi r16,7

mov r7,r16

ldi r16,$7f

mov r8,r16

ldi r16,$6f

mov r9,r16

ldi r16,$77

mov r10,r16

ldi r16,$7C

mov r11,r16

ldi r16,$39

mov r12,r16

ldi r16,$5E

mov r13,r16

ldi r16,$79

mov r14,r16

ldi r16,$71

mov r15,r16

clr ZH

uno:

ldi ZL,0

in r16,pinb

add ZL,r16

ld r16,Z

out porta,r16

rjmp uno

**Pruebas en ProteusDiagrama, Esquemático

Descripción generada automáticamenteDiagrama, Esquemático

Descripción generada automáticamente**

**Diagrama, Esquemático

Descripción generada automáticamenteDiagrama, Esquemático

Descripción generada automáticamenteDiagrama, Esquemático

Descripción generada automáticamente**

**Diagrama, Esquemático

Descripción generada automáticamente**

**Conclusiones**

**Maximiliano Cazares Martínez.** Para la realización del presente ejercicio nos percatamos que era necesario hacer uso de más registros del microcontrolador y es por esto que la instrucción move fue de vital importancia, ya que sin ella no habríamos sido capaces de realizar dicho ejercicio. Además, entendímos en mayor profundidad el uso tanto del AVR Studio y de Proteus para poder trabajar con microcontroladores.

**Oscar Lozano Rivera:** Durante la realización del ejercicio como equipo fuimos capaces de trabajar en conjunto con los programas AVR Studio y Proteus para completar la simulación de un decodificador binario a hexadecimal a través de su implementación con instrucciones a nivel ensamblador programadas en el microcontrolador virtual en proteus ATMEGA 8535.

**Adrian Ramos Nieves:** En el presente ejercicio se realizó un decodificador de 7 segmentos de cátodo común, del cual se sacó su tabla para poder convertir a hexadecimal los números del 1 al 15. Se comprendió el uso de proteus junto al AVR studio, además de que ideamos cómo usar la instrucción move para poder mostrar en el display todos los números, por lo que quedo mas claro como se implementan las nuevas instrucciones utilizadas del micro.