



**INSTITUTO POLITÉCNICO
NACIONAL**
ESCUELA SUPERIOR DE CÓMPUTO



“Decodificador BCD a 7 Segmentos”

Integrantes del Equipo:

Contreras Cardoso Adolfo

Martínez Alvarado Bryan Alexis

Maya Martínez Alonso Rubén

Pérez Gómez Santiago

Grupo:

3CM15

Profesor:

Ing. José Juan Pérez Pérez

Asignatura:

Introducción a los Microcontroladores

Planteamiento del Problema

Se debe rediseñar el código correspondiente al decodificador de BCD a 7 segmentos, el cual fue creado y simulado durante la clase, para que funcione desde el valor 0 hasta el valor F.

Desarrollo

Para realizar la decodificación de BCD a 7 segmentos se almacenó en los registros altos los valores en 7 segmentos los números del 0 al 9, de manera que cuando se lea un dato de entrada por medio del registro apuntador Z se le sume \$0014, esto para que el apuntador Z contenga la dirección de memoria del dato a mostrar en el display 7 segmentos.

Lo que ahora se debe hacer es rediseñar todo el código de manera que se pueda almacenar los números del 0 al F en otros registros y que el resultado sea el adecuado. La visualización en el display sería como se describe a continuación:

Cifra	gfedcba	Display
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7
1000	01111111	8
1001	01101111	9
1010	01110111	A
1011	01111100	b
1100	00111001	C
1101	01011110	d
1110	01111001	E
1111	01110001	F

Display Cátodo de 7 Segmentos

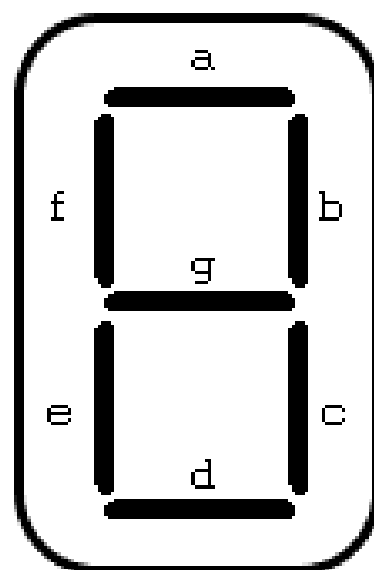


Figura 1. Imagen con la conversión BCD a 7 segmentos (versión adaptada) [1]
A continuación se muestra el diagrama de flujo correspondiente al algoritmo:

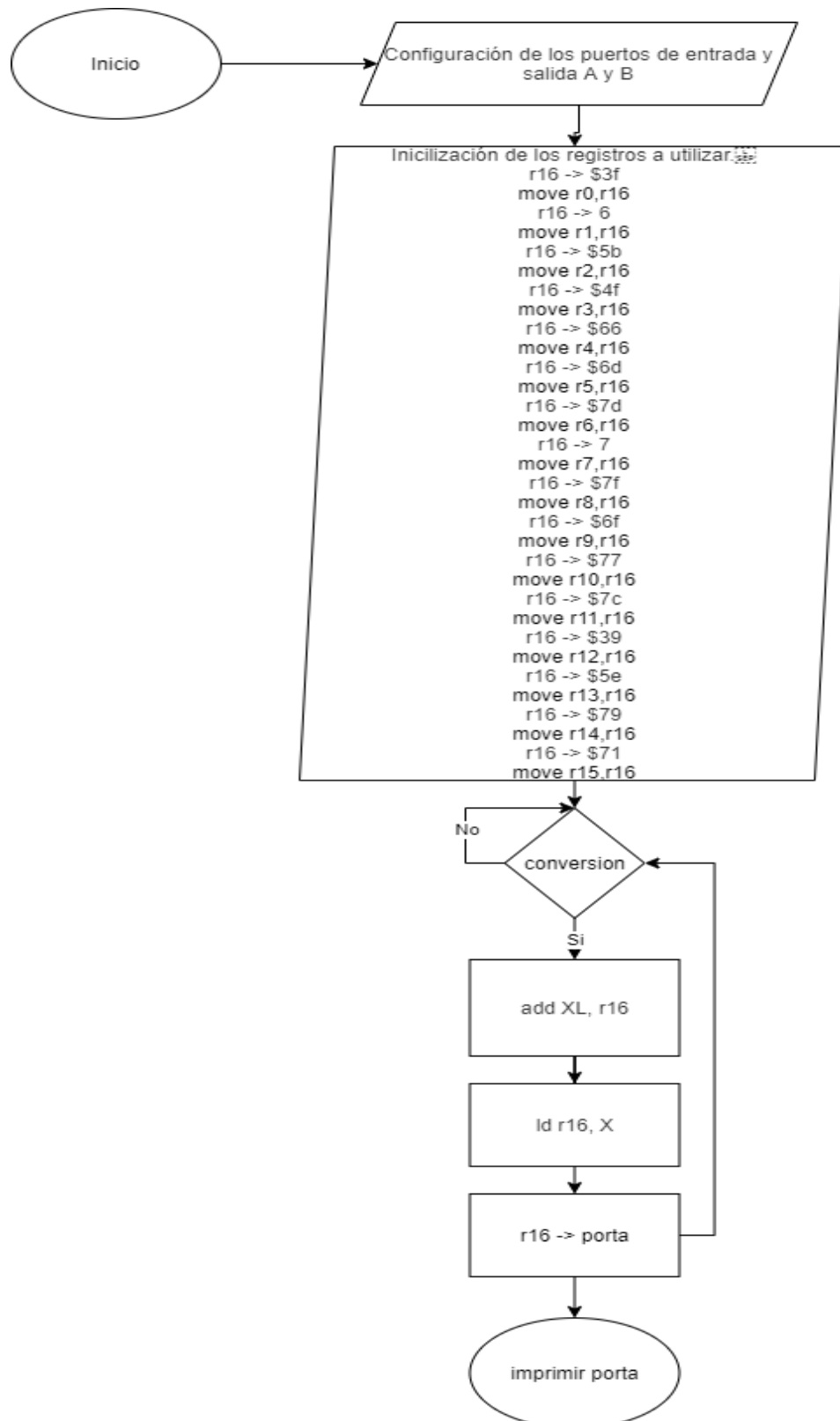


Figura 2. Diagrama de Flujo

El respectivo código en ensamblador es el siguiente:

```
;Decodificador BCD a 7 segmentos ;numero 4
.include "m8535def.inc"          ldi r16,$66
                                  mov r4,r16

;configurando puerto a salida    ;numero 5
;y el puerto b entrada          ldi r16,$6d
ser r16                          mov r5,r16
out ddra,r16
out portb,r16

                                  ;numero 6
clr XH                           ldi r16,$7d
                                  mov r6,r16

;numero 0                        ;numero 7
ldi r16,$3f                      ldi r16,7
mov r0,r16                       mov r7,r16

;numero 1                       ;numero 8
ldi r16,6                        ldi r16,$7f
mov r1,r16                       mov r8,r16

;numero 2                       ;numero 9
ldi r16,$5b                      ldi r16,$6f
mov r2,r16                       mov r9,r16

;numero 3                       ;letra A
ldi r16,$4f                      ldi r16,$77
mov r3,r16                       mov r10,r16
```

Figura 3. Primera Sección de Código

```

;letra B
ldi r16,$7c
mov r11,r16

;letra C
ldi r16,$39
mov r12,r16

;letra D
ldi r16,$5e
mov r13,r16

;letra E
ldi r16,$79
mov r14,r16

;letra F
ldi r16,$71
mov r15,r16







conversion:
clr XL
in r16,pinb
add XL,r16
ld r16,X
out porta,r16
rjmp conversion

```

Figura 4. Segunda Sección de Código







Para la parte de simulación en AVR Studio, se propondrán algunos ejemplos:

- Número 7:

Name	Add...	Value	Bits
 PORTA	0x1B...	0x07	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRA	0x1A...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINA	0x19...	0x00	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PORTB	0x18...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRB	0x17...		<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINB	0x16...	0x07	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>







Register			×
R00= 0x3F	R01= 0x06	R02= 0x5B	
R03= 0x4F	R04= 0x66	R05= 0x6D	
R06= 0x7D	R07= 0x07	R08= 0x7F	
R09= 0x6F	R10= 0x77	R11= 0x7C	
R12= 0x39	R13= 0x5E	R14= 0x79	
R15= 0x71	R16= 0x07	R17= 0x00	
R18= 0x00	R19= 0x00	R20= 0x00	
R21= 0x00	R22= 0x00	R23= 0x00	
R24= 0x00	R25= 0x00	R26= 0x07	
R27= 0x00	R28= 0x00	R29= 0x00	
R30= 0x00	R31= 0x00		

- Letra A:

Name	Add...	Value	Bits
 PORTA	0x1B...	0x77	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRA	0x1A...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINA	0x19...	0x07	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PORTB	0x18...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRB	0x17...		<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINB	0x16...	0x0A	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Register				×
R00= 0x3F	R01= 0x06	R02= 0x5B		
R03= 0x4F	R04= 0x66	R05= 0x6D		
R06= 0x7D	R07= 0x07	R08= 0x7F		
R09= 0x6F	R10= 0x77	R11= 0x7C		
R12= 0x39	R13= 0x5E	R14= 0x79		
R15= 0x71	R16= 0x77	R17= 0x00		
R18= 0x00	R19= 0x00	R20= 0x00		
R21= 0x00	R22= 0x00	R23= 0x00		
R24= 0x00	R25= 0x00	R26= 0x0A		
R27= 0x00	R28= 0x00	R29= 0x00		
R30= 0x00	R31= 0x00			

- Letra F:

Name	Add...	Value	Bits
 PORTA	0x1B...	0x71	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRA	0x1A...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINA	0x19...	0x77	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PORTB	0x18...	0xFF	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 DDRB	0x17...		<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
 PINB	0x16...	0x0F	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Register				×
R00= 0x3F	R01= 0x06	R02= 0x5B		
R03= 0x4F	R04= 0x66	R05= 0x6D		
R06= 0x7D	R07= 0x07	R08= 0x7F		
R09= 0x6F	R10= 0x77	R11= 0x7C		
R12= 0x39	R13= 0x5E	R14= 0x79		
R15= 0x71	R16= 0x71	R17= 0x00		
R18= 0x00	R19= 0x00	R20= 0x00		
R21= 0x00	R22= 0x00	R23= 0x00		
R24= 0x00	R25= 0x00	R26= 0x0F		
R27= 0x00	R28= 0x00	R29= 0x00		
R30= 0x00	R31= 0x00			

Conclusiones Individuales

Contreras Cardoso Adolfo

En esta práctica aprendimos a utilizar las subrutinas, las cuales son equivalentes a las funciones que conocemos de los lenguajes de programación de alto nivel. También aprendimos a generar retardos para poder utilizarlos para codificar contadores, por ejemplo. Para poder realizar la práctica también recordamos algunos conceptos que habíamos visto en ejercicios previos, por ejemplo hacer un decodificador para mostrar los números en los displays.

Martínez Alvarado Bryan Alexis

Durante el desarrollo de esta práctica hemos podido aplicar los conocimientos adquiridos en las clases de teoría de la materia Introducción a los microcontroladores, hemos podido reafirmar los conocimientos adquiridos y darnos cuenta cómo es que trabajan en la parte práctica.

Se utilizó el microcontrolador ATmega8535 e hicimos uso de funciones e instrucciones como son ldi, mov, in, add, out, ld, etc, de las cuales, logramos implementarlas para lograr una decodificación BCD a 7 segmentos y además pudimos ver y analizar su comportamiento cuando son empleadas.

Maya Martínez Alonso Rubén

En el desarrollo hemos puesto en práctica todo lo aprendido en esta y otras materias como FDD. A lo largo de toda la carrera hemos programado decodificadores, cada vez de más alto nivel a mas bajo, es

importante entender cómo funcionan desde el más bajo nivel, es por eso que aprender a hacerlo con un microcontrolador ayuda a comprender todo desde muy bajo nivel. También nos ayuda a comprender que con un microcontrolador se pueden hacer miles de cosas diferentes y que esto es algo básico aún.

Pérez Gómez Santiago

El diseño e implementación del decodificador BCD a 7 segmentos con las letras A a la F a simple vista denotaba un alto nivel de dificultad, sin embargo, el manejo del manual en conjunto con anteriores sistemas facilitó en gran manera el encontrar una posible solución al problema propuesto. Personalmente, partir de un código visto en clase resulta más interesante y sencillo de visualizar, con lo cual se consigue comprender mejor su funcionamiento.

Referencias

[1] PROGRAMACIÓNSIEMENS. “Ejemplo de codificación de un display de 7 segmentos”. Recuperada de: <https://programacionsiemens.com/display-de-7-segmentos-step-7/>