



CENTRO DE CIENCIAS BÁSICAS
DEPARTAMENTO DE SISTEMAS ELECTRÓNICOS
ORGANIZACIÓN COMPUTACIONAL
4° "A"

PRÁCTICA 3

M. en CC. Juan Pedro Cisneros Santoyo

Alumno: Joel Alejandro Espinoza Sánchez

Fecha de Entrega: Aguascalientes, Ags., 27 de mayo de 2020

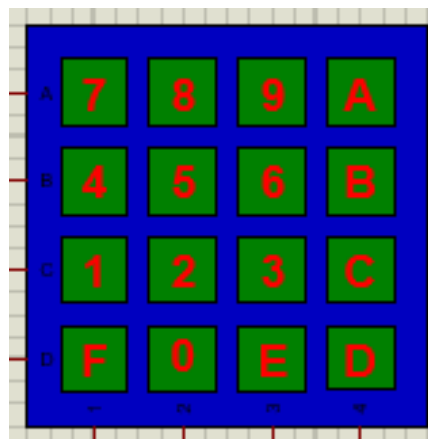
Práctica 3

Objetivo

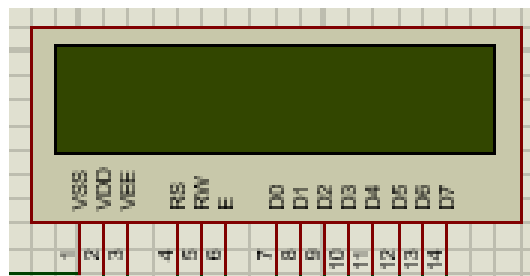
La manipulación básica del microcontrolador 8051 (en cualquiera de sus variantes).
Utilizar el software Keil μ Vision para realizar el código en lenguaje ensamblador.
Hacer uso de la LCD

Antecedentes

Durante clases se han revisado distintos métodos de entrada y salida de datos que interactúen con el microcontrolador 8051. Hemos trabajado en la evaluación parcial con un teclado matricial el cual se modificó para generar el siguiente:



Y añadido a esto, se ha revisado un dispositivo llamado LCD, la cual es un display de cristal líquido, que permite ir más allá que el display de siete segmentos, ya que permite representar el código ASCII entero (que es un conjunto de caracteres muy extenso) y tiene la forma siguiente:



Debido a que en trabajos anteriores, se nos había encargado la forma de conectar el arreglo matricial y el display de 7 segmentos mediante el microcontrolador, esta práctica consistiría en la manipulación del código existente para poder conectar ahora el teclado con el LCD de modo que, al presionar una tecla de éste, se desplegara en la pantalla del LCD la letra correspondiente a la presionada en el teclado.

Pregunta de Investigación

¿De qué manera se puede modificar el código dado de modo que pueda usarse un código prerrealizado para un display de siete segmentos pero que sea funcional en un LCD?

Predicción

Creo que podemos usar de base cualquiera de los códigos hechos, ya sea el de prueba de LCD o el del teclado con el display de siete segmentos. Personalmente pienso que será más fácil implementarlo usando de base el de prueba de LCD para ir insertando los anexos correspondientes del código del teclado en los sitios correspondientes.

Materiales

Una computadora con ciertas especificaciones.

El software Keil μ Vision.

El software Proteus 8.8.

Método (Variables)

Dependiente: El LCD mostrando la letra correspondiente.

Independiente: La tecla oprimida en el teclado matricial y los bits generados a su puerto.

Controlada: El circuito elaborado y el programa en ensamblador.

Seguridad

No existen riesgos físicos en la elaboración de esta práctica (a excepción de las medidas de precaución con el uso de una computadora).

Procedimiento

1.- Se realizó en el software Proteus 8.8 el siguiente circuito:

```
inicio:
ACALL iniciar_lcd
```

```

MOV P1,#11111110B
MOV R1,P1
CJNE R1,#11111110B,numero
MOV P1,#11111101B
MOV R1,P1
CJNE R1,#11111101B,numero
MOV P1,#11111011B
MOV R1,P1
CJNE R1,#11111011B,numero
MOV P1,#11110111B
MOV R1,P1
CJNE R1,#11110111B,numero
SJMP inicio

```

```

mostrar_lcd:
;Regreso a casa
ACALL en_h
MOV A,#02H
MOV datos,A
ACALL en_l

```

```

numero:
MOV P3,P1
JNB P3.0,F1
JNB P3.1,F2
JNB P3.2,F3
JNB P3.3,F4

```

```

continue:
MOV DPTR,#tabla
MOVC A,@A+DPTR
ACALL en_h
ACALL rs_h
MOV datos,A
ACALL en_l
ACALL delay
ACALL delay
SJMP inicio

```

```

F1:
MOV A,#00H
ACALL compare

```

```

F2:

```

```
MOV A,#04H
ACALL compare
```

```
F3:
MOV A,#08H
ACALL compare
```

```
F4:
MOV A,#0CH
ACALL compare
```

```
compare:
MOV P3,R1
JNB P3.4,C1
JNB P3.5,C2
JNB P3.6,C3
JNB P3.7,C4
RET
```

```
C1:
ADD A,#00H
SJMP continue
```

```
C2:
ADD A,#01H
SJMP continue
```

```
C3:
ADD A,#02H
SJMP continue
```

```
C4:
ADD A,#03H
SJMP continue
```

```
iniciar_lcd:
ACALL rw_l
ACALL rs_l
```

```
;Comienza la configuración de encendido de la LCD
ACALL en_h
MOV A,#38H
MOV datos,A
ACALL en_l
```

```
ACALL en_h
MOV A,#38H
MOV datos,A
ACALL en_l
```

```
ACALL en_h
MOV A,#38H
MOV datos,A
ACALL en_l
```

```
ACALL en_h
MOV A,#38H
MOV datos,A
ACALL en_l
```

```
;Apaga la pantalla
ACALL en_h
MOV A,#08H
MOV datos,A
ACALL en_l
```

```
;Limpiar pantalla
ACALL en_h
MOV A,#01H
MOV datos,A
ACALL en_l
```

```
;Modo de entrada
ACALL en_h
MOV A,#06H
MOV datos,A
ACALL en_l
```

```
;Encender la pantalla
ACALL en_h
MOV A,#0FH
MOV datos,A
ACALL en_l
RET
```

```
en_l:
CLR en
ACALL delay
```

RET

en_h:
SETB en
ACALL delay
RET

rs_l:
CLR rs
ACALL delay
RET

rs_h:
SETB rs
ACALL delay
RET

rw_l:
CLR rw
ACALL delay
RET

rw_h:
SETB rw
ACALL delay
RET

delay:
MOV R6,#65H ;1
d1:
MOV R7,#49H ;101
d2:
DJNZ R7,d2 ;2*101*73
DJNZ R6,d1 ;2*101
RET ;15,052

tabla:
DB 37H ;7
DB 38H ;8
DB 39H ;9
DB 41H ;A

DB 34H ;4
DB 35H ;5


```
DB 36H ;6  
DB 42H ;B
```

```
DB 31H ;1  
DB 32H ;2  
DB 33H ;3  
DB 43H ;C
```

```
DB 46H ;F!  
DB 30H ;0  
DB 45H ;E!  
DB 44H ;D!
```

END

Éste fue de extrema utilidad. El código consigue el despliegue tras la conversión de datos de la tabla a hexadecimal señalados por el código ASCII, pero el tiempo que deja el carácter en pantalla es relativamente poco. Podría solucionarse con un retraso mayor.

Conclusiones

Este programa nos fue de utilidad para poder investigar un poco más usando la lógica entre dos programas elaborados previamente. Tuvimos el ejemplo en clase y con algunas pistas de cómo desarrollar esta práctica y con un poco de indagación conseguimos terminar esta práctica.

Como propuesta de mejora he de reconocer que el circuito no le da tiempo a mostrar el carácter un rato, pues el retraso implementado es de 15 milisegundos, cuando podría implementarse un retraso aún mayor, sin embargo, el funcionamiento de “traducción” del teclado al LCD funciona correctamente.

Referencias

- Anónimo. (2005). Knight Rider. Mayo 7, 2020, de Wikipedia Sitio web: https://es.wikipedia.org/wiki/Knight_Rider
- Anónimo. (2007). Microcontrolador. Mayo 2, 2020, de Wikipedia Sitio web: <https://es.wikipedia.org/wiki/Microcontrolador>
- Anónimo. (2013). Microcontrolador. Mayo 3, 2020, de EcuRed Sitio web: <https://www.ecured.cu/Microcontrolador>
- Pahrami, B. (2005). Arquitectura de Computadoras. México: McGraw Hill.
- Wackerly, J. (2008). Diseño Digital. Principios y prácticas. México: Pearson.

Anexos

Anexo 1: Código del programa prueba del LCD

en equ P2.2

rs equ P2.0

rw equ P2.1

datos equ P0

ORG 0000H

main:

ACALL iniciar_lcd

mostrar_lcd:

;Regreso a casa

ACALL en_h

MOV A,#02H

MOV datos,A

ACALL en_l

ACALL en_h

ACALL rs_h

MOV A,#30H

MOV datos,A

ACALL en_l

aqui:

AJMP aqui

iniciar_lcd:

ACALL rw_l

ACALL rs_l

;Comienza la configuración de encendido de la LCD

ACALL en_h

MOV A,#38H

MOV datos,A

ACALL en_l

ACALL en_h

MOV A,#38H

MOV datos,A

ACALL en_l

ACALL en_h

MOV A,#38H

```
MOV datos,A
ACALL en_1
```

```
ACALL en_h
MOV A,#38H
MOV datos,A
ACALL en_1
```

```
;Apaga la pantalla
ACALL en_h
MOV A,#08H
MOV datos,A
ACALL en_1
```

```
;Limpiar pantalla
ACALL en_h
MOV A,#01H
MOV datos,A
ACALL en_1
```

```
;Modo de entrada
ACALL en_h
MOV A,#06H
MOV datos,A
ACALL en_1
```

```
;Encender la pantalla
ACALL en_h
MOV A,#0FH
MOV datos,A
ACALL en_1
RET
```

```
en_1:
CLR en
ACALL delay
RET
```

```
en_h:
SETB en
ACALL delay
RET
```

```
rs_1:
```

```

CLR rs
ACALL delay
RET

rs_h:
SETB rs
ACALL delay
RET

rw_l:
CLR rw
ACALL delay
RET

rw_h:
SETB rw
ACALL delay
RET

delay:
MOV R6,#65H           ;1
d1:
MOV R7,#49H           ;101
d2:
DJNZ R7,d2            ;2*101*73
DJNZ R6,d1            ;2*101
RET                   ;15,052

```

END

Anexo 2: Código del programa que enlaza el teclado matricial con el display de siete segmentos

ORG 0000H

```

inicio:
MOV P2,#11111110B
MOV R1,P2
CJNE R1,#11111110B,numero
MOV P2,#11111101B
MOV R1,P2
CJNE R1,#11111101B,numero
MOV P2,#11111011B
MOV R1,P2
CJNE R1,#11111011B,numero

```

```
MOV P2,#11110111B
MOV R1,P2
CJNE R1,#11110111B,numero
SJMP inicio
```

```
numero:
MOV P3,P2
JNB P3.0,F1
JNB P3.1,F2
JNB P3.2,F3
JNB P3.3,F4
```

```
continue:
MOV DPTR,#tabla
MOVC A,@A+DPTR
MOV P0,A
SJMP inicio
```

```
F1:
MOV A,#00H
ACALL compare
```

```
F2:
MOV A,#04H
ACALL compare
```

```
F3:
MOV A,#08H
ACALL compare
```

```
F4:
MOV A,#0CH
ACALL compare
```

```
compare:
MOV P3,R1
JNB P3.4,C1
JNB P3.5,C2
JNB P3.6,C3
JNB P3.7,C4
RET
```

```
C1:
ADD A,#00H
```

SJMP continue

C2:

ADD A,#01H

SJMP continue

C3:

ADD A,#02H

SJMP continue

C4:

ADD A,#03H

SJMP continue

tabla:

DB 78H ;7

DB 00H ;8

DB 18H ;9

DB 08H ;A

DB 19H ;4

DB 12H ;5

DB 02H ;6

DB 03H ;B

DB 79H ;1

DB 24H ;2

DB 30H ;3

DB 46H ;C

DB 0EH ;F!

DB 40H ;0

DB 06H ;E!

DB 21H ;D!

END