INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

Introducción a los Microcontroladores

Reportes de Prácticas | 1 – 4

Chávez Chávez Ángel Alexis

Hernández López Pamela

Flores Alvarado Diego Armando

Profesor | Pérez Pérez José Juan

3CM6

Semestre 18/19-1

Reporte para el viernes, 6 de abril de 2018

Índice de Contenido

Reporte de Práctica 1

Introducción

En esta práctica el alumno conocerá el entorno del software Avr Studio empleando dos programas en lenguaje ensamblador, el cual permitirá al alumno entender la lógica que emplea el circuito integrado ATMega8535.

Objetivo

El alumno se familiariza con el lenguaje Ensamblador para que pueda realizar las futuras prácticas a lo largo del curso

Material Empleado

- AVR Studio
- 2 ProtoBoard
- Circuito integrado ATMega16
- Cables para conexiones
- 1 dip switch
- Resistencias de 100 ohm
- Leds
- Fuente de alimentación

Desarrollo

Para esta práctica se emplearon los siguientes códigos:

```
1. .include "m8535def.inc"
2.
3.          ldi R16, $5A
4.          ldi R17, $79
5.          mov R18, R16
6.          mov R19, R17
7.
8.          loop:   nop
9.                          inc R16
10.                               dec R17
11.                               rjmp loop
```

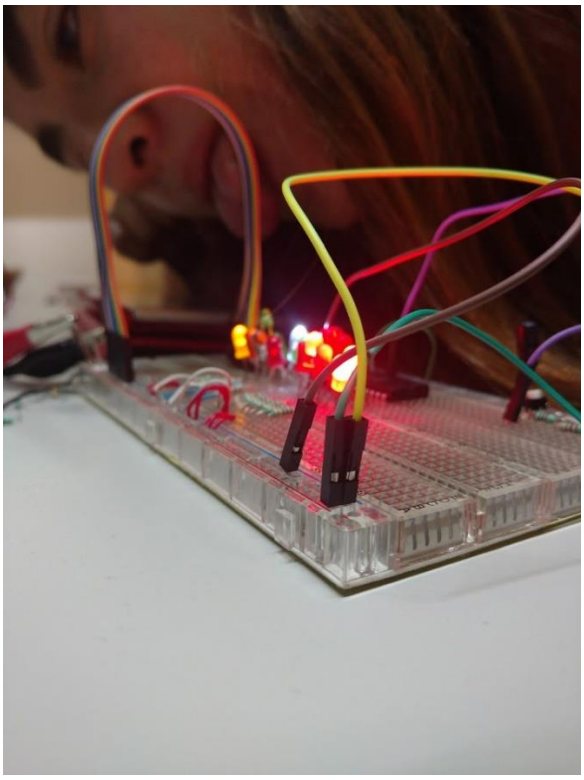
En este código, se escriben algunos valores en la memoria que no son perceptibles en el circuito armado para la práctica, pero sí se puede ver a la hora de compilarlo en AVR Studio. Seguido de, se usó el siguiente código:

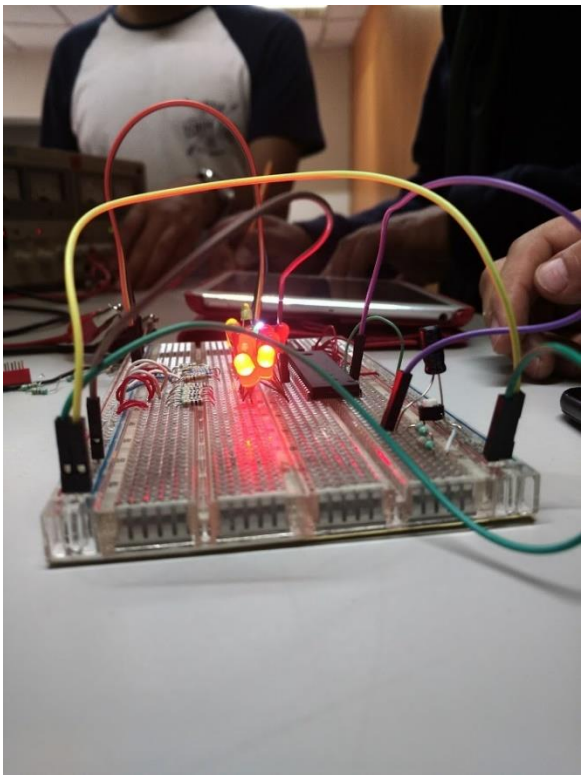
```
1. .include "m8535def.inc"
2.
3.          ser R16
4.          out DDRA, R16
5.          out DDRC, R16
```

```
6.          out PORTD, R16
7.
8. ciclo:   in R17, PIND
9.                  out PORTA, R17
10.                       com R17
11.                       out PORTC, R17
12.                       rjmp ciclo
```

Con este código ya podemos observar la salida, si hay algún valor escrito en la memoria se va a reflejar en los leds.

Captura de Evidencias





Conclusiones

Chávez Chávez Ángel Alexis

En la primera práctica del curso observamos el funcionamiento del ATMega16 mediante el uso de un loop, además de probar la salida de datos por el puerto B con leds.

Hernández López Pamela

Esta práctica fue bastante interesante ya que con esta nuestra curva de aprendizaje con el lenguaje ensamblador comenzó, no estábamos seguros de qué queríamos hacer, una vez programado el código en el microcontrolador no sabíamos que resultados debíamos ver en el circuito, pero al final preguntamos y poco a poco fuimos entendiendo más. Aprendí muchas cosas nuevas en esta clase.

Flores Alvarado Diego Armando

Una introducción a ensamblador. Nunca había visto algo de este lenguaje de programación y me lo hacían ver como algo muy difícil, pues no lo es, al menos en esta práctica pude entender lo que tecleamos y programamos en el microcontrolador. Una buena introducción para conocer con lo que trabajaremos todo el semestre.

Reporte de Práctica 2

Introducción

En esta práctica el alumno debe hacer un circuito con un microcontrolador programado un comparador de dos valores introducidos a través un dip switch, mostrando el valor mayor en un lugar y el menor en otro.

Objetivo

El alumno entenderá cómo invertir los nibble con un microcontrolador programado.

Material Empleado

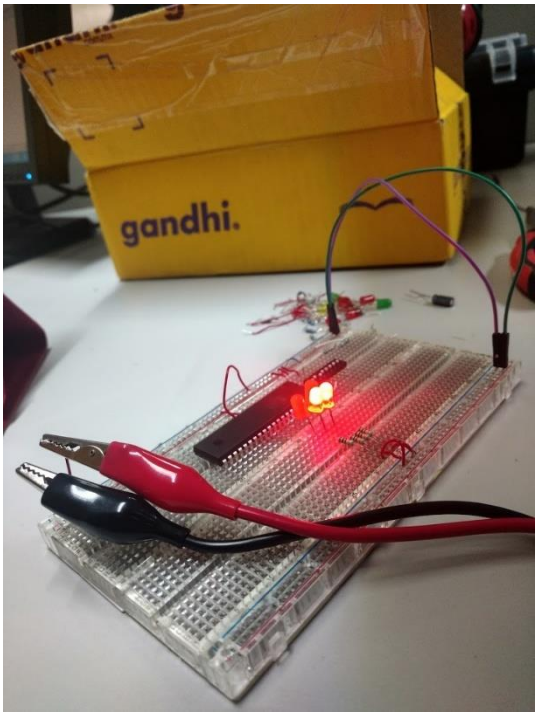
- AVR Studio
- 2 ProtoBoard
- Circuito integrado ATMega16
- Cables para conexiones
- 1 dip switch
- Resistencias de 100 ohm
- Leds
- Fuente de alimentación

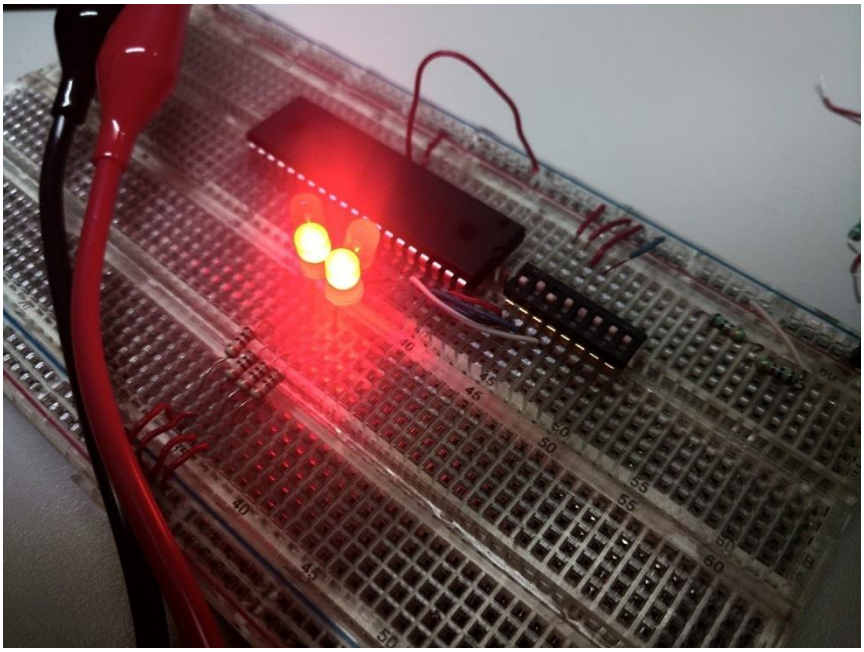
Desarrollo

Para esta práctica se implementó el siguiente código:

```
1. .include "m8535def.inc"
2.
3.          .def aux = R16
4.          ldi aux, $f0
5.          out DDRA, aux
6.          swap aux
7.          out PORTA, aux
8.
9. lee: in aux, PINA
10.              swap aux
11.              ori aux, $0f
12.              out PORTA, aux
13.              rjmp lee
```

Captura de Evidencias





Conclusiones

Chávez Chávez Ángel Alexis

En la práctica 2 implementamos un código que nos permitió intercambiar el valor de los nibbles que se tienen de entrada en el puerto D.

Hernández López Pamela

En esta práctica las cosas se complicaron un poco ya que tuvimos algunos problemas con el microcontrolador puesto que el nuestro es diferente al que usa el profesor, igual hubo problemas con las salidas, pero al final todo salió bien y se vio correctamente a la hora de probar el circuito. Fue un tanto difícil pero entretenido

Flores Alvarado Diego Armando

Con algunas dificultades, pero logramos salir adelante, tuvimos algunos problemas a la hora de programar el microcontrolador y con algunos otros detalles del circuito. Al final se logró el objetivo de la práctica y nos queda como retroalimentación de lo visto en clase.

Reporte de Práctica 3

Introducción

En esta práctica el alumno debe hacer un circuito con un microcontrolador programado que decodifique por medio de un dip switch los números del 1 al 9 en un display.

Objetivo

Entender el uso de valores (en este caso números) en hexadecimal y poderlos plasmar en un display.

Material Empleado

- AVR Studio
- 2 ProtoBoard
- Circuito integrado ATMega16
- Cables para conexiones
- 1 dip switch
- Resistencias de 100 ohm
- Leds
- Fuente de alimentación
- Display

Desarrollo

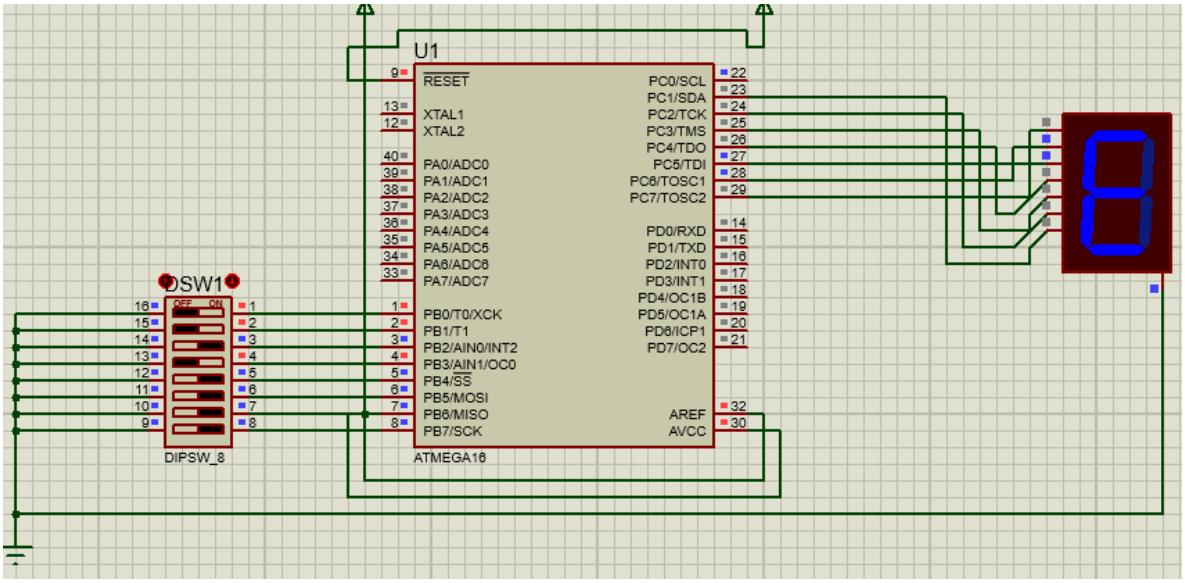
Para esta práctica se empleó el siguiente código:

```
1. .include "m8535def.inc"
2.          .def aux = R16
3.          .def dato = R17
4.
5.          ser aux
6.          out DDRC, aux
7.          out PORTB, aux
8.          ldi R20, $77
9.          ldi R21, $41
10.               ldi R22, $6e
11.               ldi R23, $6b
12.               ldi R24, $59
13.               ldi R25, $3b
14.               ldi R26, $3f
15.               ldi R27, $71
16.               ldi R28, $7f
17.               ldi R29, $7b
18.               clr zh
19.       otro:
20.               ldi zl, 20
21.               in dato, PINB
22.               cpi dato, 10
23.               brsh ponE
```

```
24.             add zl, dato
25.             ld dato, z
26.     ponlo:
27.             out PORTC, dato
28.             rjmp otro
29.     ponE:
30.             ldi dato, $3d
31.             rjmp ponlo
```

Captura de Evidencia

Aquí está la imagen resultado de la simulación que se hizo de este código tras tener algunas dificultades con el circuito.



Conclusiones

Chávez Chávez Ángel Alexis

En el desarrollo de esta práctica observamos el manejo de entradas y salidas al atmega16, así como el almacenamiento de los valores de codificados de los números hexadecimales.

Hernández López Pamela

Esta práctica fue bastante buena ya que en cuanto resultados fuimos eficientes para la realización por lo que el resultado final fue exitoso. Nos quedó bien desde el circuito hasta la codificación del microcontrolador y la lógica del programa fue la correcta, por lo que podemos con la práctica fue satisfactoria.

Flores Alvarado Diego Armando

Aquí comienza el cambio. Usamos un código un poco más largo, además de la implementación de un display en nuestro circuito y también se implementó el uso de números hexadecimales para poder hacer nuestro decodificador.

Reporte de Práctica 4

Introducción

En esta práctica el alumno debe hacer un circuito con un microcontrolador programado que consiste en un contador, hará que los números vayan cambiando a una frecuencia diferente administrada por un dip switch.

Objetivo

Retroalimentar al alumno en el tópico del timer, aprender a programarlo y ejecutarlo en el código correspondiente.

Material Empleado

- AVR Studio
- 2 ProtoBoard
- Circuito integrado ATMega16
- Cables para conexiones
- 1 dip switch
- Resistencias de 100 ohm
- Leds
- Fuente de alimentación
- Display

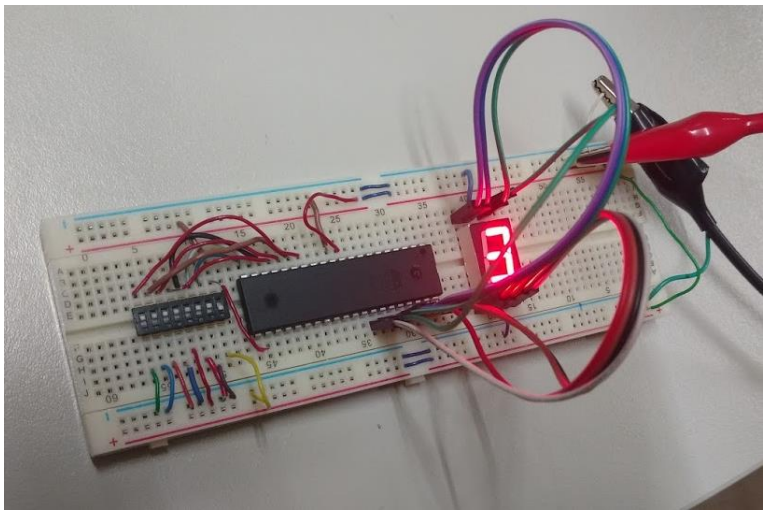
Desarrollo

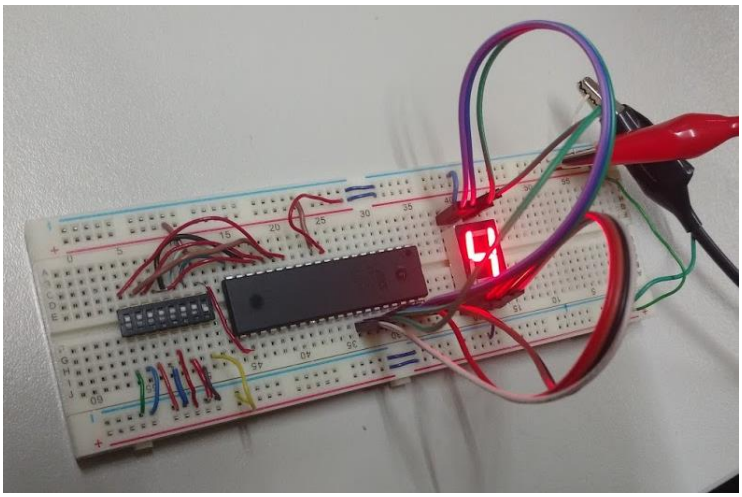
Para esta práctica se empleó el siguiente código:

```
1.          .include "m8535def.inc"
2.          .def deco = R17
3.          .def cuenta = R16
4.          .def aux = R18
5.
6. ; ---- Iniciar apuntador de pila ---
7.          ldi aux, low(RAMEND)
8.          out spl, aux
9.          ldi aux, high(RAMEND)
10.               out sph, aux
11.      ; ---------------------------------
12.               ser aux
13.               out DDRC, aux
14.               out PORTB, aux
15.               ldi R20, $77
16.               ldi R21, $41
17.               ldi R22, $6e
18.               ldi R23, $6b
19.               ldi R24, $59
20.               ldi R25, $3b
21.               ldi R26, $3f
```

```
22.             ldi R27, $71
23.             ldi R28, $7f
24.             ldi R29, $7b
25.
26.     uno:
27.             clr cuenta
28.     dos:
29.             rcall decodifica
30.             out PORTC, deco
31.
32.             in aux, PINB
33.             andi aux, 0b00000011
34.             inc aux
35.     otro:
36.             rcall delay
37.             dec aux
38.             brne otro
39.
40.             inc cuenta
41.             cpi cuenta, 10
42.             brne dos
43.             rjmp uno
44.     decodifica:
45.             clr zh
46.             ldi zl, 20
47.             add zl, cuenta
48.             ld deco, z
49.             ret
50.     delay:
51.             push aux
52.             ldi R17, $A7
53.     WGLOOP0:
54.             ldi aux, $02
55.     WGLOOP1:
56.             ldi R19, $F8
57.     WGLOOP2:
58.             dec R19
59.             brne WGLOOP2
60.             dec R18
61.             brne WGLOOP1
62.             dec R17
63.             brne WGLOOP0
64.             nop
65.             pop aux
66.             ret
```

Captura de Evidencias





Al momento de probar el código en el circuito, se puede observar el cambio entre números a una frecuencia de tiempo, ésta puede ser cambiada en el dip switch.

Conclusiones

Chávez Chávez Ángel Alexis

En esta práctica tuvimos muchas complicaciones ya que nuestro manejo de librerías no era el adecuado y nos creímos tener problemas de pines quemados en nuestro ATMega16, pero al final logramos desarrollar el contador propuesto.

Hernández López Pamela

Para la práctica esta práctica invertimos mucho tiempo ya que hicimos todo como debía hacerse y aún así hubo errores, por más que hicimos cambios no encontrábamos el elemento que fallaba en el circuito o en el programa pero al final al cambiar todos los displays nos dimos cuenta que ese era el problema real ya que solo funcionaba con un display y sin importar en qué lugar lo colocáramos solo funcionaba ese display y así encontramos la forma de solucionarlo.

Flores Alvarado Diego Armando

Una de las prácticas más estresantes ¿Por qué? Porque no sabíamos qué era lo que fallaba. Al final nos dimos cuenta de que había algunos problemas con nuestro microcontrolador y con algunas librerías que utilizamos. Finalmente solucionamos los problemas y la práctica volvió a tomar un rumbo correcto y nuestro contador funcionó.