



Laboratorio de Microprocesadores - 86.07

Trabajo Práctico N°3: Auto fantástico / Rotación de LEDs

Profesor:			Ing. Guillermo Campiglio									
Cuatrimestre/Año:			1º/2020									
Turno de las clases prácticas			Miércoles									
Jefe de trabajos prácticos:			Ing. Pedro Ignacio Martos									
Docentes guía:			Ing. Fabricio Baglivo, Ing. Fernando Pucci									
Fecha de presentación:			03/06/2020									
Alumno			Seguimiento del proyecto									
Nombre	Apellido	Padrón										
Maximiliano Daniel	Reigada	100565										

Observaciones:

.....

.....

.....

.....

.....

.....

.....

.....

Fecha de aprobación			Firma J.T.P	

Índice

1. Objetivo	1
2. Descripción	1
3. Diagrama de conexiones en bloques	1
4. Circuito esquemático	2
5. Listado de componentes	2
6. Diagrama de flujo del Software	3
7. Código de programa	4
8. Resultados	5
9. Conclusiones	6

1. Objetivo

El presente trabajo práctico tiene como objetivo progresar con el manejo de puertos, poniendo en práctica el control en forma independiente de cada pin, a fin de completar una secuencia de encendido y apagado. Además de esto, se busca analizar la corriente entregada por el microcontrolador durante la práctica, y si ésta cumple con las especificaciones detalladas en la hoja de datos del dispositivo.

2. Descripción

Mediante la implementación de una placa de microcontrolador de código abierto basado en el microchip ATmega328P, denominada Arduino Uno, se busca controlar el desplazamiento del encendido de un LED en una serie de seis. La rotación que se busca es de izquierda a derecha y viceversa, manteniendo solo un LED encendido a la vez, siguiendo la secuencia 000001, 000010, 000100, , 100000, 010000, 001000, etc.

A su vez, se calcula el consumo, corriente por pin y corriente suministrada por el microcontrolador a fin de obtener conclusiones relacionadas al comportamiento en DC del dispositivo.

3. Diagrama de conexiones en bloques

A continuación, puede verse el diagrama de conexiones en bloques correspondiente a este trabajo práctico:

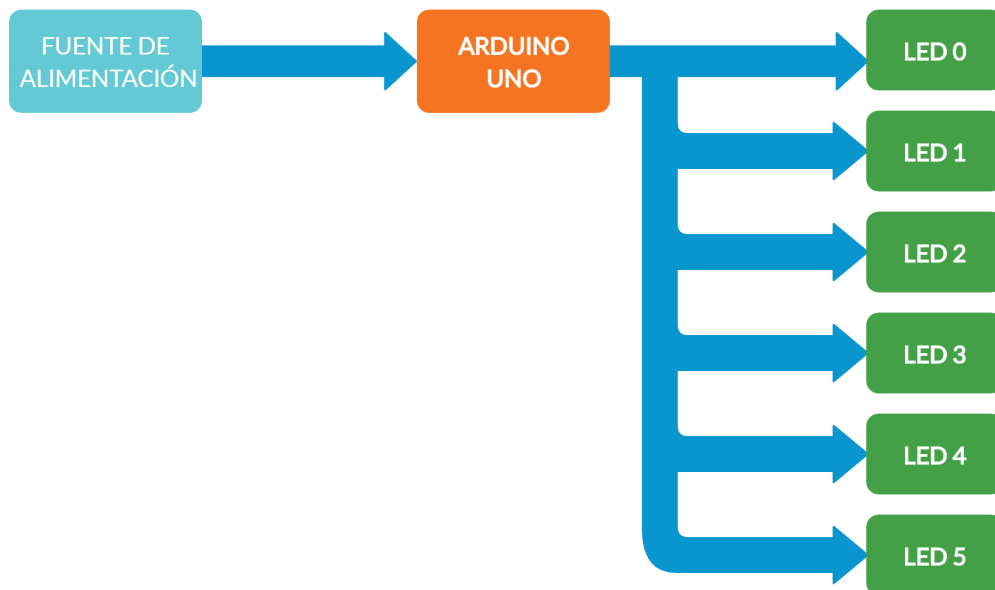


Figura 1: Diagrama de conexiones en bloques.

6. Diagrama de flujo del Software

A continuación, se explicitan los diagramas de flujo correspondientes al código desarrollado:

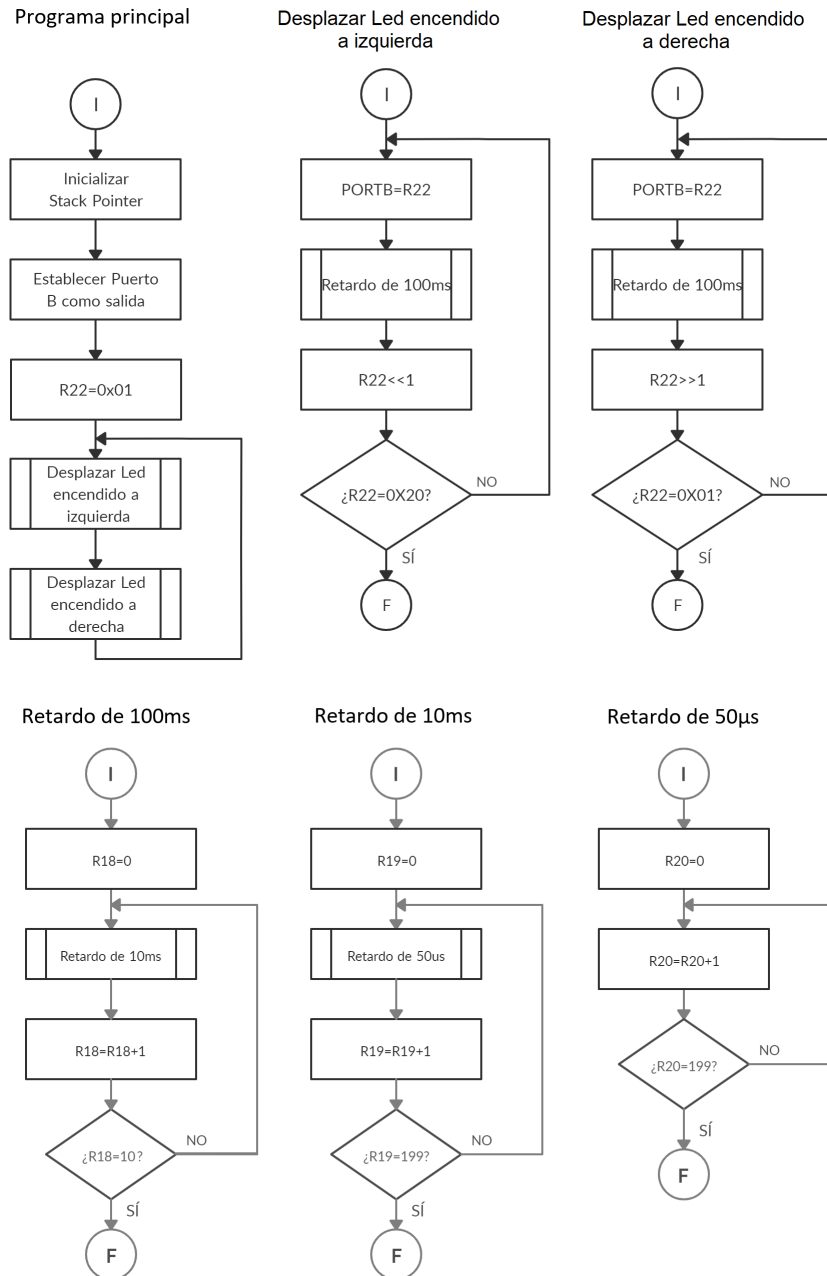


Figura 3: Diagramas de flujo.

7. Código de programa

A continuación, puede verse el código desarrollado para este proyecto:

```

1  ;*****
2  ; Rotación en encendido de leds conectados al PORTB
3  ;
4  ; Alumno: Reigada Maximiliano Daniel
5  ; Padrón: 100565
6  ;*****
7
8  .include "m328pdef.inc"
9
10 .cseg
11 .org 0x0000
12     rjmp main
13
14 .org INT_VECTORS_SIZE
15
16 main:
17
18     ldi R16, HIGH(RAMEND)    ;Se inicializa el Stack Pointer al final de la RAM
19     out SPH, R16            ;Carga el SPH
20     ldi R16, LOW(RAMEND)    ;Carga el SPL
21     out SPL, R16
22
23     ldi R16, 0xFF            ;Se configura el puerto B
24     out DDRB, R16           ;Configura al puerto B como salida
25     ldi R22, 0x01           ;Se rota el led encendido a izquierda
26
27 desplaz_led_izquierda:
28     out PORTB, R22
29     rcall retardo_100ms
30     lsl R22
31     cpi R22, 0x20
32     brne desplaz_led_izquierda ;Se vuelve a desplazar hasta que R22=0x20
33
34                                     ;Se rota el led encendido a derecha
35 desplaz_led_derecha:
36     out PORTB, R22
37     rcall retardo_100ms
38     lsr R22
39     cpi R22, 0x01
40     brne desplaz_led_derecha
41
42     jmp desplaz_led_izquierda ;Se vuelve a desplazar hasta que R22=0x01
43
44
45
46 ;*****
47 ; Retardo de 100ms calculado con un cristal de 16MHz
48 ;El loop interno se debe ejecutar un numero X de veces para que el tiempo total sea 100ms.
49 ;Considerando los ciclos de máquina que conlleva cada instrucción ejecutada:
50 ;100ms=(1/16MHz)*(1+X*160004-1+4) ----> X=10
51 ;*****
52 retardo_100ms:
53     eor R18, R18            ;1 CM
54 loop_retardo_100ms:
55     rcall retardo_10ms      ;160000 CM
56     inc R18                 ;1 CM
57     cpi R18, 10             ;1 CM
58     brne loop_retardo_10ms ;2 CM
59                                     ;1 CM (brne conlleva 1CM cuando R18=100)
60     ret                     ;4 CM
61
62
63 ;*****
64 ; Retardo de 10ms calculado con un cristal de 16MHz
65 ;El loop interno se debe ejecutar un numero X de veces para que el tiempo total sea 10ms.
66 ;Considerando los ciclos de máquina que conlleva cada instrucción ejecutada:
67 ;10ms=(1/16MHz)*(1+X*804-1+4) ----> X=199
68 ;*****

```

```

69 retardo_10ms:
70     eor     R19, R19                ;1 CM
71 loop_retardo_10ms:
72     rcall   retardo_50us           ;800 CM
73     inc     R19                    ;1 CM
74     cpi     R19, 199                ;1 CM
75     brne    loop_retardo_10ms      ;2 CM
76                                     ;1 CM (brne conlleva 1CM cuando R19=199)
77     ret                                     ;4 CM
78
79
80
81 ;*****
82 ; Retardo de 50us calculado con un cristal de 16MHz
83 ;El loop interno se debe ejecutar un numero X de veces para que el tiempo total sea 50us.
84 ;Considerando los ciclos de máquina que conlleva cada instrucción ejecutada:
85 ;50us=(1/16MHz)*(1+X*4-1+4) ----> X=199
86 ;*****
87 retardo_50us:
88     eor     R20, R20                ;1 CM
89 loop_retardo_50us:
90     inc     R20                    ;1 CM
91     cpi     R20, 199                ;1 CM
92     brne    loop_retardo_50us      ;2 CM
93                                     ;1 CM (brne conlleva 1CM cuando R20=199)
94     ret                                     ;4 CM

```

8. Resultados

Luego de desarrollar el código descrito en las secciones anteriores y armar el circuito dispuesto, se logra controlar el encendido y apagado secuencial de una serie de LEDs conectados al puerto B del microcontrolador utilizado.

Además de esto, se procede a calcular la corriente por pin suministrada por el microcontrolador, y la potencia entregada por éste. En este caso particular, se utilizan LEDs de color verde, por lo que la caída de tensión que cada uno genera es de aproximadamente 2,4V, de modo que:

$$I_{pin} = \frac{V_{CC} - V_{LED}}{R_{ext}} = \frac{5V - 2,4V}{220\Omega} = 11,82mA \quad (1)$$

$$\Rightarrow P_{pin} = V_{CC} \cdot I_{pin} = 5V \cdot 11,82mA = 59,1mW \quad (2)$$

Considerando que solo se enciende de a un LED a la vez, y que todos presentan características constructivas casi idénticas, se deduce que la corriente y potencias calculadas anteriormente cumplen para cualquiera de los pines utilizados. Por otro lado, según la hoja de datos del dispositivo, la máxima corriente que el microcontrolador puede entregar por un pin (I/O) es de 40mA, de manera que se verifica que este caso se trabaja por debajo del caso límite permitido.

Finalmente, se procede a realizar un listado de los componentes utilizados y sus costos:

Componentes	Costos
Leds verde de 5mm	\$42,00
Resistores de 220 Ω	\$33,00
Cables de conexión	\$14,00
Protoboard	\$232,00
Arduino Uno	\$720,00
TOTAL	\$1041,00

9. Conclusiones

Tras haber realizado todos los pasos pedidos en el enunciado de este trabajo práctico, resta destacar las conclusiones que la experiencia ha aportado.

A partir de este trabajo se logra afianzar el manejo de pines de manera independiente y secuencial, junto con el control de registros mediante operaciones de desplazamiento y comparación. Por otra parte, también se da una idea de las verificaciones técnicas asociadas al consumo y corrientes, que se deben hacer al momento de comunicar periféricos con dispositivos de control.