



(6609) LABORATORIO DE MICROCOMPUTADORAS

Proyecto:
Trabajo Práctico 1: Parpadeo de un LED

Profesor:	Ing. Guillermo Campiglio
Cuatrimestre / Año:	1º Cuatrimestre de 2020
Turno de clases prácticas:	Miércoles
Profesores de Trabajos Prácticos:	Ing. Pedro Martos, Ing. Fabricio Baglivo, Ing. Fernando Pucci

Autor			Seguimiento del proyecto						
Nombre	Apellido	Padrón							
Jose Maria	Borgatello	67758							

Observaciones:

Fecha de aprobación		

Firma J.T.P.	

COLOQUIO	
Nota final	
Firma Profesor	

Índice:

Objetivo del Trabajo.....	Pag 3
Descripción del Trabajo.....	Pag 3
Diagrama de conexiones en bloques.....	Pag 4
Círculo esquemático.....	Pag 4
Listado de componentes.....	Pag 4
Diagrama de flujo del software.....	Pag 5
Código de programa.....	Pag 6
Resultados.....	Pag 10
Conclusiones.....	Pag 10

Objetivo del Trabajo:

Realizar dos programas que hagan parpadear un LED conectado al pin 14 de un microcontrolador ATmega328p.

Cada uno de estos programas utilizaran dos métodos distintos de programación:

- 1) Utilizando todo el puerto
- 2) Utilizando solo un bit del puerto (el que corresponde al pin donde se conecta el LED que parpadeará).

Descripción del Trabajo:

Se instaló los correspondientes softwares para la programación del Micro mencionado.

AVR Studio como IDE para la programación en assembler.

WinAVR para el quemado o traspaso del programa al microcontrolador.

Se utilizó la plataforma Arduino para acceder al micro ATmega328p.

Se debió consultar el pinout del micro ATmega328p y el conexionado interno de la plataforma arduino para estudiar la concordancia de pines internos y externo de la placa mencionada.

El pin 14 del micro es el PBO.

Esto llevo a la conclusión que para conectar el LED al pin 14 del micro ATmega328p se debía usar la I/O digital 8 de la plataforma Arduino.

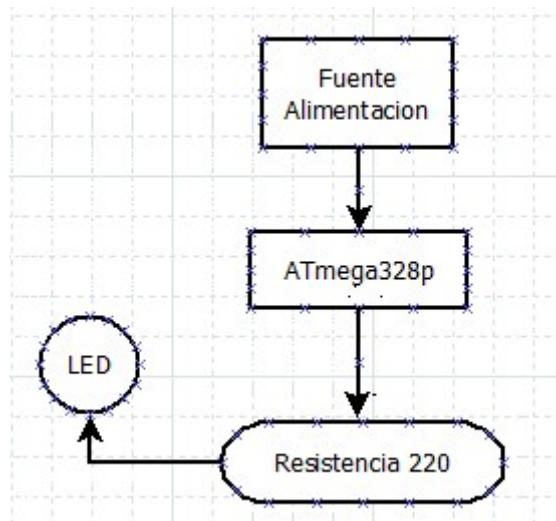
Para programar en assembler se usó lo aprendido en la materia.

Para que la programación utilice todo el puerto simplemente setea todo DDRB, como salida y cuando se desea utilizar un solo bit del puerto se setea solo el bit correspondiente de DDRB.

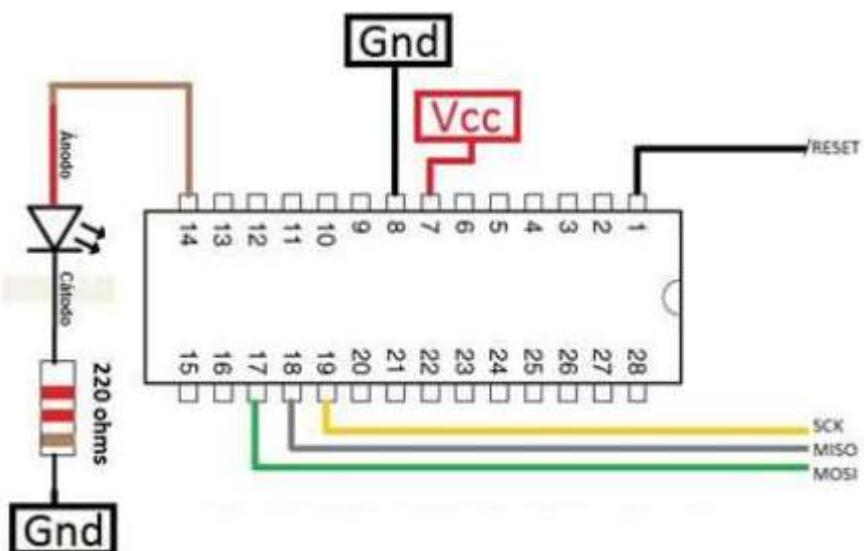
Ambas programaciones fueron realizadas con éxito, resultando los archivos .asm correspondientes.

Nótese que debido al uso de variables y constantes, la única variación entre cada código .asm es cambiar el valor de .equ PUERTOB por un valor u otro

Diagrama de conexiones en bloques:



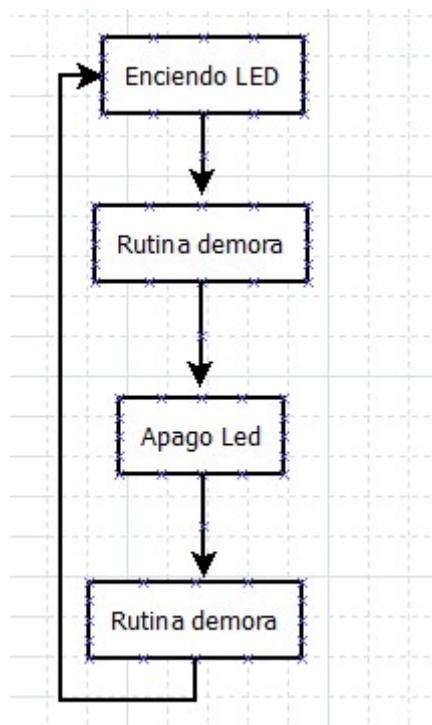
Circuito esquemático:



Listado de Componentes:

- 1 microcontrolador ATmega328p (se usó la plataforma Arduino UNO)
- 1 resistencia de 220 Ω
- 1 cables de conexión
- 1 Plaqueta universal (Para la presente versión de entrega se uso un ProtoBoard)

Diagrama de flujo del software:



Codigo de programa:

1) PORTB completo como salida:

```
; TP1.1.asm
;
; Created: 20/5/2020 15:09:58
; Author : Jose Maria Borgatello
;

.include "m328pdef.inc"

.equ PBX=0 ; Led en PB0
.equ PUERTOB=0b11111111 ;(PORTB completo como salida)

.cseg
.org 0x0000
        jmp      main
.org INT_VECTORS_SIZE

main:
; Configuro puerto B
        ldi      r20,PUERTOB
        out      DDRB,r20

xsiempre:
; rutina de encendido y apagado
prendo:      sbi      PORTB,PBX      ; encendido del led

demora1:
        ldi      r20,0x00
        ldi      r21,0x00
        ldi      r22,0x03
ciclo1:      inc      r20
        cpi      r20,0xff
        brlo   ciclo1
        ldi      r20,0x00
        inc      r21
        cpi      r21,0xff
        brlo   ciclo1
        ldi      r21,0x00
        inc      r22
        cpi      r22,0x20
        brlo   ciclo1

        cbi      PORTB,PBX      ; apagado del led

demora2:
        ldi      r20,0x00
        ldi      r21,0x00
        ldi      r22,0x01
ciclo2:      inc      r20
        cpi      r20,0xff
        brlo   ciclo2
```

```
ldi      r20,0x00
inc      r21
cpi      r21,0xff
brlo    ciclo2
ldi      r21,0x00
inc      r22
cpi      r22,0x20
brlo    ciclo2

rjmp  xsiempre
```

2) PORTB completo como salida:

```
; TP1.2.asm
;
; Created: 20/5/2020 15:10:58
; Author : Jose Maria Borgatello
;

.include "m328pdef.inc"

.equ PBX=0 ; Led en PB0
.equ PUERTOB=0b00000001 ;(PORTB con bit 0 solamente como salida)

.cseg
.org 0x0000
        jmp      main

.org INT_VECTORS_SIZE

main:
; Configuro puerto B
        ldi      r20,PUERTOB
        out      DDRB,r20

xsiempre:
; rutina de encendido y apagado

prendo:          sbi      PORTB,PBX      ; encendido del led

demora1:
        ldi      r20,0x00
        ldi      r21,0x00
        ldi      r22,0x03
ciclo1:          inc      r20
        cpi      r20,0xff
        brlo   ciclo1
        ldi      r20,0x00
        inc      r21
        cpi      r21,0xff
        brlo   ciclo1
        ldi      r21,0x00
        inc      r22
        cpi      r22,0x20
        brlo   ciclo1

        cbi      PORTB,PBX      ; apagado del led

demora2:
        ldi      r20,0x00
        ldi      r21,0x00
        ldi      r22,0x01
ciclo2:          inc      r20
        cpi      r20,0xff
        brlo   ciclo2
        ldi      r20,0x00
        inc      r21
        cpi      r21,0xff
```

```
brlo  ciclo2
ldi   r21,0x00
inc   r22
cpi   r22,0x20
brlo  ciclo2

rjmp xsiempre
```

Resultados:

El resultado obtenido en el presente TP fue exitoso.

Se logró cumplir las consignas correctamente.

Costo de los componentes (a la fecha de entrega del presente TP 20/05/2020):

1 Arduino UNO = \$1288

1 Resistencia 220 Ω = \$5,5

1 Led Verde = \$10.60

2 Cables de conexión = \$9,47

Protoboard = \$195

Conclusiones:

El presente trabajo tuvo dificultades referidas al aprendizaje en el uso del assembler y el uso de los softwares específicos.

Resultó ser un trabajo práctico muy motivador, para encarar los siguientes