

Práctica 3

i. Programación en Lenguaje Ensamblador del ATmega 328/2560

Objetivo: Mediante esta práctica el alumno analizará la implementación de retardos por software, así como también se familiarizará con la configuración y uso de puertos.

Equipo: - Computadora Personal con AVR Studio, AVRDUDE y tarjeta TJuino

Teoría: 1) Análisis y cálculos de retardos de la práctica -- subir día del Lab.

2) Teoría sobre puertos de E/S (uC ATmega328/2560) -- en reporte

Descripción:

1. Crear en AVR Studio el proyecto correspondiente para la práctica no. 3 el cual deberá de incluir el archivo **Prac3.asm** mostrado en el **Listado 1**.
2. Analice y determine el tiempo de retardo del macro llamdo **miRetardo**.
3. Ensamble el proyecto y cargue el archivo resultante (**.hex**) a la tarjeta TJuino.
4. Una vez cargado el archivo, la tarjeta TJuino lo ejecutará automáticamente y deberá encender y apagar el indicar tipo LED marcado con la letra **L**.

Listado 1:

```
.INCLUDE "m2560def.inc" ; Include Register/Bit Definitions for the 2560;
***** M A C R O S *****
;*****
; miRetardo
; Descripción: Retardo de X mS basado en ciclos anidados
; Registros usados: R24, R25 y R26
; valores de Retorno: N/A
; Nota: Esto registros son modificados y se pierde el valor original
;*****
.MACRO      miRetardo
    clr     r24
nxt0:      clr     r25
nxt1:      clr     r26
nxt2:      dec     r26
          brne    nxt2
          dec     r25
          brne    nxt1
          dec     r24
          brne    nxt0
.ENDMACRO

;***** Inicialización *****
    sbi     DDRB,PB7      ; configurar PB7 como salida

;***** Ciclo Principal *****
next:      sbi     PORTB,7      ; Escribir 1 en PB7
          miRetardo      ; Esperar X mS
          cbi     PORTB,7      ; Escribir 0 en PB7
          miRetardo      ; Esperar X mS
          rjmp    next
;***** FIN archivo *****
```

Modificaciones a Realizar

1. Realice los cambios necesarios para convertir el macro **miRetardo** a un procedimiento que deberá ser llamado en el ciclo principal en lugar de invocar al macro. Verifique el funcionamiento del procedimiento ensamblando y cargando el archivo correspondiente a la tarjeta TJuino.
2. Implementar la subrutina **PresentarBits** la cual presenta los bits del registro R24 en la termin PE1 (bit 1 del puerto E) de forma secuencial (uno a la vez).

```

;***** Algoritmo PresentarBits *****
Consideraciones:
El bit PE1 ya está programado como salida y se le ha escrito un 1 lógico

inicio:
Paso 1: Poner en bajo (0 lógico) a PE1.
Paso 2: Esperar un retardo de 103us.
Paso 3: Pasar el bit LSB (menos significativo) de R24 a PE1.
Paso 4: Esperar un retardo de 103us.
Paso 5: Hacer corrimiento a la derecha a R24.
Paso 6: Si no es el último bit ir al paso 3 de lo contrario continuar
        Nota: El último bit es cuando el MSB original ha llegado al LSB
Paso 7: Poner PE1 alto (1 lógico). Nota: se deja como estaba al inicio
Paso 8: Esperar un retardo de 103us.
fin:

```

Una vez implementado el procedimiento verificar su funcionalidad mediante la siguiente secuencia incluida el ciclo principal del programa.

```

; **** incluir en la inicialización
sbi DDRE,PE1 ; Programa PE1 como salida
sbi PORTE,PE1 ; Escribir 1 en PE1
clr r16 ;
sts UCSR0B,r16 ;
;***** Ciclo Principal *****
next: sbi PORTB,7 ; Escribir 1 en PB7
      rcall miRetardo ; Esperar X mS
      cbi PORTB,7 ; Escribir 0 en PB7
      rcall miRetardo ; Esperar X mS
;***** prueba de PresentarBits *****
      ldi r24,0x55
      rcall PresentarBits
      ldi r24,0x41
      rcall PresentarBits
      ldi r24,0x42
      rcall PresentarBits
      ldi r24,0x43
      rcall PresentarBits
      rjmp next

***** ETN archivo *****

```

Comentarios y Conclusiones

Bibliografía y Referencias

-0-