

Examen 2do Departamental

Pozos Reyes Luis Angel 3CV17

Introducción a los Microprocesadores y Microcontroladores

1.- Con el sobreflujo del TIMER0 aproximadamente cada 1 mseg. haga un toggle en la señal de salida por el pin PB0. (2.5 pts)

```
//Programa 1
    .def  xd=r17
    .def  xd2=r18

    .cseg
    .org  0
    rjmp  main

    .org  $020
    rjmp  timer0

reset:
    ldi  xd, $01
    out  ddrb, xd

    ldi  xd, $00
    out  portb, xd
//preescalador a 8
    ldi  xd, $02
    out  tccr0b, xd

    ldi  xd, $01
    sts  tmsk0, xd

    sei

main:
    rjmp  main

timer0:
    in  xd, pinb
    com  xd
    andi  xd, $01
    out  portb, xd
    reti
```

2.- Se tiene conectado un push-boton con resistencia de pull-up interna en el pin PB1 de manera que cada que se presiona el botón se toma una muestra de una señal analógica en ADC0, la resolución del ADC es de 8 bits y el resultado de la conversión se muestra en 8 leds conectados en el puerto D. (2.5 pts)

3.- Utilizando el TIMER 0 generar una señal PWM con una frecuencia de 244 Hz aproximadamente y un ciclo de trabajo del 25% aprox. (2.5 pts)

//Programa 3

```
.def xd=r16
.def cont=r17

.cseg
.org 0

ldi xd, $40
out ddrd, xd

ldi xd, $83
out tccr0a, xd
//preescalador a 64 para frecuencia 244
ldi xd, $03
out tccr0b, xd
//bit 63 es 25%
ldi xd, $3f
out ocr0a, xd
```

4.- Modifique el código del problema anterior de manera que por medio de la USART se pueda modificar el ciclo de trabajo de la señal PWM, es decir los datos que se reciben en la USART modifican el ciclo de trabajo de la señal PWM. (2.5 pts)

//Programa 4

```
.def xd=r16
.def cont=r17

.cseg
.org 0
rjmp reset

.org $024
rjmp recibe

reset:

ldi xd, $fe
out ddrd, xd

ldi xd, $98
sts ucsr0b, xd

ldi xd, 103
sts ubrr01, xd

ldi xd, $83
out tccr0a, xd
//preescalador a 64 para frecuencia 244
ldi xd, $03
out tccr0b, xd
```

```

main:
    rjmp main

recibe:
    lds xd, udr0

    out ocr0a, xd
    reti

```

**5.- Obtenga una señal cuadrada en el pin PB0 de un periodo aproximado de 32 mseg, utilice un delay.
(2.5 pts)**

//Programa 5

```

    .def xd=r16

    .def cont1=r17
    .def cont2=r18
    .def cont3=r19

    .cseg
    .org 0

    ldi xd, $01
    out ddrb, xd

main:
    ldi xd, $01
    out portb, xd
    call delay_32m
    ldi xd, $00
    out portb, xd
    rjmp main

delay_32m:
    ldi cont1, 1
lazo3: ldi cont2, 250
lazo2: ldi cont3, 205
lazo1: nop
        nop
        nop
        nop
        nop
        nop
        dec cont3
        brne lazo1
        dec cont2
        brne lazo2
        dec cont1
        brne lazo3
    ret

```

RESOLVER ÚNICAMENTE 4 PROBLEMAS, CONSIDERAR UNA FRECUENCIA DE TRABAJO DE 16 MHz.