



**INSTITUTO POLITÉCNICO  
NACIONAL**  
ESCUELA SUPERIOR DE CÓMPUTO



**“Manejo de Interrupciones Externas”**

**Integrantes del Equipo:**

Contreras Cardoso Adolfo

Martínez Alvarado Bryan Alexis

Maya Martínez Alonso Rubén

Pérez Gómez Santiago

**Grupo:**

3CM15

**Profesor:**

Ing. José Juan Pérez Pérez

**Asignatura:**

# Introducción a los Microcontroladores

## Planteamiento del Problema

Escribe un programa utilizando las interrupciones externas INT0 e INT1, que realice lo siguiente:

Se tendrán conectados interruptores momentáneos (pushbutton) en INT0 e INT1 y 8 leds conectados en el puerto A, al inicio por cada vez que se pulsa el botón conectado a INT0 se deberá conmutar el estado del PA0, y por cada vez que se pulsa el botón conectado en INT1 se deberá cambiar al siguiente led en el puerto A a conmutar.

## Desarrollo

Para realizar la implementación se utilizó como base el programa propuesto por el profesor, con el cual se consiguió encender y apagar un bit de la barra led con el uso de interrupciones externas. Como añadido se agregó un segmento de código que reinicia el bit elegido, pues al presionar nueve veces la interrupción que cambia de bit había un error por el cual no modificaba ningún led hasta que se presionaba una décima vez.

El diagrama de flujo correspondiente al resultado es el siguiente:

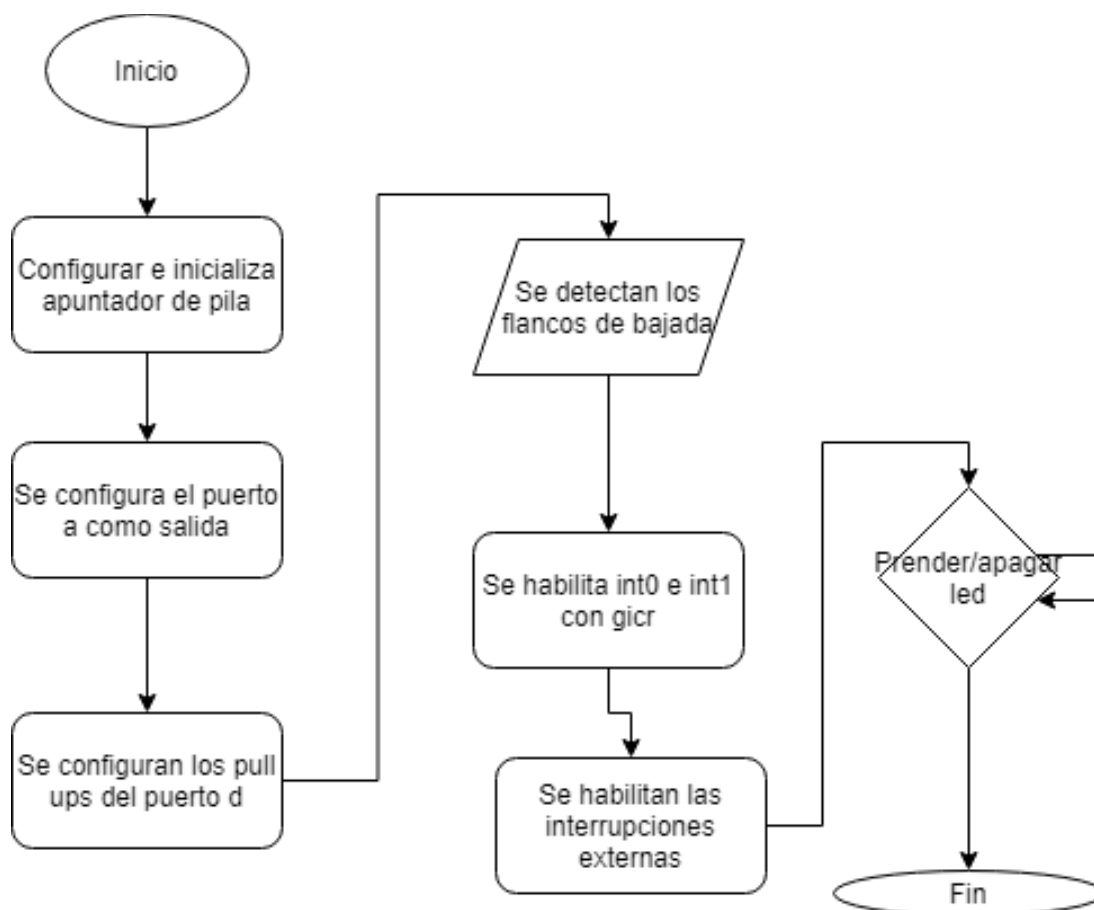


Figura 1. Diagrama de Flujo

El respectivo código en ensamblador es el siguiente:

```
    ;Código para prender y apagar
    ;led con interrupciones externas

    .include "m8535def.inc"
    .def aux = r16
    .def bit = r17

    ;Tiene la dirección 0
    rjmp main

    rjmp prenderApagar

    rjmp cambiarLed

main:
    ;inicializando apuntador de pila
    ldi aux,low(ramend)
    out spl,aux
    ldi aux,high(ramend)
    out sph,aux

    ;configurando puerto a como salida
    ser aux
    out ddra,aux

    ;configurando pull-ups del puerto d
    out portd,aux
```

**Figura 2.** Primera Sección de Código

```

;detectando los flancos de bajada
;para int0 e int1 en el registro mcucr
ldi aux,0b00001010
out mcucr,aux

;habilitando int0 e int1 con gicr
;(se puede habilitar int2 con otro uno)
ldi aux,0b11000000
out gicr,aux

;habilita las interrupciones externas
;cambiando la bandera i del registro estado
sei

;para apagar led
clr aux
ldi bit,1

;ciclando el cargado del puerto a (barra led)
;despues de una interrupción vuelve a esta
;sección de código siempre
loop:
    out porta,aux
    rjmp loop

```

**Figura 3.** Segunda Sección de Código

```

prenderApagar:
    ;leyendo el puerto a
    in aux,pina

    ;aplicando xor
    eor aux,bit
    reti

cambiarLed:
    ;rotando bit a la izquierda
    rol bit

    ;comparando si es cero
    cpi bit,0
    brne saltarReinicio

    ;cargando valor inicial
    ldi bit,1

    ;limpiando carry
    clc

saltarReinicio:
    reti

```

**Figura 4.** Tercera Sección de Código

Para la simulación, se cambiaron el primer, medio y último bit, el resultado es el siguiente:

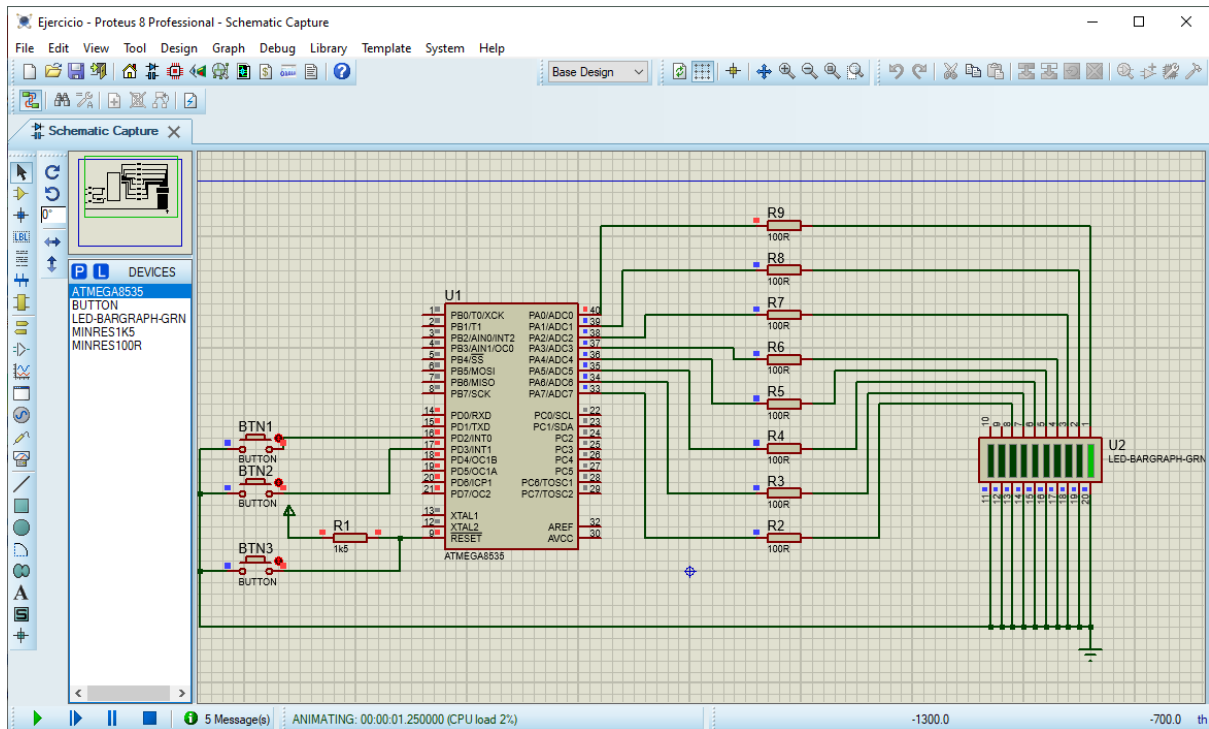


Figura 5. Primer Bit Encendido

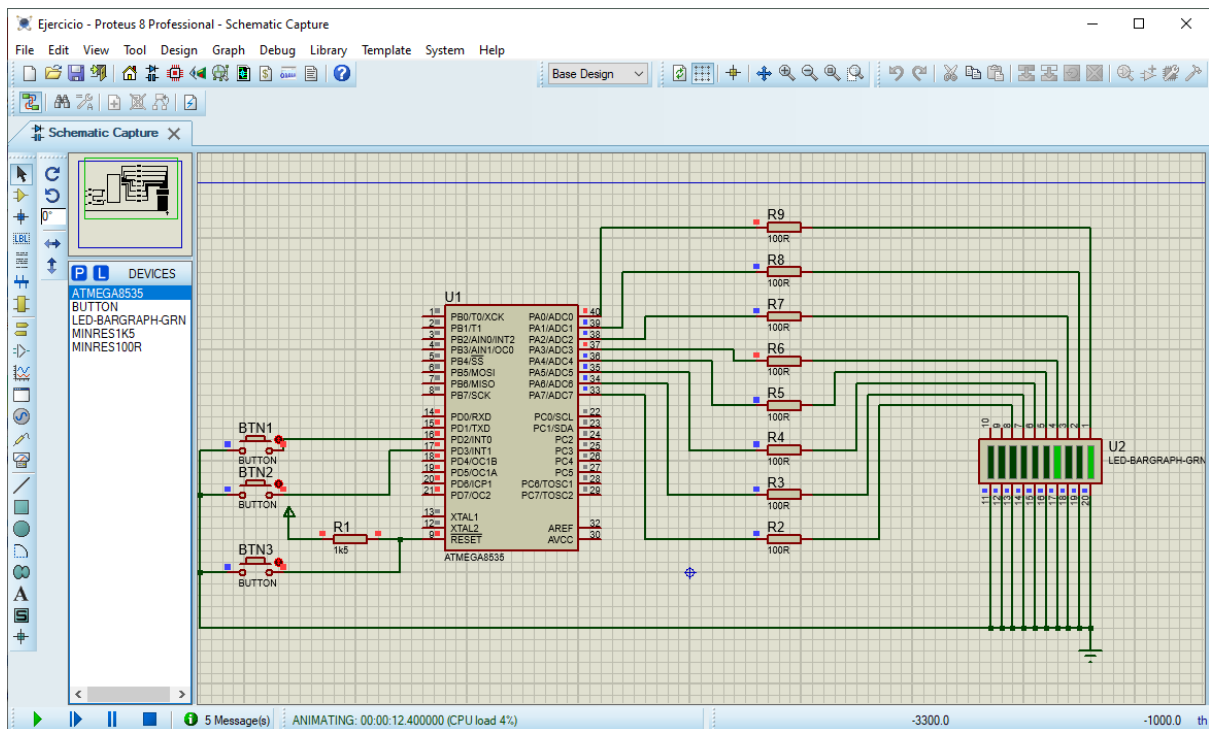


Figura 6. Uno de los Bits Intermedios Encendido

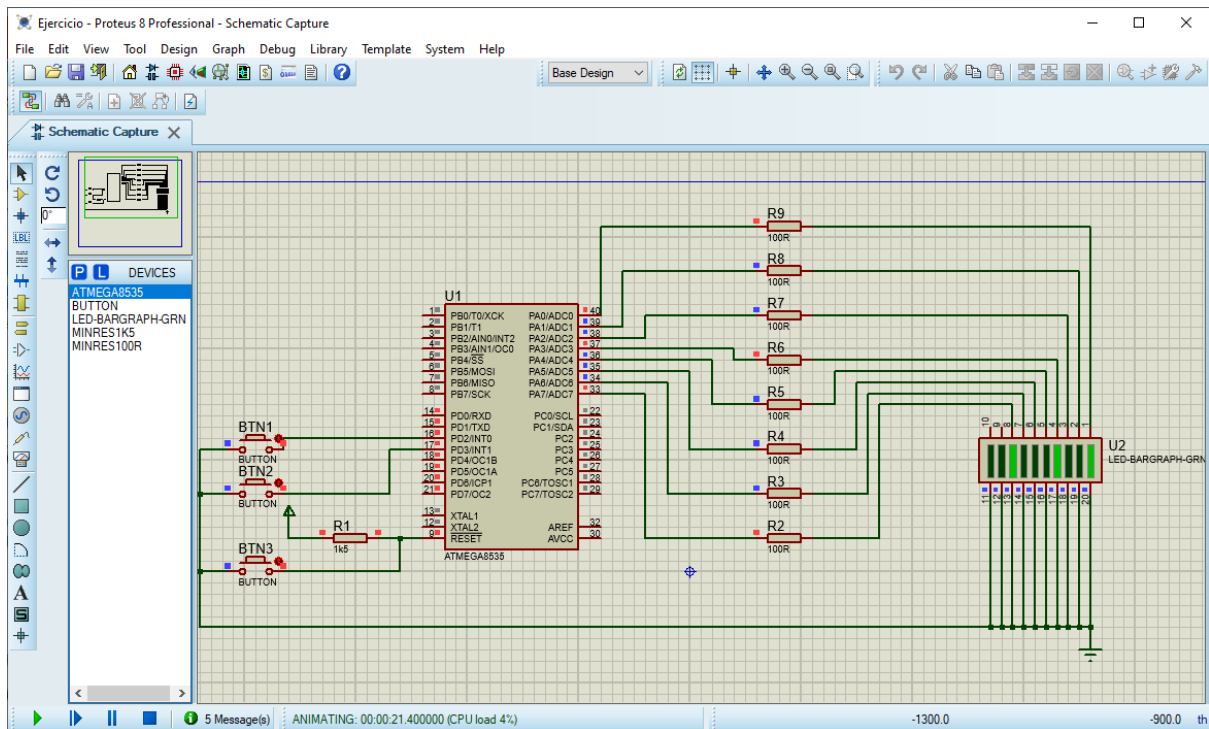


Figura 7. Último Bit Encendido

## Conclusiones Individuales

### Contreras Cardoso Adolfo

Tras la realización de esta tarea y alcanzar los objetivos de la misma, se comprendió como utilizar y manejar las interrupciones externas, sumando así mas instrucciones que aprendemos sobre el microcontrolador, personalmente me costó entender bien el uso y desarrollo de la tarea, pero gracias al trabajo en equipo pude comprender un poco mejor el tema de la tarea.

### Martínez Alvarado Bryan Alexis

Durante la realización de esta práctica hemos aplicado lo adquirido durante la clase de microcontroladores en el tema de retardo de los mismos, logramos observar y analizar la implementación de los mismos y hemos reafirmado como es que funcionan ciertos comandos empleados y el uso del simulador de circuitos. En este caso logramos encender y apagar leds de acuerdo a lo que se requería y necesitaba.

### Maya Martínez Alonso Rubén

El uso de interruptores ayudan a entender cómo funcionan cosas como teclados en dónde se necesita esperar a una interacción del usuario para que el programa o microcontrolador haga una acción. En este caso particular fue un poco complicado encontrar la solución al problema usando las

instrucciones que nos da el microcontrolador, el uso de la compuerta xor para hacer el switch de un solo bit en el registro de salida fue una solución bastante ingeniosa y muy difícil de encontrar. Discutimos muchas soluciones en equipo y la dinámica hizo que la solución fuera más fácil de encontrar y comprender.

**Pérez Gómez Santiago**

**Con la realización del trabajo planteado, se alcanzaron los objetivos propuestos, analizar y comprender el manejo de interrupciones externas en el microcontrolador ATmega8535, personalmente, resultó útil e interesante conocer otra posibilidad que brinda el microcontrolador para poder operar con las diversas instrucciones que posee.**