

# **FACULTAD DE ESTUDIOS SUPERIORES ARAGON**

## **LABORATORIO MICROPROCESADORES Y MICROCONTROLADORES**

### **PRACTICA VII**

Presentado a:

**VILLAR PEREZ ILEBARDO JOEL**

Presentado por:

**ROMERO TRUJILLO JUAN ANDRES**

Fecha de entrega:

**06/10/2025**

## Código de la práctica VII en Microchip Studio

The screenshot shows the Microchip Studio interface with the following details:

- Title Bar:** Practica 7 (Running) - Microchip Studio
- Menu Bar:** File, Edit, View, VAssistX, ASF, Project, Build, Debug, Tools, Window, Help
- Toolbar:** Includes icons for Open, Save, Print, Undo, Redo, Cut, Copy, Paste, Find, Replace, etc.
- Code Editor:** Main.asm file titled "Practica 7". The code is as follows:

```
; Practica 7.asm
; Created: 30/10/2025 09:45:58 a. m.
; Author : Andres

; Replace with your application code
.org 0x00
RJMP INICIO
.org 0x001A
RJMP RS1_T1

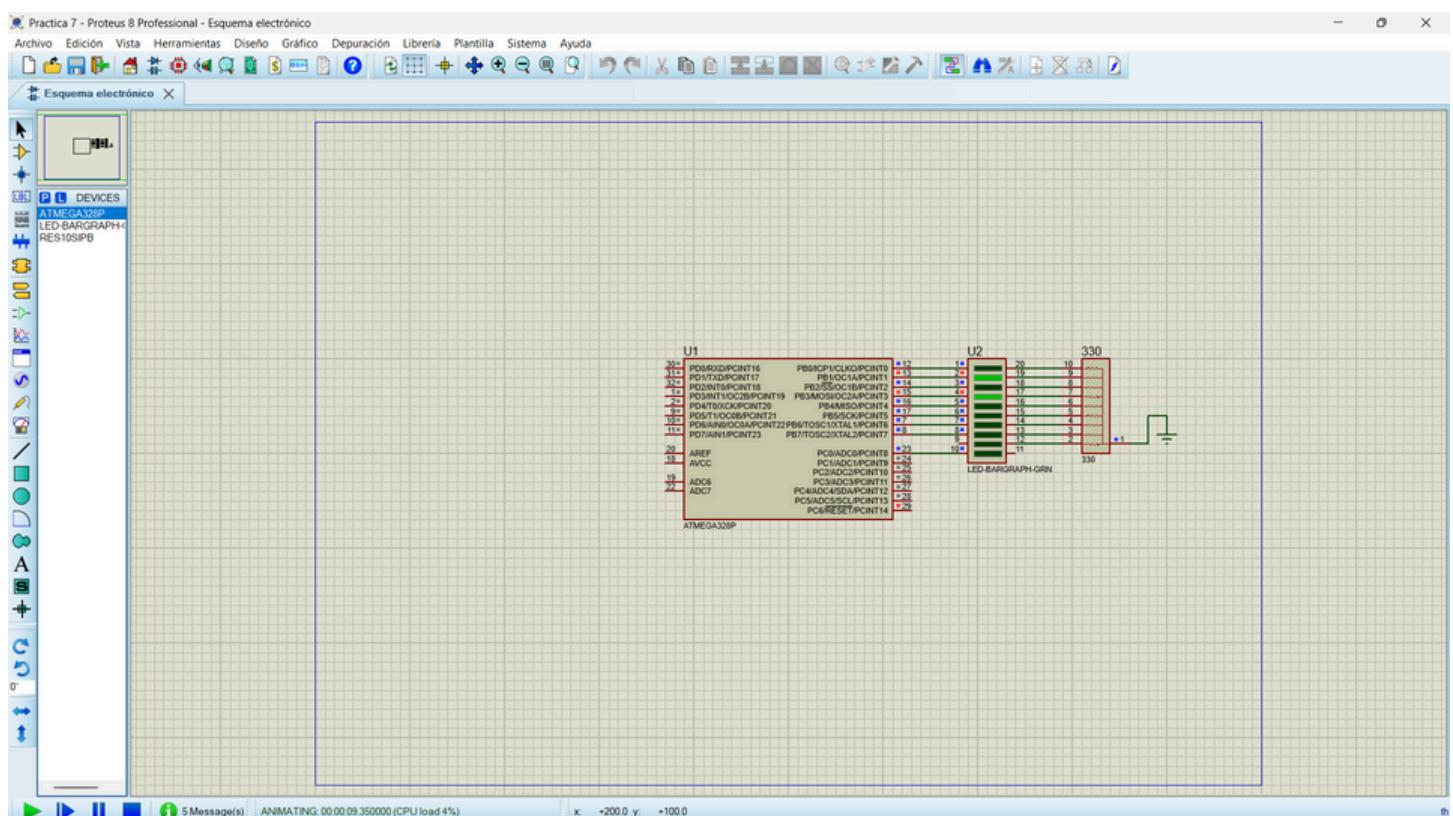
INICIO:
LDI R16,0x08
LDI R17,0xF
OUT SPL,R17
OUT SPH,R16
LDI R17,0xF
OUT DOR8,R17
LDI R16,0x01
STS TDSK1,R16
LDI R20,0x01
OUT TIF1,R20
LDI R17,0x03
STS TCCR1B,R17
LDI R18,0xF6
LDI R19,0xC3
STS TCH1H,R19
STS TCH1L,R18
OUT PORTB,R20
SEI

FIN:
    RJMP FIN

RS1_T1:
    STS TCH1H,R19
    STS TCH1L,R18
    INC R20
    OUT PORTB,R20
    RETI
```

- Memory View:** Shows "Memory 4" with the message "Unavailable when debugger is running."
- I/O Registers:** Shows the I/O register list with various pins and their functions.
- Status Bar:** Running

## Circuito en Proteus 8 con el código de Microchip Studio



# Lineas del código comentado

; Practica 7.asm  
; El programa configura el Timer1 en modo normal para generar interrupciones periódicas por overflow. En cada interrupción se incrementa el valor enviado al PORTB.

```
.org 0x00          ; Vector de reset ubicado al inicio del programa
RJMP INICIO        ; Al iniciar el micro, se salta a INICIO

.org 0x001A        ; Dirección del vector de overflow del Timer1
RJMP RSI_T1         ; Cuando ocurre overflow, se ejecuta la ISR RSI_T1

INICIO:
LDI R16,0x08        ; Valor alto para el puntero de pila (Stack Pointer)
LDI R17,0xFF        ; Valor bajo para el puntero de pila
OUT SPL,R17          ; Se establece SPL en 0xFF
OUT SPH,R16          ; Se establece SPH en 0x08 (Stack en 0x08FF)

LDI R17,0xFF        ; Preparación de máscara para configurar PORTB como salida
OUT DDRB,R17         ; PORTB se configura completamente como salidas

LDI R16,0x01        ; Bit TOIE1 activado para habilitar interrupción por overflow
STS TIMSK1,R16       ; TIMSK1 habilita la interrupción del Timer1

LDI R20,0x01        ; Se prepara un valor inicial para enviar al PORTB
OUT TIFR1,R20        ; Se limpia cualquier bandera previa de overflow del Timer1

LDI R17,0x03        ; Configuración del Timer1 con prescaler /64 y modo normal
STS TCCR1B,R17       ; El Timer1 comienza a contar con esas configuraciones

LDI R18,0xF6        ; Parte baja de la precarga inicial del Timer1 (TCNT1L)
LDI R19,0xC2        ; Parte alta de la precarga (TCNT1H)
STS TCNT1H,R19       ; Se carga primero la parte alta
STS TCNT1L,R18       ; Se carga luego la parte baja → valor inicial C2F6h

OUT PORTB,R20        ; Se envía el valor inicial (0x01) al PORTB
SEI                  ; Se habilitan las interrupciones globales

FIN:
RJMP FIN             ; Bucle infinito mientras las interrupciones se ejecutan

RSI_T1:
STS TCNT1H,R19       ; Se recarga la parte alta del timer para mantener el periodo
STS TCNT1L,R18       ; Se recarga la parte baja del timer
INC R20              ; El contador se incrementa en cada overflow
OUT PORTB,R20         ; Se actualiza PORTB con el nuevo valor del contador
RETI                 ; Se finaliza la rutina de interrupción
```