

TRABAJO PRÁCTICO FINAL

RTOS

Se desea implementar un sistema de control de acceso a un edificio de oficinas mediante el uso de una tarjeta magnética y un PIN (clave de acceso). El sistema contará con un encoder rotativo, display de siete segmentos, y un lector de tarjetas magnéticas. Además, visualizará en tiempo real la ocupación de los pisos mediante una matriz de LEDs. Toda la lógica deberá estructurarse sobre un RTOS.

1. Requerimientos Obligatorios

1.1 Base del Sistema

El control de acceso deberá estar basado en código propio desarrollado para el **TP1**.

1.2 Flujo

Al pasar una tarjeta y completar el acceso exitosamente, el sistema:

- Incrementa el contador del piso correspondiente
- Actualiza inmediatamente la visualización en la matriz LED
- Mantiene la fluidez de la interfaz de usuario sin delays.

1.3 Visualización de la Ocupación

Debe utilizarse una matriz LED 8x8 **WS2812B** controlada mediante **DMA**, con la siguiente organización:

- Distribución por pisos:
 - Columnas 1-2: Piso 1.
 - Columnas 3-4: Piso 2.
 - Columnas 5-6: Piso 3.
 - Columnas 7-8: Reservadas para indicadores del sistema.
- Representación de personas:
 - Cada persona se representa con 2 filas consecutivas iluminadas.
 - Máximo 4 personas por piso (8 filas disponibles).
 - LEDs encendidos de abajo hacia arriba conforme aumenta la ocupación.

1.4 Asignación de Tarjetas

Debe haber **al menos** 9 tarjetas pregrabadas:

- Tarjetas 1, 2 y 3: Usuarios del piso 1.
- Tarjetas 4, 5 y 6: Usuarios del piso 2.
- Tarjetas 7, 8 y 9: Usuarios del piso 3.

1.5 Implementación con RTOS (Micrium III)

1.5.1 Arquitectura de Threads

A diferencia del TP1 se deberá usar un **RTOS**, *Micrium III*, para separar el programa en **al menos** dos threads:

1. Thread principal: Se ocupará de la funcionalidad del TP1, adaptada a RTOS.
2. Thread de visualización: Se ocupará del control de la matriz LED y la visualización de ocupación por piso.

1.5.2 Eliminación de Polling

Se deberán eliminar loops de polling en **todo** el código, reemplazándolos por mecanismos de sincronización del sistema operativo. Para ello, considerar:

1. Drivers adaptados: Usar semáforos o eventos en lugar de flags.
2. Loops sincronizados: Usar OsSemPend() u OsPendMulti() en lugar de polls infinitos.
3. Comunicación entre threads: El thread principal deberá comunicarse con el de visualización usando Message Queue u otro mecanismo de sincronización.

1.5 Diseño de la Placa de Circuito Impreso

Se deberá diseñar una placa de circuito impreso utilizando el **módulo breakout** del microcontrolador provisto por la catedra. La placa deberá contar con:

1. Componentes necesarios para el funcionamiento del microcontrolador (cristal/oscilador, capacitores, etc).
2. Un encoder rotativo, 4 displays de 7 segmentos, 3 LEDs, decoders y shift registers (el mismo hardware utilizado en la placa del display del TP1).
3. Conectores para: programación, alimentación, tarjeta magnética, matriz LED.

2. Hardware Requerido

1. Módulo breakout K64F
2. Componentes de la placa Encoder-Display
3. Lector de banda magnética (TP1)
4. Matriz LED 8x8 WS2812B

Importante: Considerar los **niveles de tensión** (5V y/o 3.3V) y la dirección de las señales para la matriz WS2812B, además de la **corriente** requerida por el módulo al nivel o niveles de brillo implementados.

3. Requerimientos Deseables

1. **Colores/Animaciones:**
 - a. Diferenciar pisos por colores.
 - b. Diferenciar niveles de ocupación por colores.

- c. Agregar animaciones en los cambios de ocupación (transiciones, parpadeos, etc.).
- 2. **Feedback del sistema:** Mostrar mensajes visuales en columnas 7-8 ante errores o accesos correctos (PIN incorrecto, tarjeta inválida, etc.).
- 3. **Otras funcionalidades:**
 - a. Ajuste de brillo de la matriz mediante encoder (usuario administrador).
 - b. Reset de contadores (usuario administrador).
 - c. Alta/baja de usuarios.
 - d. Persistencia de usuarios y contadores frente a reinicios o cortes de energía.
 - e. Tolerancia a fallos: detección y reconexión automática de la matriz LED (*hotplug*).

4. Reglas Generales

- 1. Se debe utilizar Micriuum III como el RTOS del sistema.
- 2. El control de la matriz LED se debe implementar mediante DMA.
- 3. Este enunciado tiene validez por 1 año. Luego de ese período, se deberá realizar un nuevo TP.
- 4. Se permite el uso de SDK, Processor Expert y bibliotecas de terceros.
- 5. Es obligatoria la entrega de un documento que enumere todo el material externo utilizado.

5. Entrega y Evaluación

La entrega se realizará mediante una exposición oral de 30 minutos, en la cual el equipo deberá presentar tanto el funcionamiento del sistema como el diseño e implementación del firmware.

El trabajo se calificará considerando los siguientes aspectos:

- 1. Funcionamiento obligatorio: Cumplimiento de los requerimientos mínimos.
- 2. Diseño e implementación del firmware: Estructura, drivers utilizados, correcta arquitectura, uso correcto de los mecanismos de sincronización, adecuada protección de recursos compartidos entre Threads y/o Thread-ISR, medición del uso de CPU, cumplimiento de requerimientos temporales.
- 3. Diseño e implementación del PCB: Correcto funcionamiento, seguimiento de buenas prácticas de diseño, documentación.
- 4. Calidad de la presentación: Claridad, recursos visuales, tiempo y distribución de la exposición.
- 5. Características opcionales: Funcionalidades extra, robustez, usabilidad.

Además de la presentación, deberá entregarse la siguiente documentación:

- 1. Código fuente completo.

2. Documento técnico con la descripción del sistema, diagrama de arquitectura, mecanismos de sincronización utilizados y listado del material externo empleado.
3. Archivos de diseño del PCB junto con documentación asociada (esquemático, BOM).

Anexo

- [User Manual *Micrium III*](#)
- [Documentación WS2812B](#)