

Informe Curso UTN

Trabajo Integrador de Curso UTN

1^{ro} Joaquin Vigneta
jvignetta@gmail.com

2^{do} Santiago Zacarias
sanfran120771@gmail.com

Abstract—El informe del trabajo integrador del curso UTN, En este se va a encontrar un programa con FreeRTOS que usa una LPC845. Contiene varias tareas que se van a explicar y demostrar en un diagrama.

Index Terms—component, formatting, style, styling, insert

I. INTRODUCTION

Se va a usar FreeRTOS en un microcontrolador LPC845 para ejecutar varias tareas en paralelo que interactuan con varios de los perifericos. Se van a usar colas y semaforos para gestionar la prioridad de cada tarea sobre los perifericos.

II. DESCRIPCIÓN DE PROGRAMA

A. Descripción general

El programa va a usar sus tareas para con una medición de intensidad luz, ajustando un setpoint y dependiendo del valor cambiando el color de un led y alternar entre mostrar la medición de intensidad de luz y el setpoint. Tambien sonara un buzzer cuando se tape un sensor infrarrojo y se puede ajustar la intensidad de un led usando un potenciómetro.

B. Lista de tareas

Tarea	Descripción
task_init	Inicia todos los elementos a usar como colas, semaforos y perifericos y luego se elimina para ahorrar recursos.
task_adc	Realiza una conversion analogica a digital cada 250ms para luego ser leida.
task_display_change	Administra una cola con los datos a mostrar en el display.
task_display	Lee y muestra los datos desde la cola del display.
task_pwm	Ajusta la intensidad de un LED (0–100%) usando el potenciómetro RV22.
task_bh1750	Mide la intensidad de luz con el sensor BH1750 y la muestra en el display de 7 segmentos.
task_buzzer	El buzzer suena mientras el sensor infrarrojo esté cubierto.
task_counter_btns	Ajusta el setpoint (25–75) usando los botones S1 y S2.

Tarea	Descripción
task_user	Alterna entre mostrar luminosidad y setpoint con el botón USER. El punto decimal indica que se muestra el setpoint.
task_ledmulti	El LED se enciende azul si el setpoint es mayor que la iluminación, rojo si es menor.
task_terminal	Muestra por consola: tiempo en ms, intensidad de luz, setpoint y nivel del LED.

III. PROGRAMA

A. Diagrama de bloques del codigo



Fig. 1. Diagrama del sistema

IV. CONCLUSION

El trabajo realizado permitió adquirir experiencia en el uso de FreeRTOS para la gestión de tareas concurrentes en sistemas embebidos. Se aprendió a utilizar colas y semáforos para la comunicación y sincronización entre tareas, logrando una correcta interacción con periféricos como sensores, display, buzzer y LEDs.