

Software de Sistemas Embebidos

Informe Final

21 de junio de 2022

Profesor : Laprovitta , Agustín

Alumno : Nievas Aramayo, Pablo Agustín

1. Estructura física

El trabajo final de la materia se implementa en la placa STM32L100C-DISCO. Se programa en el procesador STM32L100RCT6 utilizando STM32CubeIDE 1.8.0 como herramienta de desarrollo.

Los periféricos a utilizar en el proyecto son:

- Pantalla LCD 16x2 con adaptador i2c
- Sensor de distancia por ultrasonido HCSR04
- Pulsador de control
- Optocoplador MOC3021
- Optocoplador 4N25
- Triac BT136

El LCD 16x2 se conecta con 4 conectores a la placa. El sensor de ultrasonido HCSR04 también requiere de 4 conectores. Ambos periféricos tienen incluidos en sus conectores sus respectivas alimentaciones. Mientras que el pulsador de control realiza una conexión entre 5v y el pin de control de la placa.

Por otro lado, tenemos la red eléctrica, que se le conecta en paralelo el optocoplador 4N25, y el output de este último, al pin de cruce por cero de la placa. Y de la placa, sale una señal de control al optocoplador MOC3021, que funciona como driver para el control del triac BT138. Y finalmente, este triac está conectado en serie al foco de luz.

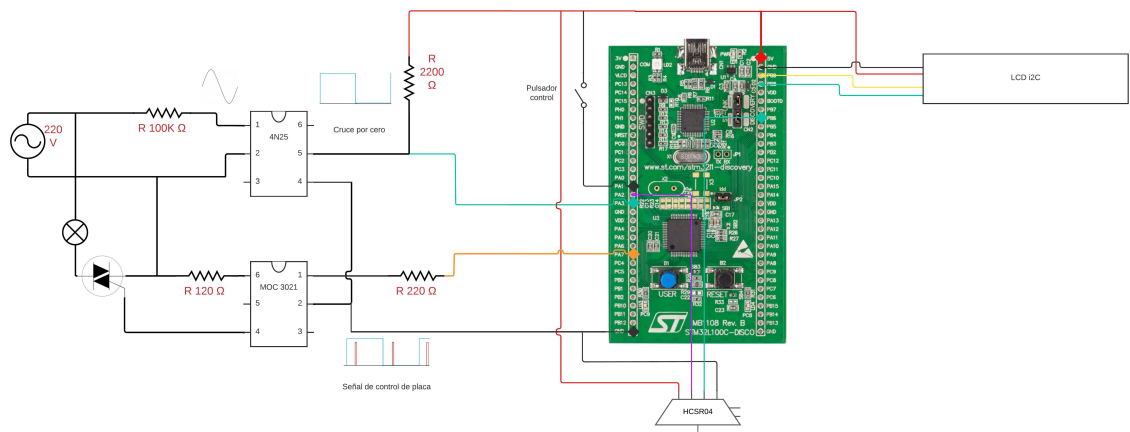


Figura 1: Diagrama de implementación física real

2. Funcionamiento

El proyecto tiene como objetivo controlar la intensidad de la luz de un foco de corriente alterna en Vrms de manera lineal, según la distancia detectada por el sensor ultrasonido. En primer lugar, el usuario controla el sistema con el pulsador de control, cambiando el estado del sistema entre modo On, Dinámico y Off. Es visible en el LCD el estado del sistema como el porcentaje de potencia en el modo dinámico. Y finalmente, en el modo dinámico, la intensidad de luz es determinada por el sensor de ultrasonido después de ser procesado matemáticamente en el procesador para mantener la linealidad.

Para finalizar, se realiza la programación con RTOS siguiendo el diagrama:

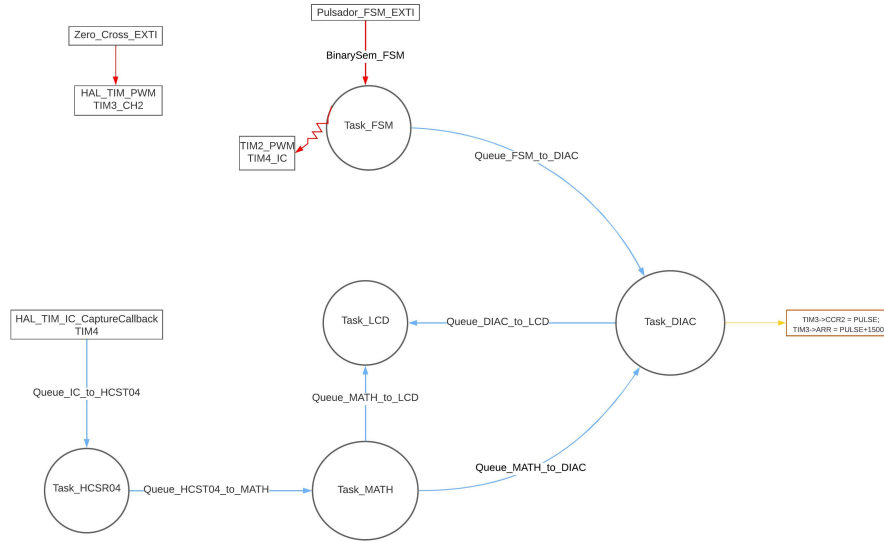


Figura 2: Diagrama de implementación de tareas en rtos

2.1. FSM

Esta tarea permanece bloqueada hasta que se libera el semáforo BinarySem-FSM, que ocurre cuando se realiza la interrupción externa Pulsador-FSM-EXTI cuando el usuario presiona el pulsador de control. Cada vez que la tarea se desbloquea, avanza al siguiente estado el sistema, entre estado standby, On, Dinámico, y Off. Esta tarea apaga los temporizadores TIM2 PWM y TIM4 Input capture en cada estado excepto en el Dinámico que los prende. El estado del sistema se envía como un int a la tarea DIAC a través de la cola Queue-FSM-to-DIAC.

2.2. DIAC

La tarea DIAC permanece bloqueada hasta una actualización del estado (Queue-FSM-to-DIAC), y si se está en el estado Dinámico, también se desbloqueará por los resultados matemáticos (Queue-MATH-to-DIAC). Actualiza los registros CCR2 y ARR del temporizador TIM3, acorde del estado del sistema, o si está en el modo dinámico, por la cola proveniente de la tarea MATH. Al terminar de actualizar TIM3, reenvía el estado del sistema al LCD por la cola Queue-DIAC-to-LCD.

2.3. HCSR04

Tarea bloqueada por la Queue-IC-to-HCSR04, por la cual recibe la cantidad de microsegundos de ancho de pulso del sensor ultrasonido, una muestra. Realiza un promedio de 5 muestras y envía esta por la cola Queue-HCSR04-to-MATH.

2.4. MATH

Tarea bloqueada por la cola Queue-HCSR04-to-MATH. Al recibir este dato, según ciertos parámetros, calcula un porcentaje. Según este porcentaje calcula la cantidad de microsegundos se debe de esperar del cruce por cero para activar el triac que controla el foco de luz. Y finaliza por un lado, mandando el porcentaje como int a la tarea LCD por la cola Queue-MATH-to-LCD, y la cantidad de microsegundos, también como int, a la tarea DIAC por la cola Queue-MATH-to-DIAC.

2.5. LCD

Esta tarea se mantiene bloqueada hasta recibir una actualización del estado del sistema desde la tarea DIAC, por la cola Queue-DIAC-to-LCD, o un porcentaje desde la tarea MATH por la cola Queue-MATH-to-LCD. Esta tarea se encargara de actualizar el LCD por i2C según que estado del sistema, y si se encuentra en el modo dinámico también mostrara el porcentaje.

2.6. Interrupciones y Temporizadores

Se cuentan con las siguientes interrupciones:

- Pulsador-FSM-EXTI : Esta interrupción se activa con un flanco ascendente en el pin A1, por el pulsador de control.
- Zero-Cross-EXTI : Esta interrupción se activa con un flanco ascendente/descendente en el pin A3, por la señal cuadrada del cruce por cero. E inicializa el TIM3 PWM modo un pulso.
- HAL-TIM-IC-CaptureCallback : Esta interrupción se inicia con todos los TIM IC activos, pero se filtra el de TIM4. Cada vez que se detecta un flanco por el pin, se guarda el tiempo en microsegundos del temporizador.

Se procesan estos tiempos y se envía en microsegundos el ancho de pulso echo del sensor HCSR04 por la cola Queue-IC-to-HCSR04.

Se utilizan 3 timmers:

- TIM2 PWM : Señal de control trigger del HCSR04. Pin PA2
- TIM3 PWM : Control de Diac. Pin PA7
- TIM4 IC : Señal Echo del HCSR04. Pin PB6

2.7. I2C

El control por parte de la tarea LCD se realiza por libreria, transmitiendo el mensaje por el pin I2C-SDA (Pin PB9) e I2C-SCL (Pin PB8).