Sensor de estacionamiento

Programación de Microcontroladores

Trabajo práctico final

Autor:

- Sr. Juan Manuel Hernández

Profesor:

- Ing. Patricio Bos

22Co24

Índice

Recorrido por la presentación

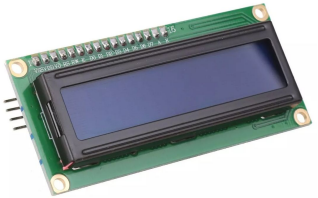
Guía

1. Aplicación
2. Periféricos
3. Máquina de estados finitos
4. Módulos
5. Detalles de código

Aplicación

Sensor de estacionamiento

- Desarrollo sobre la plataforma NUCLEO-F429ZI de ST.
- Lectura de proximidad a objetos sólidos mediante el sensor ultrasónico (HC-SR04) y el cálculo temporal del ancho de los pulsos.
- Procesamiento de los datos medidos para darle un valor de acuerdo a la resolución preestablecida (8 niveles).
- Detección de la señal de marcha atrás o reversa, representada mediante un botón.
- Escritura a una terminal serie de los valores medidos a través del protocolo UART para debugging.
- Escritura mediante I2C de los valores obtenidos en el display (PCF8574 - HD44780).



Periféricos

Periféricos

UART

- USART3
- 115200 - 8N1
- TX -> PD8
- RX -> PD9

I2C

- I2C1
- SCL -> PB8
- SDA -> PB9
- 100 kHz

GPIO

- TRIGGER PIN -> PE2
- REVERSE PIN -> PC13

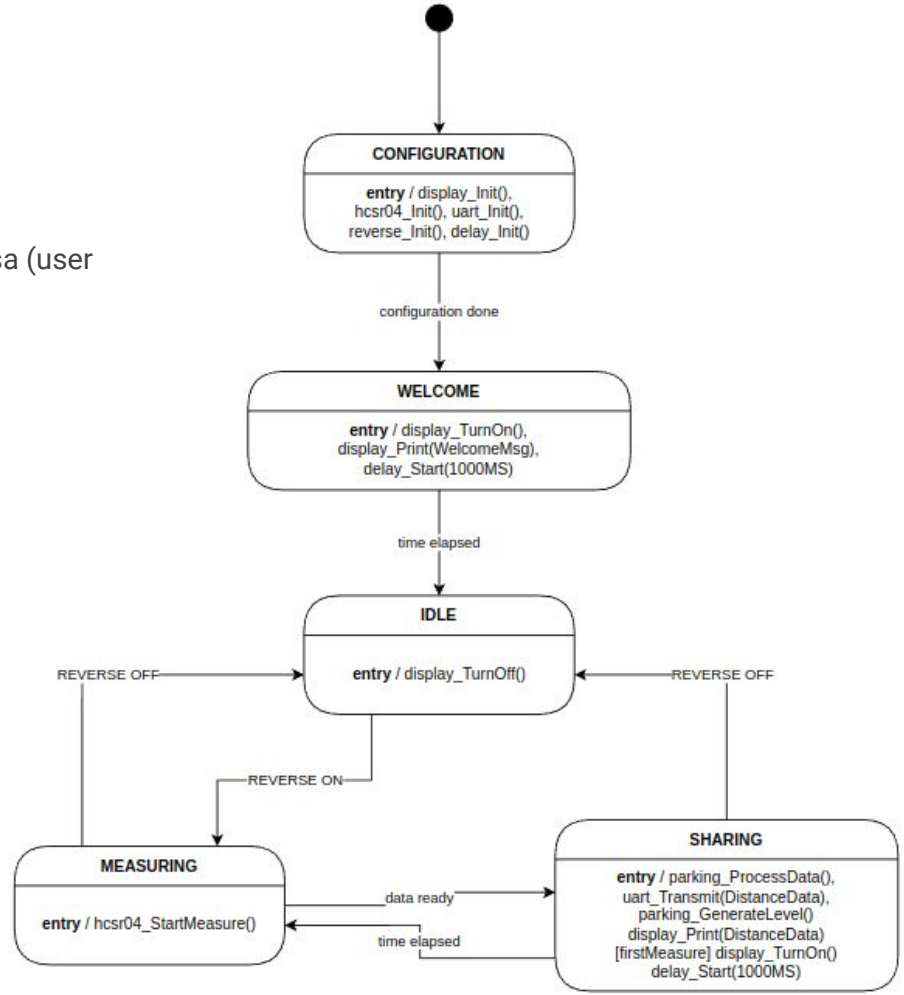
TIM

- TIM1
- ECHO PIN -> PE9
- TIM1_CC_IRQn (ambos flancos)

Máquina de estados finitos

Máquina de estados finitos

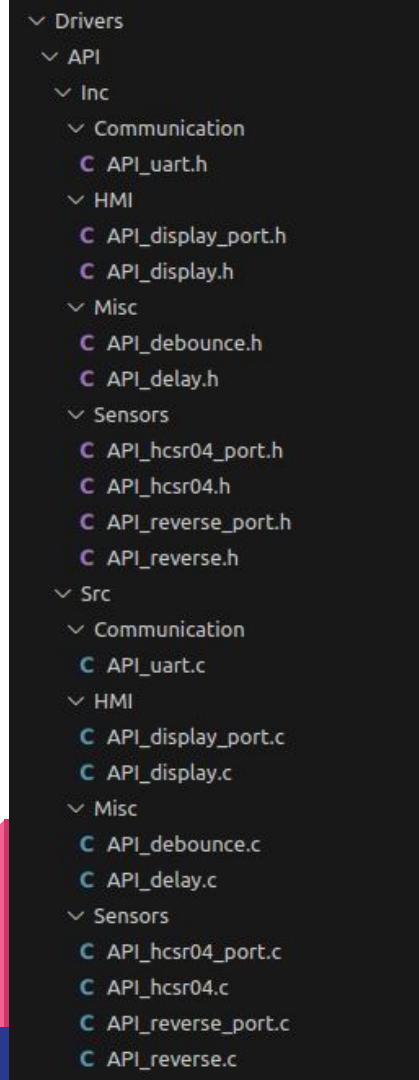
- CONFIGURATION
  - Inicializo display, sensor ultrasónico, UART, señal de reversa (user button) y timers.
- WELCOME
  - Prendo display.
  - Escribo mensaje de bienvenida.
  - Inicio 1s timer.
- IDLE
  - Apago display.
- MEASURING
  - Inicio la medición del sensor ultrasónico.
- SHARING
  - Transformo la información recibida a nivel de resolución.
  - Envío el dato por UART y lo escribo en el display.
  - Prendo el display si es la primera medición.
  - Inicio 1s timer.



Módulos

Módulos

- Se separó en módulos de acuerdo a funcionalidades comunes.
  - Comunicación con el exterior (UART).
  - Interfaz de usuario (Display).
  - Herramientas y utilidades (debounce y delay no bloqueante).
  - Sensado de señales externas (ultrasónico y reversa).
- Los drivers del display, ultrasónico y reversa fueron planteados de forma genérica para no tener dependencias con el hardware. Se implementó una capa de aplicación y una capa de bajo nivel (port.c).



¡Gracias!