Examples

Included examples are:

ARM Cortex M4 Asembler

- `asm_lpc_open`: assembler with LPC Open library
- `asm_sapi`: assembler with sAPI library

C language

- **sAPI library**
  - Bare metal:
    - `adc_dac`: ADC and DAC example.
    - bluetooth
      - `hm10_uart_bridge`: Bluettoth HM10 module AT, tx and rx example.
    - `cycles_counter`: clock cycles counter functions, only work in debug mode. Allows execution time trazability.
    - `dht11_temp_humidity`: humidity and temperature sensor.
    - gpio
      - `gpio_01_switches_leds`: each switch drives the upper led.
      - `gpio_02_blinky`: the simply led blinky with a blocking delay.
      - `gpio_03_blinky_switch`: led blinky with a with a non-blocking delay, to allow you to respond to a switch at the same time.
      - `gpio_04_led_sequences`: led sequences by using a non-blocking delay.
    - `hcsr04_ultrasonic_sensor`: HC-SR04 utrasonic distance sensor.
    - i2c:
      - IMUs
        - `mpu9250`: MPU9250 IMU 9DOF (Gyroscope, Accelerometer and Magnetometer) + Temperature.
      - magnetometers. In Chinese GY-273 module you can have one of this magnetometers, that have the same pinout but different register map. To difference them see the chip, ignore the board serigraphy.
        - `hmc5883l`: HMC5883L magnetometer.
        - `qmc5883l`: QMC5883L magnetometer.
    - `keypad_7segment`: Drives a keypad and 7 segment display.
    - `lcd`: Drives a LCD display.
    - pwm
      - `pwm_01`: PWM applied to LEDs.
      - `pwm_02_rgb_controller_uart`: RGB LED example.
    - `rtc`: RTC peripheral to have date and time clock.
    - `sapi_basic_example`: a basic sAPI library example
    - `servo`: angular servomotor PWM control example.

        - spi_sd_card
            - `spi_sd_card_01_wite_file`: Wite a text file in a SD/MicroSD Card (SPI connected) by using a FAT File System (Elm-Chan FAT FS).
            - `spi_sd_card_02_log_time_stamp`: Wite a text file in a SD/MicroSD Card (SPI connected) by using a FAT File System (Elm-Chan FAT FS) add time-stamp in samples.
            - `spi_sd_card_03_list_files`: List files in a SD/MicroSD Card (SPI connected) by using a FAT File System (Elm-Chan FAT FS). Show results in UART.
        - `tick_hook`: periodic tick function (interrupt-based) with periodic callback.
        - uart
            - `uart_01_echo`: UART echo, it respond the same that you send from PC.
            - `uart_02_receive_string_blocking`: waits until receive a certain pattern String in a UART or timeout expire (blocking code). Don't save received string.
            - `uart_03_receive_string`: waits until receive a certain pattern String in a UART or timeout expire (non-blocking code). Don't save received string.
        - wifi_esp8266
            - `wifi_esp8266_01_uart_bridge`: use this to send AT commands directly to ESP01 module.
            - `wifi_esp8266_02_thingspeak`: send data to thingspeak dashboards.
        - usb
            - `usb_device_01_hid_keyboard`: USB Device. Board as USB Keyboard.
            - `usb_device_02_cdc_virtualSerialPort`: USB Device. Using USB OTG as a virtual serial port.
    - Operating Systems:
        - rtos_cooperative (see M.J. Pont's book at https://www.safetty.net/publications/pttes)
            - `scheduler_01_seos`: first cooperative scheduler
            - `scheduler_02_seos_background_foreground`: second cooperative scheduler
            - `seos_pont_2014_01_first_app`: third cooperative scheduler
        - rtos_freertos
            - dynamic_mem
                - `freeRTOS_01_blinky`
                - `freeRTOS_03_ChanFatFS_SPI_SdCard_ADC_log`: ADC logging in a SD/MicroSD Card (SPI connected) by using a FAT File System (ChanFS), freeRTOS and sAPI.
            - static_mem
                - `freeRTOS_01_blinky`
- **LPC Open library**
    - bare_metal
        - `blinky`
        - `blinky_input`
        - `blinky_ram`
        - `blinky_rit`
        - `boot`
        - `i2c`
        - `lpc_open_basic_example`
        - `mpu`

        - `pwm`
    - operating_systems
        - `freertos_blinky`
    - statecharts
        - `statechart`
- `newlib_printf_scanf`
- `without_libs`

C++ language

- `cpp_sapi_basic_example`: a C++ basic example with sAPI library.

More information

Back to README.