**STM32 FreeRTOS LED Blinking & UART Communication**

**Overview**

This project demonstrates a simple FreeRTOS-based application running on an STM32F103 microcontroller. It features two tasks:

* **LED Blinking Task**: Toggles an LED connected to PC13 at a fixed interval.
* **UART Communication Task**: Sends a character ('A') over USART1 and prints debug messages using printf.

**Features**

* Uses **FreeRTOS** for multitasking.
* **LED Blinking Task**: Toggles the LED every 500ms.
* **UART Task**: Sends data every 1 second and outputs debug messages via printf.
* Configures **SysTick** for FreeRTOS scheduling.

**Hardware Requirements**

* STM32F103-based development board (e.g., Blue Pill)
* USB-to-serial adapter (for UART debugging)
* LED connected to PC13 (onboard LED for most STM32 boards)

**Software Requirements**

* STM32 Standard Peripheral Library (STM32F10x)
* FreeRTOS
* Keil uVision IDE

**Code Structure**

**Initialization Functions**

* **ledInit()**: Configures GPIOC Pin 13 as output for LED blinking.
* **uartInit()**: Configures USART1 (TX on PA9) for serial communication at 9600 baud.
* **setupSysTick()**: Configures the system tick timer for FreeRTOS task scheduling.

**FreeRTOS Tasks**

* **vTaskLedBlink()**:
  + Toggles the LED every 500ms.
  + Prints "LED Blink Task Executed!" via UART.
* **vTaskUartSend()**:
  + Sends the character 'A' over USART1.
  + Prints "UART Task Executed!" via UART every 1 second.

**fputc() Function**

* Redirects printf() to USART1, allowing formatted debug output.

**Build and Flash Instructions**

1. Install Keil uVision IDE.
2. Compile the project and generate the firmware binary.
3. Use **ST-Link** or a **USB-to-serial adapter** to flash the firmware.
4. Open a serial monitor at **9600 baud** to view debug messages.

**Expected Output**

On a serial monitor (9600 baud), you should see:

LED Blink Task Executed!

UART Task Executed!

LED Blink Task Executed!

UART Task Executed!

...

Additionally, the onboard LED should blink at **500ms intervals**.

**License**

This project is open-source and available under the MIT License.

**Author**

Developed by [Manvendra and Raju].