

STM32F103 LED Blinking Lab Guide

Objective: Learn the basics of STM32 microcontroller development by creating a simple blinking LED program using the STM32F103 (e.g., BluePill board). This lab uses STM32CubeIDE and references the STM32F103C8 datasheet.

Required Hardware & Software

- STM32F103C8 development board (e.g., BluePill)
- ST-Link programmer/debugger
- PC with STM32CubeIDE installed
- USB cable

Datasheet Highlights

The STM32F103C8 is based on an ARM Cortex-M3 core running up to 72 MHz. It operates at 2.0–3.6 V and provides multiple GPIO ports (A, B, C). GPIO pins can be individually configured as inputs or outputs and are clock-gated.

Project Setup (STM32CubeIDE)

- Create a new STM32 project in STM32CubeIDE.
- Select MCU: STM32F103C8.
- Name the project (e.g., Blink_LED).
- Finish and open the CubeMX configuration.

GPIO Configuration

Select pin PC13 and configure it as GPIO Output (Push-Pull). Set speed to Low and disable pull-up/pull-down resistors. On most BluePill boards, PC13 is connected to the onboard LED and is active-low.

Blink Code Example

Inside main.c, after initialization, configure GPIOC and toggle PC13 inside the main loop with a delay of 500 ms between state changes.

Build & Flash

- Build the project.
- Connect ST-Link to the board.
- Flash and run the program.

Expected Result

The onboard LED should blink continuously with a visible on/off period of approximately 500 ms.

Next Steps

After completing this lab, explore GPIO input, timers, UART communication, and ADCs to deepen your STM32 embedded systems knowledge.