Digital Systems Electronics Laboratory 07

Group 13

s295391 Giorgio Zoccatelli
s295567 Vittorio Macripò

Due Date:	May 20, 2024
Delivery date:	May 19, 2024
Instructor:	Professor Guido Masera

Politecnico di Torino
Accademic Year 2023/24

Contents

1 Introduction

The aim of this laboratory is to get familiar with Nucleo board STM32F401RE by handling the functioning of the LED. The procedure will firstly consist of performing a Low-Level Approach by modifying directly the registers, followed by an higher approach through the use of a STM32CubeIDE dedicated tool.

2 Controlling the LED

2.1 Using a switch to switch on/off the LED, Low Level Approach

As can be seen in the file *lab_7_es1.1* the whole procedure is based on the manipulation of regiters and through the use of appropriate masks.
The result reflects the expectations and the **LED** (LD2 or green led) is turned on whenever the **pushbutton** (or user button) is pressed.

2.2 Using a switch to switch on/off the LED, STM32Cube Approach

The procedure of the first point is repeated with a different approach. As can be seen in the Figure below, a specific tool is used to determine the specific funcitoning of the pins.

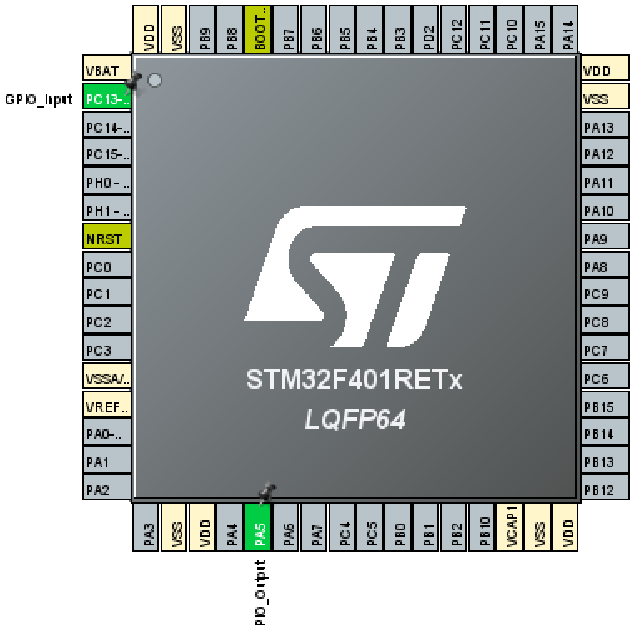

Figure 1: Pin Configuration

2.3 Varying the blinking frequency

The last variant of the operation consists of blinking the LD2 at a frequency that is doubled at each press of the user button starting from a frequency of 0.25 Hz.

At the beginning of the code a INIT constant equal to 6000000 is initialized in order to set the correct timing of the working frequency of the LED. The value is chosen after repeated and multiple operation tests.

3 Generating a Square Wave

A C project is realized with the purpose of using the Nucleo board to realize a square wave generator with 1khz working frequency. Whenever the wave changes its state (from low to high or viceversa) the LED changes its state as well. The whole procedure is then tested through the use of a waveform viewer as explained in the Appendix of the laboratory document.
As it was done before, a constant WAIT is used to set the correct timing of operation.
As indicated in the document, a portion of code has been added in order to enable the interrupt function via the SysTick Timer that starts its own ISR. This results in a temporary suspension of execution of the main() for a time that depends on the function implemented in the ISR, where a variable **x** is iterated through a cycle for and at the same time updated.
This results in a change in the blinking frequency of the LED, which will keep its current state until the ISR finishes.

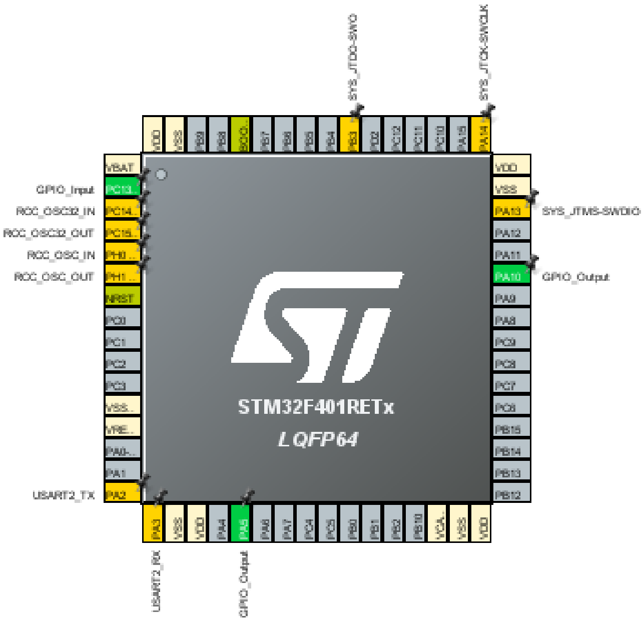

Figure 2: Pin Configuration