**Pomodoro clock Architecture**

**Version 1.0**

**April 29, 2016**

**Author: Kaloyan Nikolov**

Table of Contents

[1. GitHub 3](#_Toc451014600)

[1.1. Overview 3](#_Toc451014601)

[1.2. Setting up a GitHub Project 3](#_Toc451014602)

[2. Board Testing 3](#_Toc451014603)

[2.1. Board Upgrade 3](#_Toc451014604)

[2.2. Test Code 4](#_Toc451014605)

[2.3. Test run and results 5](#_Toc451014606)

[3. Component Dependencies 6](#_Toc451014607)

[4. Logical flow 6](#_Toc451014608)

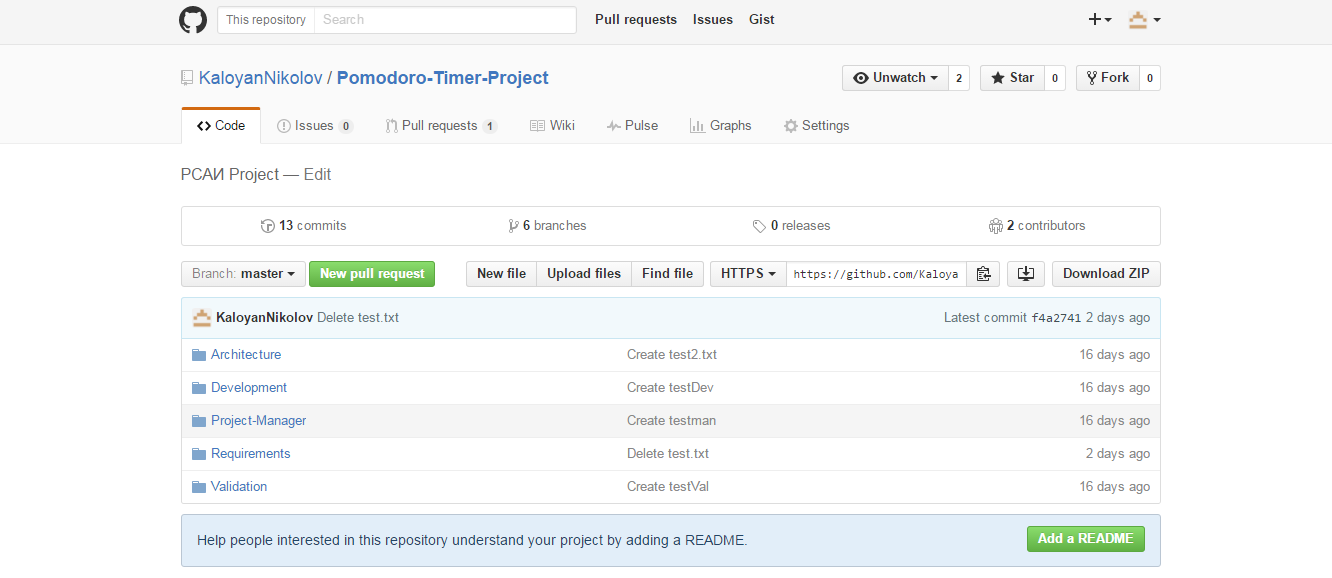
# GitHub

## Overview

In order to work together on a project, a team needs a platform, where they can upload their work documents and also communicate easily. GitHub is a platform used by millions of developers to create projects and work together on open source technologies. GitHub is easily accessible and user friendly.

## Setting up a GitHub Project

As the project architecture, it’s my duty to set up a GitHub project. In order to make that GitHub requires a user profile in the system. After registration or sign up, a project can be easily made. After the creation of the project, the user is given a master branch in which everybody, the project creator gives permission to, can create folders and files and also upload and download files. Our GitHub project is presented in Picture 1.



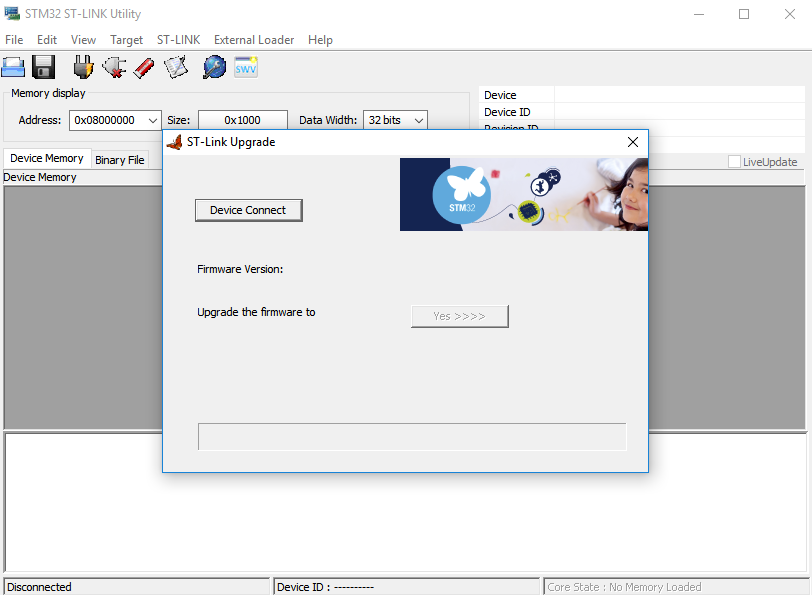
Picture 1.

# Board Testing

This section, describes the tests made in order to determine if the provided board is working properly.

## Board Upgrade

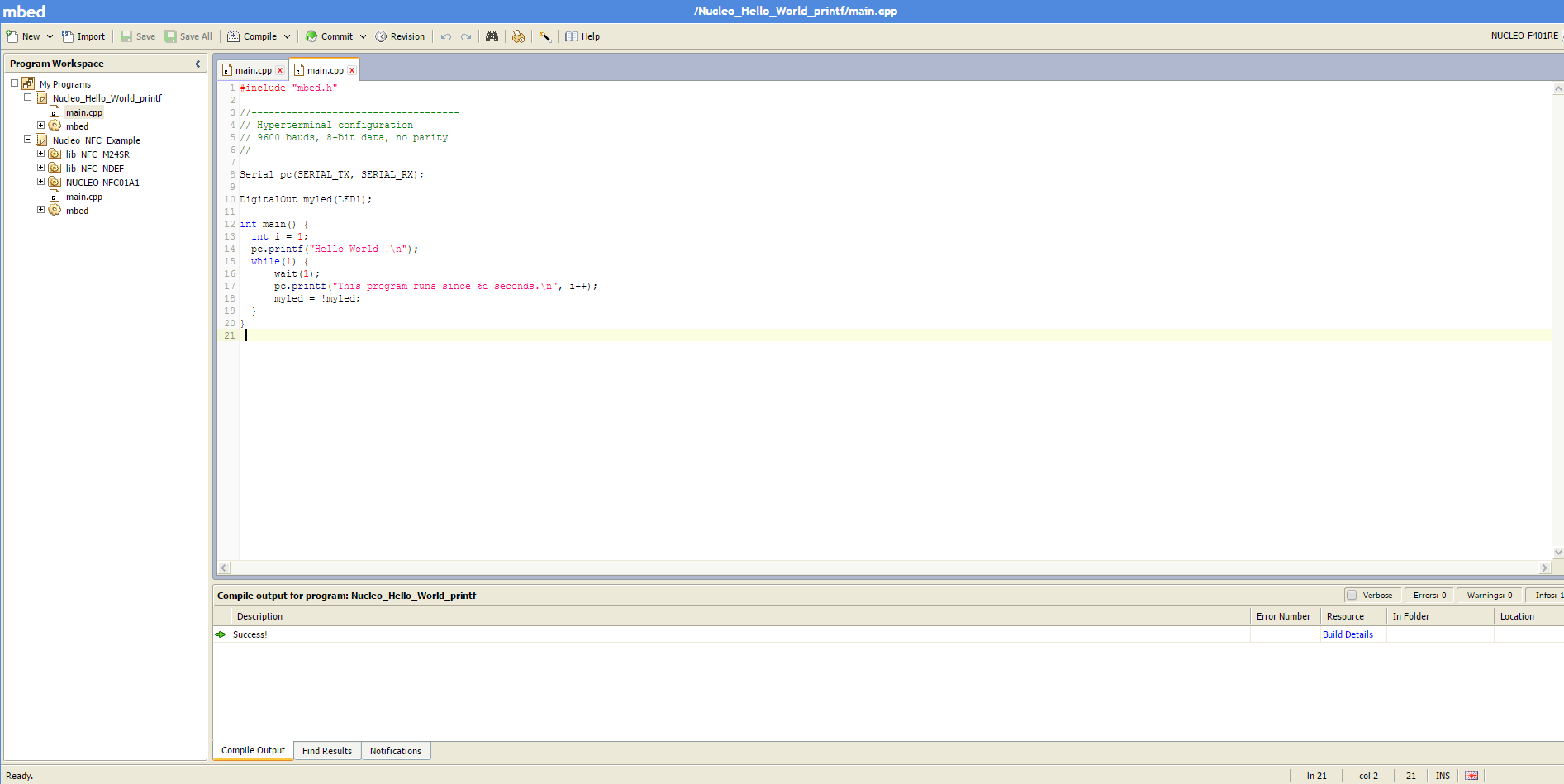
The board has to be upgraded to the latest firmware version, so the developer and the user have the newest board capabilities. STM32 ST-Link Utility is the software used to upgrade the Board firmware as displayed on Picture 2.

es

Picture 2.

## Test Code

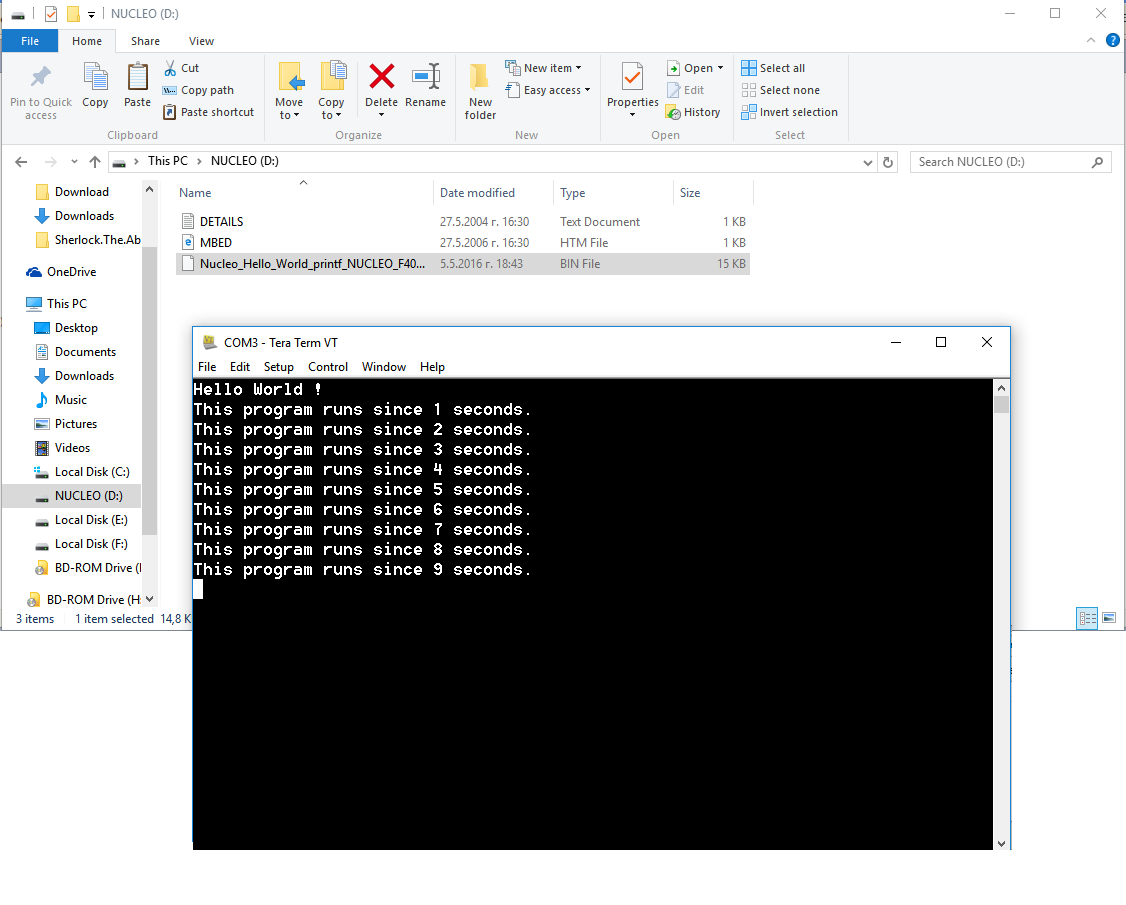
The test code is written on the mbed online compiler. The mbed online compiler requires a user profile in their system. The code represents a widely used test program - “Hello World” with a seconds counter. The program prints out the text – “Hello World!” in the console and starts a timer that counts the seconds since the start of the program. The code is shown on Picture 3.



Picture 3.

## Test run and results

After the code compilation a .bin file is created. That .bin file is moved to the board directory. The next step is to open a Terminal application that establishes connection with the board through the COM port. Tera term is used as a Terminal application. Here are the test results(Picture 4):

Picture 4.

# Component Dependencies

Components:

* STM32 F4 01 Nucleo
* Display (unknown for now)
* Buzzer or Speaker (unknown for now)

The program components are closely related. The other devices can’t work without the Nucleo board.

The display is the main component, that works like a user friendly interface. The display will make the program really easy to understand and use. The display will show how many periods of times have already run out and will also work as a loading bar. The display component is dependent on the board, so the board has to function properly.

The buzzer is the main component, that alerts the user for the end of the period. The buzzer has to work correctly and has to go off in the right time – when the display shows, that the period time has ended.

# Logical flow

The program has to be easy to work with and has to follow the logical plan. The user starts the timer by using one of the board buttons. The board has to start the inside timer in the code and has to start the loading bar progression on the display. The buzzer has to go off when the time runs and loading bar reaches the max value.