

## GUÍA DE TRABAJOS PRÁCTICOS N° 1

### Placa de Desarrollo

- Utilizando la información disponible (Ver material en el aula virtual de la materia) realice un resumen de las características más importantes de la plataforma BLUE-PILL con la que trabajará en clase.
- Utilizando la información de los terminales de la placa (Ver [STM32F103C8T6-Blue-Pill-PIN-OUT.gif](#)), analice cuales son los periféricos del MCU que están disponibles para el desarrollo de aplicaciones. Puede analizar también el circuito esquemático de la placa ([BluePill-Schematic.png](#)) para obtener información de donde se conectan los elementos básicos.
- Daremos nuestros primeros pasos utilizando el compilador de KEIL y la simulación mediante Proteus. Analizar, Comprender, Compilar y simular los ejemplos de la clase 2.

### Microcontrolador STM32F103

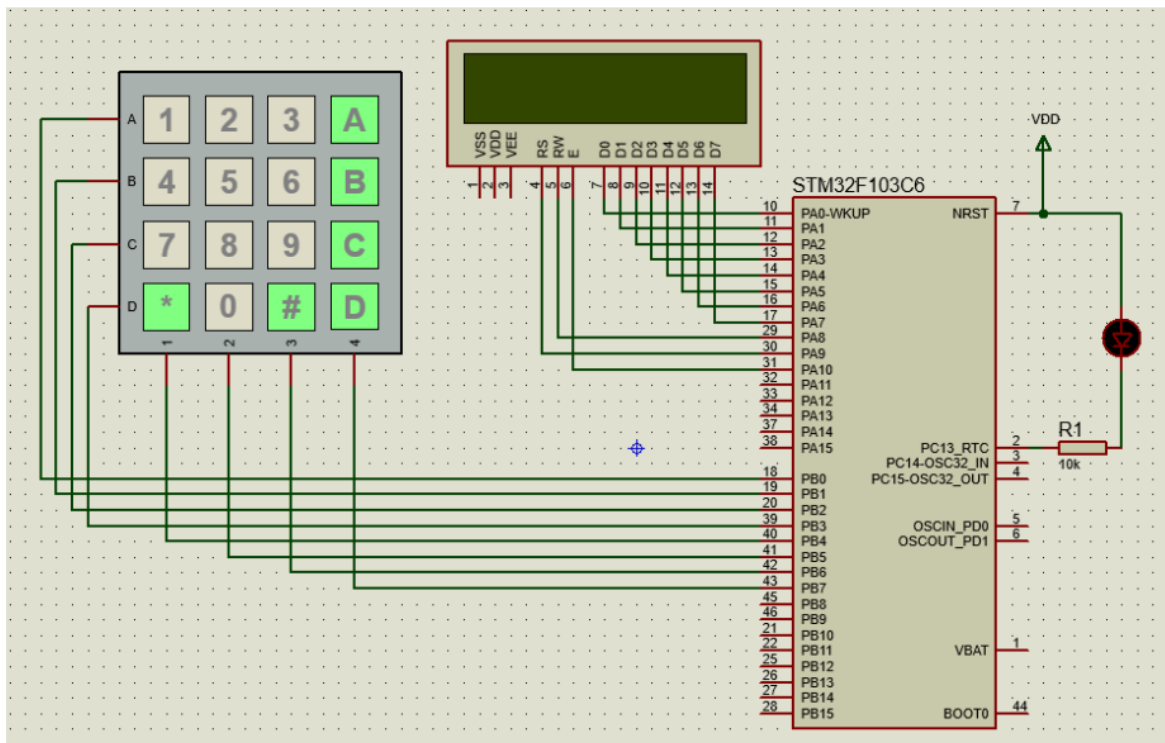
- Investigue y haga un resumen de las características más destacadas del MCU. Describa la arquitectura interna y enumere los periféricos que contiene (ver Data Sheet y Manual de usuario).
- Investigue sobre las características eléctricas de los puertos GPIO (ver especificaciones del Data Sheet del MCU) y sobre las características funcionales de los GPIO. Haga un diagrama del circuito digital de un GPIO y explique cómo funciona.
- Investigue cuales son los registros del MCU para configurar un GPIO como salida y las diferentes maneras de modificar el estado de dicha salida.
- Ídem anterior para configurar un GPIO como entrada y leer su estado .

### Ejercicio con informe grupal:

- El proyecto Grupal será de hasta 3 alumnos y con nota de calificación.
- El tiempo de ejecución de los proyectos se estima en 2 semanas.
- El alcance del proyecto (estimado en horas) se considera de 4hs semanales además de las 4 hs semanales de las clases adicionales para cada integrante.

### Enunciado:

Implementar con el MCU la cerradura electrónica que ya implementamos en el TP2 de CDyMC. Para esto ahora disponemos de un display LCD de 2 líneas, de un teclado matricial 4x4 y del STM32F103C6. La implementación deberá hacerse aplicando la arquitectura de planificador despachador temporizada con un tick de sistema (sEOS) y desarrollando bibliotecas de funciones básicas para utilizar el LCD y el teclado.



#### Requerimientos:

- Cuando el equipo se inicia deberá mostrar en la primer línea del LCD un reloj con formato HH:MM:SS y en la segunda línea el estado de la cerradura “CERRADO”.
- El sistema debe tener la clave numérica guardada por defecto 4321 de manera de poder activar o desactivar la cerradura. Si el usuario presiona la clave correcta se mostrará en la segunda línea “ABIERTO” durante 0.5 seg y luego volverá automáticamente al estado por defecto. Además se deberá activar un LED indicador en un terminal de salida.
- Si la clave es incorrecta se mostrará el estado “DENEGADO” durante 0.2 seg y luego volverá automáticamente al estado por defecto.
- Cuando se presione un tecla numérica se deberá mostrar ‘\*’. para indicar que el sistema está recibiendo las entradas. El resto de las teclas no tienen ninguna función.

Recuerde los pasos necesarios para resolver el problema y realizar el informe.

- Interpretación del problema propuesto.
- Resolución del problema: Dado que se trata de un problema ya resuelto con el Atmega328p nos vamos a enfocar en las diferencias de implementación con el nuevo MCU. Analice y describa con qué hardware cuenta, realice esquemas de conexión, explique la configuración de los distintos periféricos que utilizará, etc. En cuanto al software, describa el modelo de estados, cuáles son las tareas que se requieren y la temporización de las mismas, cual es el formato que adopta para la visualización, como

será la interfaz con el usuario, de la búsqueda de claves, entre otros. Indique cómo es la modularización del programa y la descomposición en funciones.

- c. Pseudocódigo: Realice el pseudocódigo y/o diagramas de flujo y/o máquinas de estado para describir la solución adoptada para cada parte del programa.
- d. Código: Realice el Código en C. Recuerde utilizar comentarios para documentar el mismo e indentar adecuadamente. El programa deberá coincidir con el pseudocódigo definido en el paso anterior.
- e. Validación: Realice una explicación sobre la simulación y concluya si la implementación cumple con los requerimientos pedidos. Adjunte los archivos de código fuente y simulación para su corrección.