

Índice

1. Introducción

En el siguiente trabajo se buscó implementar el control de acceso a un edificio de oficinas mediante una tarjeta magnética y un pin (clave de acceso). Para llevar a cabo dicho objetivo, se valió del uso de la placa de desarrollo Freedom-K64F (FRDM-K64F), un encoder rotativo, un display de siete segmentos y un lector de tarjetas magnéticas.

2. Implementación

2.1. Maquina de estados

El programa fue planteado como una maquina de estados, existiendo nueve estados diferentes. A su vez, cada estado es tratado como una FSM más pequeña.

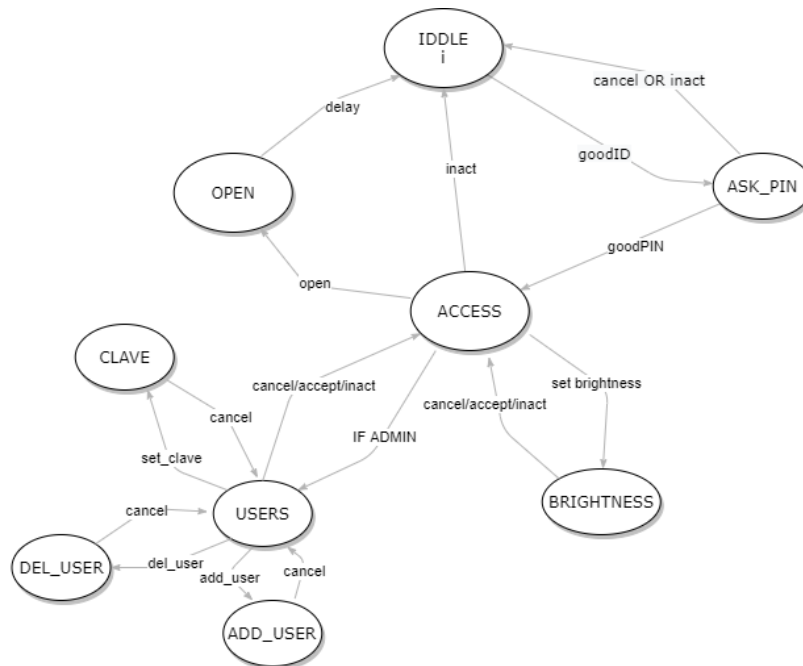


Figura 1: FSM del programa.

El estado **IDLE** es el estado principal por defecto, en el cual se debe ingresar un ID empleando tanto las dos alternativas posibles (mediante encoder o mediante la tarjeta). Al realizar la validación de dicha ID, se pasa al estado **ASK PIN**.

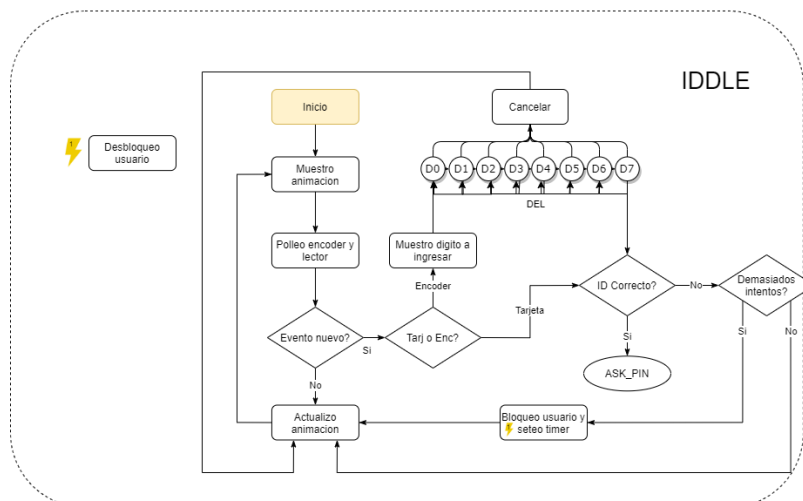


Figura 2: FSM del estado **IDLE**.

El estado **ASK PIN** se obtiene el pin de acceso del usuario, que las ser validado se continua al estado de **ACCES**.

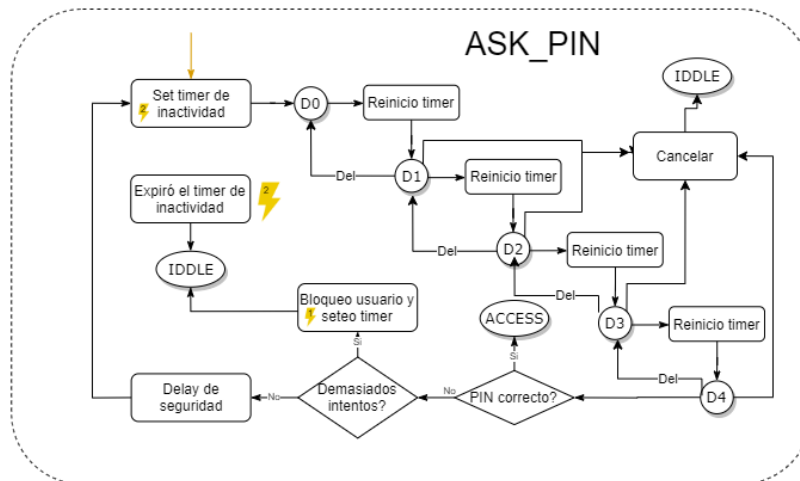


Figura 3: FSM del estado **ASK PIN**.

En **ACCESS** puede retornarse al estado **IDLE** (luego de cierto período de inactividad), al acceso de del edificio abriendo la puerta o a un panel de configuración del sistema.

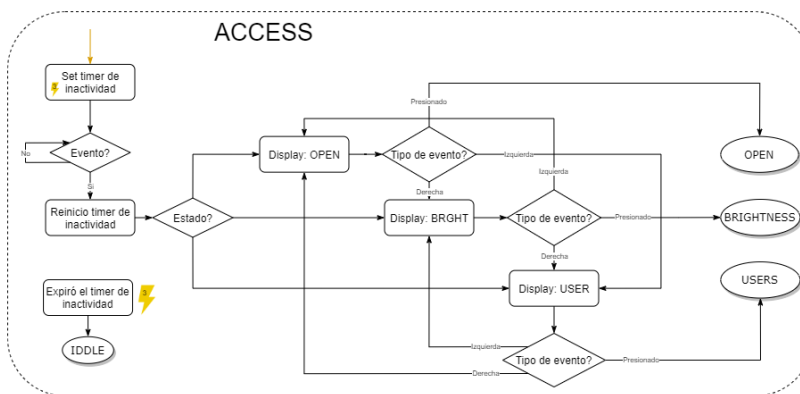
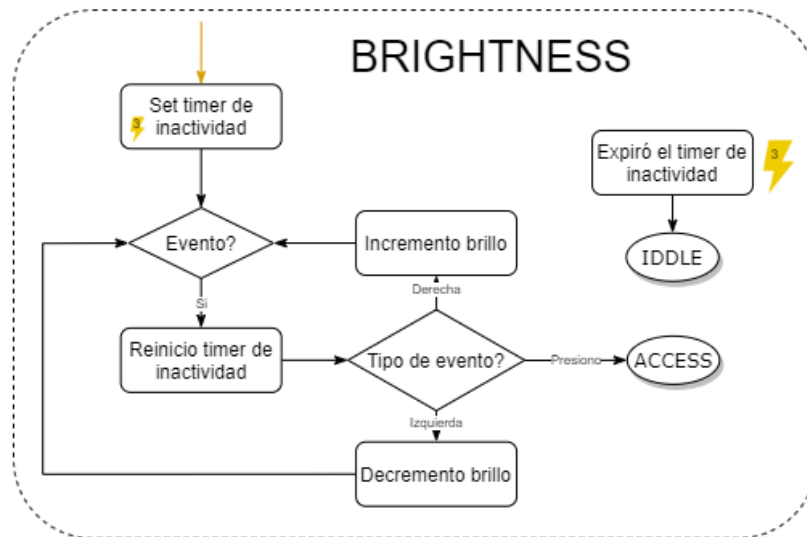
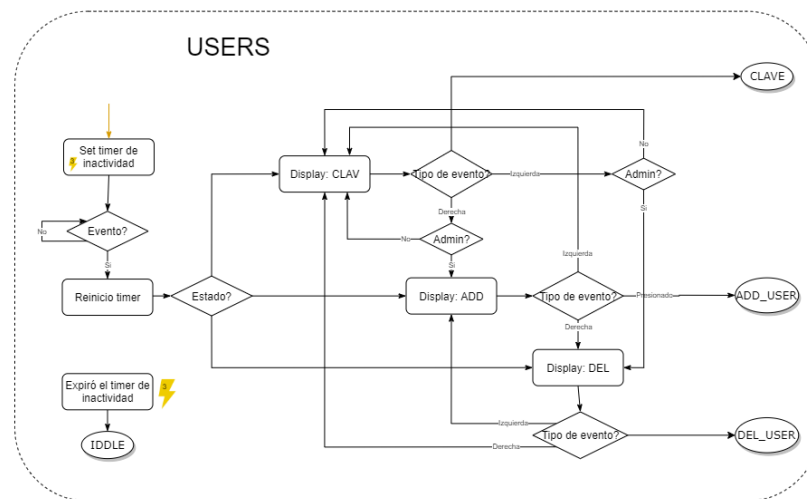


Figura 4: FSM del estado **ACCESS**.

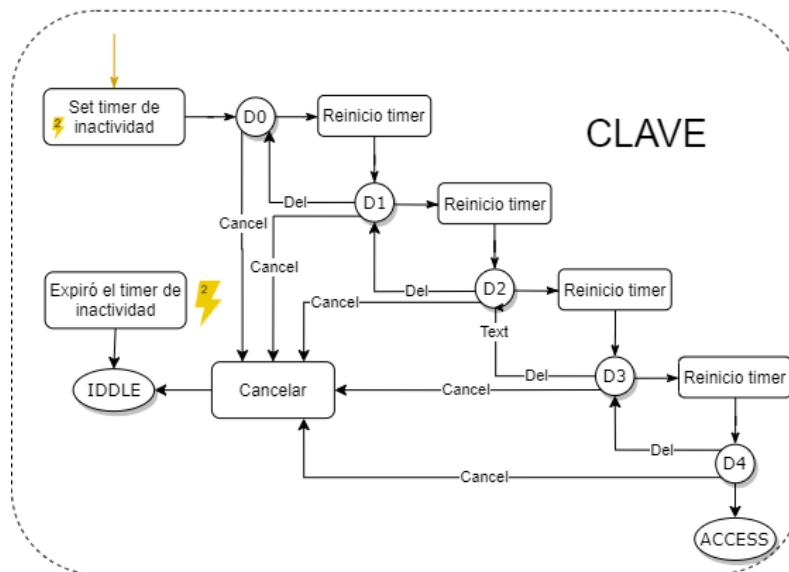
Dicho panel de configuración se encuentra conformado por dos estados distintos, siendo estos el estado **BRIGHTNESS** y **USERS**. El primero permite cambiar el brillo del display empleado, mientras que el segundo otorga ciertas configuraciones diferenciadas.

Figura 5: FSM del estado **BRIGHTNESS**.

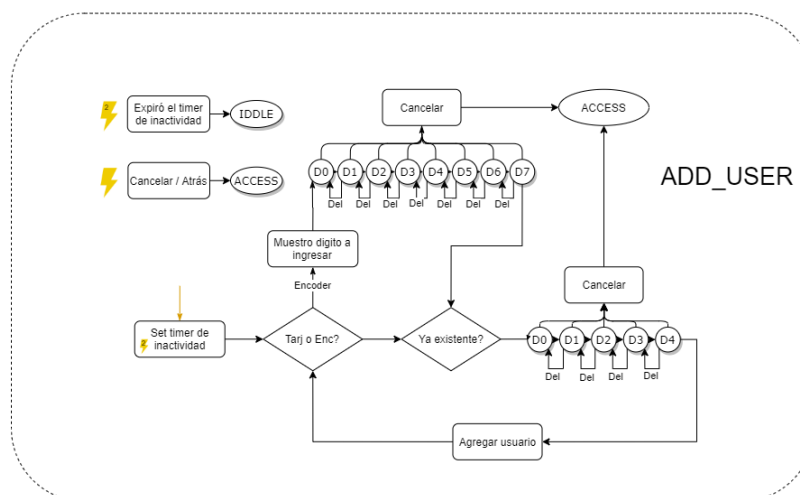
Cabe destacar que cualquier usuario puede acceder tanto a la configuración del brillo como de usuarios, pero dentro de este último estado existen ciertas limitaciones.

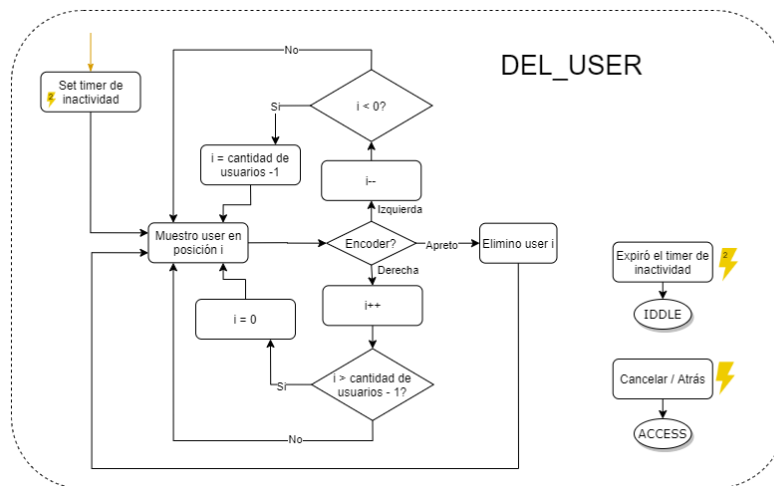
Figura 6: FSM del estado **USERS**.

Cualquier persona puede acceder al estado **CLAVE**, donde puede cambiar su propio pin de seguridad. Pero solo los que cuentan con el rango de administrador son los que tienen acceso a los dos estados restantes.

Figura 7: FSM del estado **CLAVE**.

Los estados exclusivos de **ADD USER** y **DEL USER** permiten que cualquier administrador agregue a un nuevo usuario o elimine a uno existente respectivamente.

Figura 8: FSM del estado **ADD USER**.

Figura 9: FSM del estado **DEL USER**.

2.2. Capas e interfaces

Se separó el programa por capas, siendo la **App** la principal y subdividiendo la capa de **Drivers** en las capas de **HAL** y **MCAL**. Dicha separación puede verse con detalle en la Figura (??), donde se muestra como la **App** entra en contacto únicamente con los drivers de la placa de desarrollo **FRDM**, de la **Placa Verde** y con el **Lector** de la tarjeta.

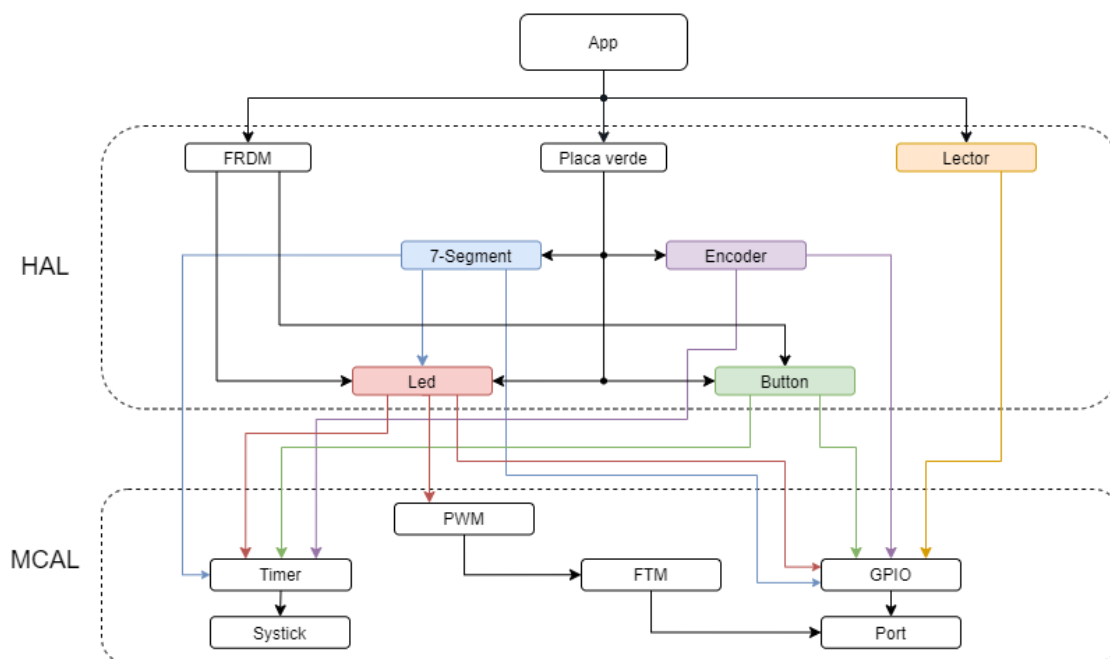


Figura 10: Arquitectura del firmware, jerarquía y relación entre los módulos.

El driver **FRDM** emplea los drivers de **Led** y **Button**, mientras que la **Placa Verde**, además de los dos recién mencionados, se vale del uso de los drivers de **Encoder** y del **7 Segment**. Todos los mencionados junto con el **Lector** conforman la Hardware Abstraction Layer.

Por otro lado, la Microcontroller Abstraction Layer se encuentra conformada por los drivers de **Timer**, **PWM**, **Gpio**, **FTM**, **Systick** y **Port**, siendo estos últimos 3 los únicos que no tienen contacto con drivers de la HAL.

En la Figura (??) se puede apreciar la interconexión entre drivers de las distintas capas, diferenciando por colores las distintas dependencias entre la HAL y la MCAL para un mejor entendimiento.

Se debe agregar un pin de testeo (TP) que se encienda mientras se ejecutan las interrupciones, a fin de medir el tiempo que se emplea en la ISR y cuánto representa porcentualmente.