



Facultad de Ciencias e Ingeniería

2021-1

Sistemas Digitales - IEE 256

Informe Final del Proyecto

Título del proyecto: Sistema de seguridad

Integrantes	Código	Nota
Alessandro Santos Romero	20185868	
Yhamil Favio Munive Calcina	20185913	

Asesor: Raymi Roman, Luis Eduardo

Horario: H052B

Observaciones:

--

Fecha: 26/11/2021

Firma Asesor

Índice

1. Carátula

2. Índice

- Índice con números de página(18 páginas)

3. Resumen del proyecto (1 página)

- Breve descripción de lo que es el proyecto
- Incluye los objetivos del proyecto

4. Requerimientos, especificaciones y restricciones

5. Metodología empleada (1 página)

- Cómo se dividieron el trabajo y el cronograma o diagrama de Gantt
- Deben mencionar además qué parte del informe final hizo cada uno.

6. Descripción del hardware y software del proyecto

- Hardware
 - i. Diagrama de bloques
 - ii. Diagrama esquemático
 - iii. Explicación del funcionamiento del hardware
- Software
 - i. Diagrama de flujo del programa principal
 - ii. Diagrama de flujo de las funciones relevantes
 - iii. Explicación del algoritmo utilizado.

7. Resultados (1 a 2 páginas)

- Aquí se presenta la evidencia del trabajo realizado, explicando las partes que funcionan y qué partes no. Deben mostrar fotos o enlaces de videos relevantes que refuercen lo escrito como evidencia.

8. Conclusiones

- ¿Qué objetivos lograron alcanzar? ¿qué partes no culminaron? ¿por qué?

9. Observaciones y Recomendaciones

- Cualquier sugerencia relacionada con el desarrollo del proyecto, o de cualquier proyecto futuro que quieran mencionar.

10. Bibliografía

- Relación de fuentes bibliográficas o multimedia empleadas (libros, blogs, videos, manuales, artículos etc. revisadas para el desarrollo del proyecto
- 11. Anexos
- Presentar el programa fuente, a espacio simple, tamaño 9, Courier New, como el empleado en las guías de laboratorio para los programas.
- El programa principal y las funciones empleadas deben estar debidamente documentados.
- Si en el proyecto han utilizado funciones o bloques de código no desarrolladas por ustedes, indicar la fuente y el autor).

Resumen de proyecto

Descripción del proyecto:

El proyecto consiste en desarrollar el código para un sistema de seguridad, el cual tiene fines de uso en casa domésticas. El sistema funciona por medio de 2 SWs que permiten ingresar la contraseña y permite configurar el sistema. Así mismo, el sistema cuenta con sensores encargados de monitorear en todo instante la seguridad del hogar.

El sistema incluye una comunicación serial con cualquier dispositivo que desee el dueño del sistema conectar, Este dispositivo permite verificar el monitoreo y el control del sistema de seguridad, así no se encuentre cerca de su hogar.

Los objetivos del proyecto son:

Poder implementar un sistema operativo, por medio de las herramientas y temas desarrollados en el curso. Además, de experimentar el proceso de creación orientado a la industria; Pues se dispone de diversas herramientas, pero no todas serán imprescindibles para el desarrollo del proyecto. Por lo cual el desarrollador tendrá que diferenciar qué herramientas serán indispensables y cuáles no.

El poder desarrollar un proyecto de manera grupal y en constante comunicación con el profesor encargado.

Cumplir con las tareas en el tiempo previsto, pues el avance de cada integrante es indispensable para el desarrollo del proyecto.

Requerimientos, especificaciones y restricciones

El sistema consiste en la implementación de la pantalla Nokia para visualizar lo que es el menú y se implementará la clave por medio de 2 SWs para la máquina principal, la cual también se puede manipular por comunicación serial.

- Al iniciar se mostrará una pantalla con el inicio del título de la marca de seguridad donde pasarán unos 10s se procede a ingresar la contraseña inicial (mediante el SW1 y SW2 la cual consta de 5 dígitos, y además se activará el led Verde representando que se activó el sistema), posteriormente, el sistema se activa y se procede a pasar al menú principal.
- Antes de pasar al menú principal, se tendrá una pantalla que avisará si se ha violado la seguridad del sistema mostrando en qué puerta se ha irrumpido.
- En el menú principal, se tendrá la opción de ingresar la contraseña (para desactivar el sistema), cambiar contraseña, configuración y activar sistema de seguridad. Se moverá mediante los SW1 y SW2 en donde uno sube y el otro baja respectivamente y para seleccionar la opción se presiona los 2 al mismo tiempo.
- En el apartado de ingresar contraseña, se intentará ingresar la contraseña con los SWs en donde por default se intentará 3 veces y solo se ingresará con la contraseña inicial (a menos que se cambie estas 2 opciones en el apartado de configuración y cambiar contraseña respectivamente). En caso sea errónea la clave, se tendrá un mensaje de contraseña equivocada. Al momento de ingresar 3 veces incorrectamente la clave, se bloqueará el sistema por 10 segundos. Al momento de ingresar correctamente la contraseña se desactiva el sistema por lo que representaremos esto mediante el uso del led Rojo del puerto F (el led Rojo en alta y el led Verde en baja).
- En la opción de cambiar contraseña, se actualizará la contraseña actual por la que se ingrese pero siempre confirmando la anterior. Se tendrá que cambiar el orden que se ingrese los SWs para la contraseña nueva de 5 dígitos.
- En configuración, se podrá cambiar el intentos para poner correctamente la contraseña hasta un máximo de 10 intentos y un mínimo de 3.
- Por último, en el apartado de activar sistema de seguridad se podrá volver a activar el sistema seguridad valga la redundancia.

Metodología empleada

La metodología empleada para el desarrollo del proyecto, así como el informe final fue por medio de tareas, esquematizada en el presente cuadro:

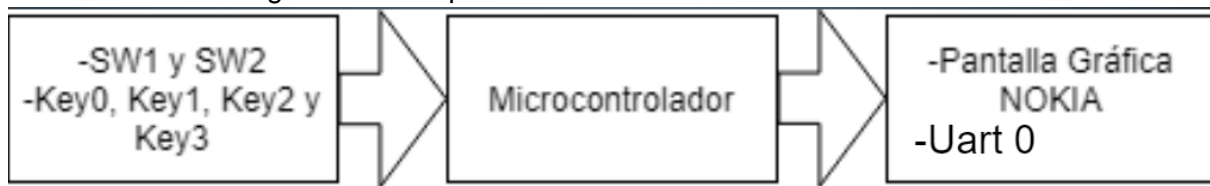
Tareas	1-5/11	8-12/11	15-19/11	22-26/11
1	M			
2		M		
3	R			
4		R		
5			A	
6				A
7				A

M	Integrante 1:Munive Calcina,Yhamil Favio
R	Integrante 2:Romero Santos, Alessandro
A	Ambos integrantes

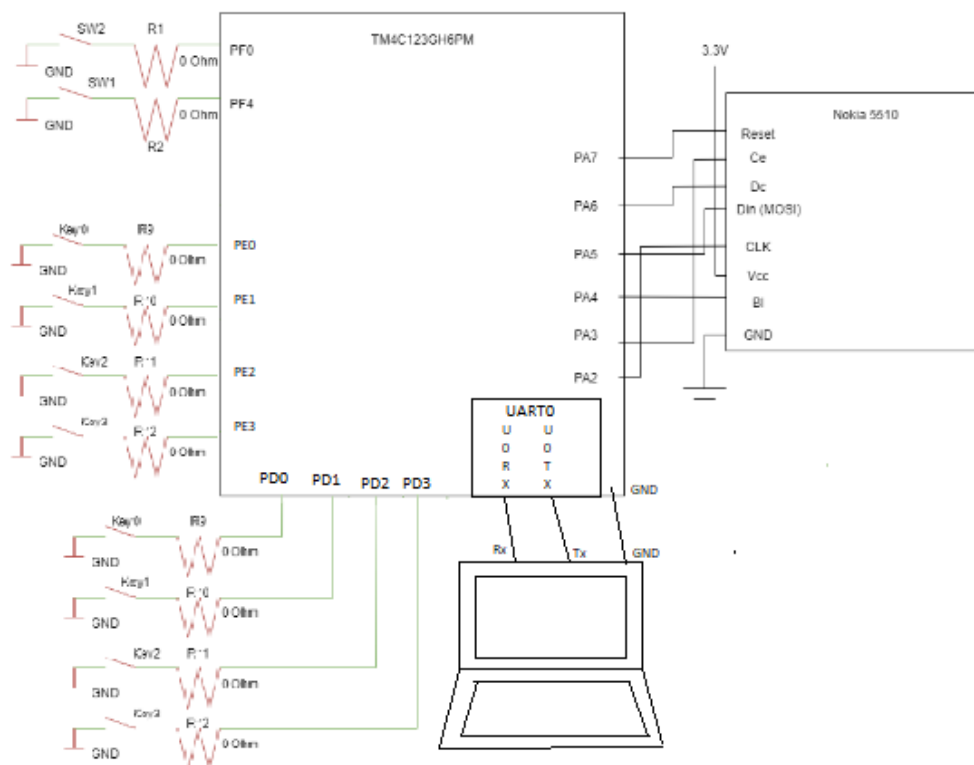
Descripción del hardware y software del proyecto

Hardware:

i. Diagrama de bloques



ii. Diagrama esquemático

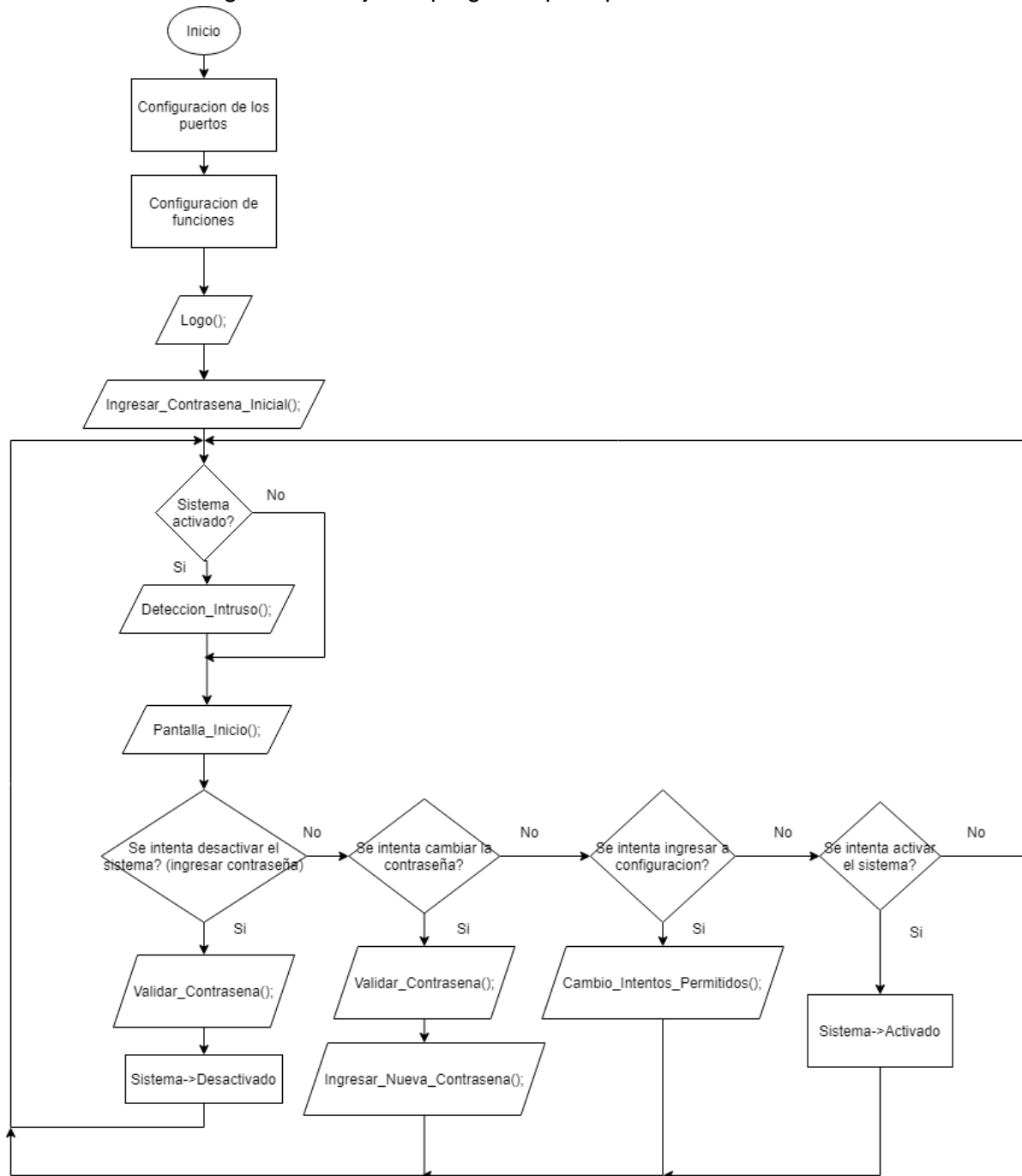


iii. Explicación del funcionamiento del hardware

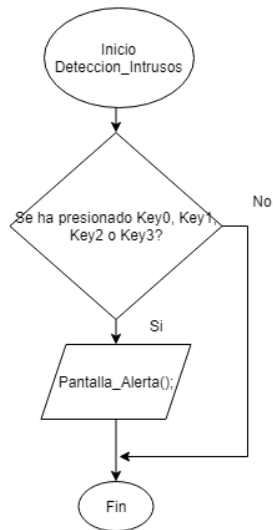
El sistema de seguridad es manipulado por medio de los sw1 y sw2, los tiene como función ingresar contraseña, así como configurar el sistema operativo. Para la facilidad de manipulación se usa una pantalla Nokia, el cual indica el proceso en el que se encuentra el sistema.

Software:

i. Diagrama de flujo del programa principal



ii. Diagrama de flujo de las funciones relevantes



iii. Explicación del algoritmo utilizado.

El algoritmo inicia con la configuración de los puertos y funciones a usar. Posterior, se ingresa a la función de pantalla logo, continua para consultar si es sistema está activado, detecta a los intrusos. Luego, ingresamos al menú principal, seguimos consultando qué opción elige, de no hacerlo solo nos mantendremos en pantalla de menú principal. cada opción tendrá su función correspondiente de acuerdo a la opción. Finalmente, regresará al proceso de detectar intrusos si el sistema está activado.

Resultados

Evidencia de resultados:

1.- Se inicia el programa con el logo del proyecto el cual es las iniciales de nuestros apellidos.

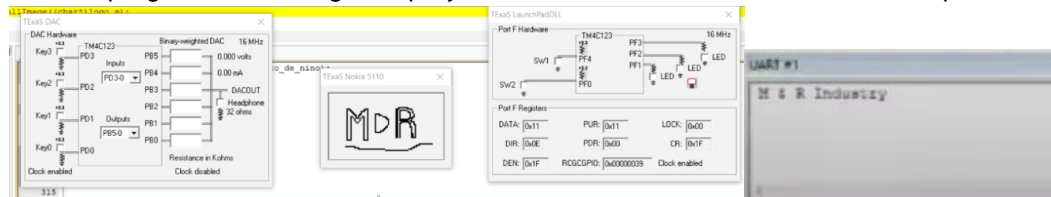


Imagen 1: Logo

2.- Posteriormente, se ingresa la contraseña inicial, con el SW1 se mueve el número al siguiente (Es decir, si está en 1 y se presiona SW1 se cambiará al 2 y así sucesivamente. En caso se llegue al 9 se reinicia el número a 1) con el SW2 se confirma el número en pantalla.

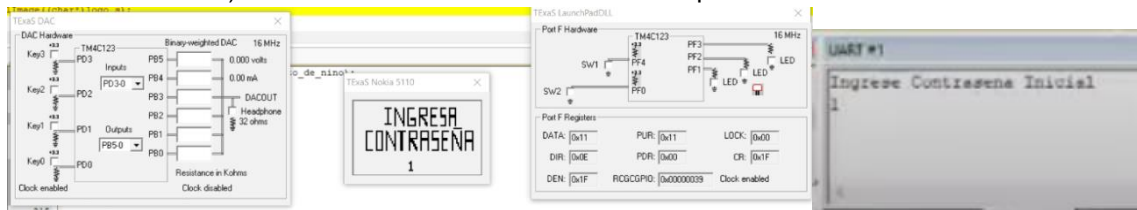


Imagen 2: Ingreso contraseña

3.- Luego de ingresar la contraseña inicial, se pasa a las instrucciones para el manejo de la pantalla principal.

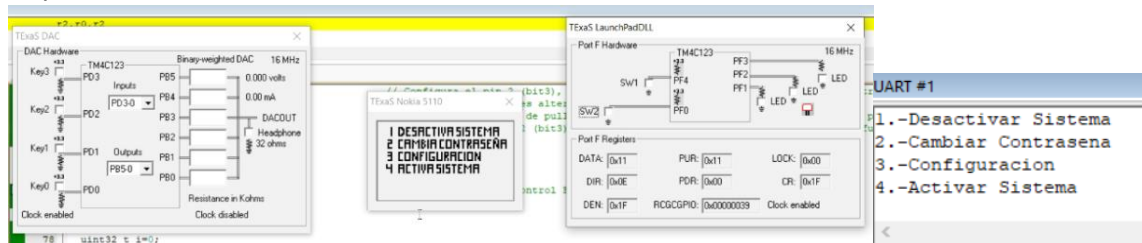


Imagen 3: indicaciones del menú principal

4.- Este es el menú principal el cual es manejado con el SW1 y SW2 donde con el primero se baja de opciones y con el segundo se confirma la opción.

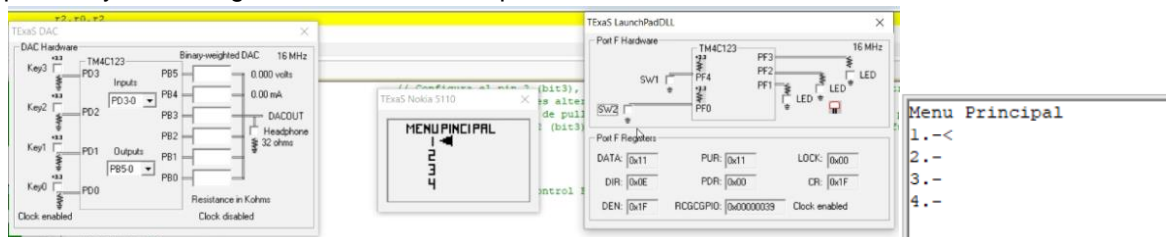


Imagen 4: Menú principal

5.- Acá se puede evidenciar otra vez el menú principal pero ya con el seleccionador en la opción 3.

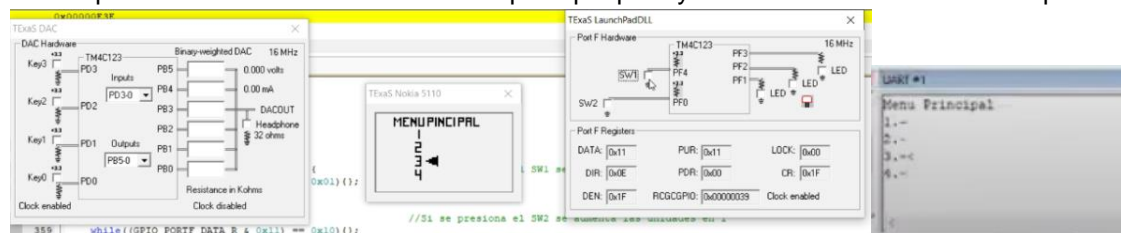


Imagen 5: Menú principal con seleccionador

6.- En el caso que el sistema ya esté activo, no se podrá acceder a la pantalla de configuración ni la de cambiar la contraseña por cuestiones de seguridad. Se evidenciará la pantalla de sistema activado en caso suceda.

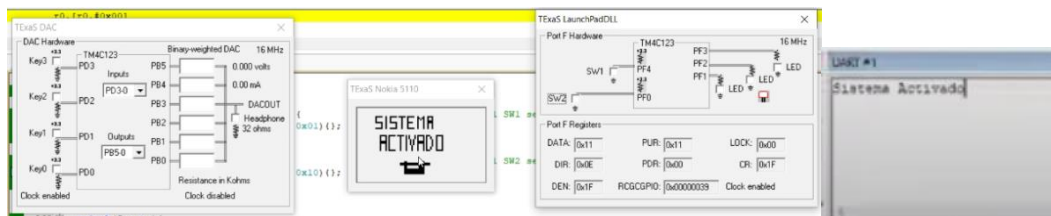


Imagen 6: Sistema activado

7.- Para la desactivación del sistema (opción 1 del menú), primero se debe ingresar la contraseña actual para comprobar que es acorde con la que se tiene en el sistema. En caso no sea acorde se bloqueará el sistema por 10s. En caso que se confirme, se procederá a desactivarlo.

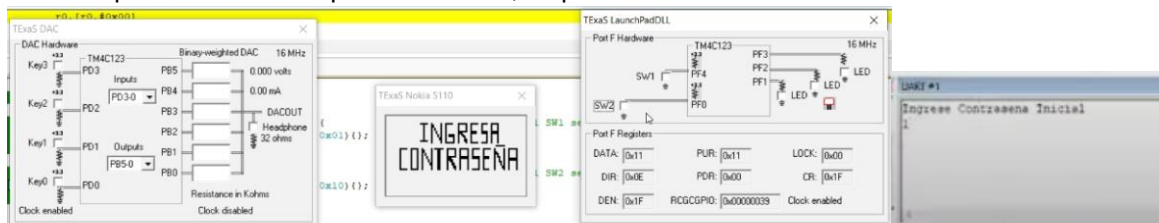


Imagen 7: Ingresar contraseña

8.- Sistema desactivado por correcta contraseña.

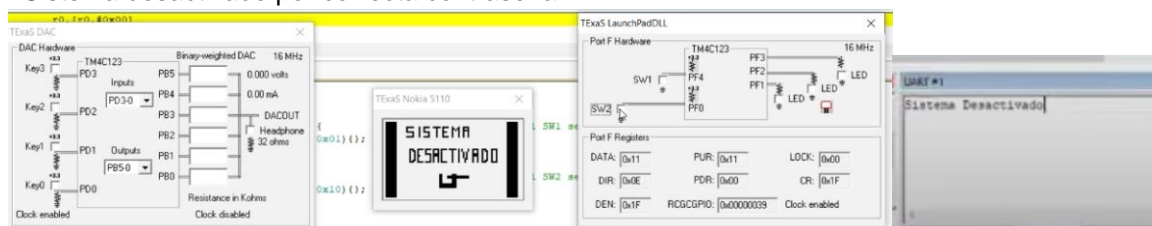


Imagen 8: Sistema desactivado

9.- En cambiar contraseña (opción 2 del menú), se podrá cambiar la ingresada por default que pide al inicio.

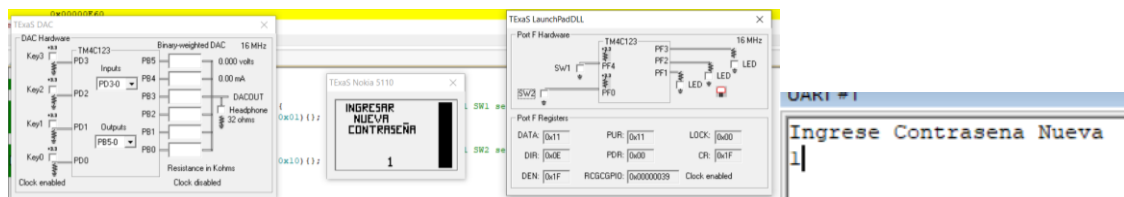


Imagen 9: Cambiar Contraseña

10.- En activar sistema (opción 4 del menú), una vez que el sistema ya esté desactivado, se podrá hacer uso de esta opción.

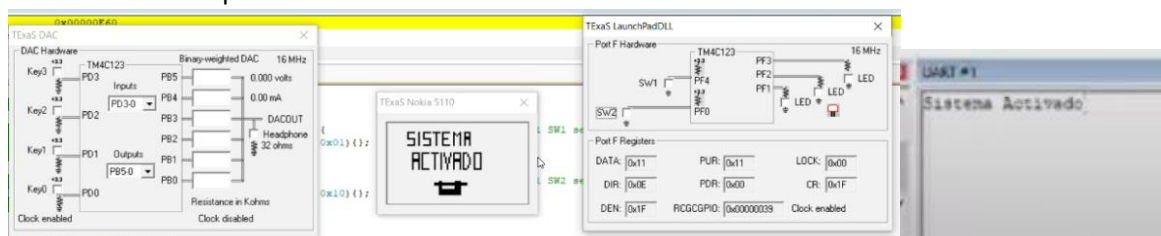


Imagen 10: Sistema activado

link de video explicativo: <https://youtu.be/IVuiugK8ZUo>

Conclusiones

El desarrollo del sistema fue culminado en su totalidad.

En conclusión, se logró desarrollar los objetivos previstos para la implementación del proyecto “sistema de seguridad”, los cuales fueron divididos en 7 tareas.

Las primeras cuatro tareas fueron desarrolladas de manera individual y las restantes de manera grupal, esto facilitó la rapidez en el desarrollo del proyecto.

Así mismo, las tareas desarrolladas grupalmente favorecieron el análisis de posibles casos de error en el proyecto.

Observaciones y Recomendaciones

-El sistema de seguridad es desarrollado de acuerdo al entorno donde se usará; Es decir, el sistema de seguridad solo se enfocó para ser instalada en una casa, por lo cual no tendrá un correcto funcionamiento si se usa en entorno.

-Recapitulando el desarrollo del proyecto logró implementar los objetivos planteados. Además, de permitir extender los conocimientos de los integrantes para futuros proyectos o entornos.

-Gracias a la experiencia obtenida por el desarrollo del proyecto, se recomienda crear una lista de pantallas, además de probar el sistema con todos los casos posibles.

-Monitoreo del profesor encargado del proyecto.

Bibliografía

Programa fuente:

```
/*****  
*****/  
/*****  
*****/  
/***** PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ *****/  
/***** PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ *****/  
/***** FACULTAD DE CIENCIAS E INGENIERÍA *****/  
/***** SISTEMAS DIGITALES *****/  
/*****  
*****/  
/***** Microcontrolador: TM4C123GH6PM *****/  
/***** EvalBoard: Tiva C Series TM4C123G LaunchPad *****/  
/***** Autor: Alessandro Romero Santos y Yhamil Favio Munive Calcina *****/  
/***** Fecha: 14/11/2021 *****/  
/***** *****/  
/*****  
*****/
```

```

#include <stdint.h>
#include "tm4c123gh6pm.h"
#include "Nokia5110.h"
#include "bitmapPhotos.h"
#include "TivaES.h"

uint32_t Sistema=0, Cursor=1, Seleccionar=0, Clave[10], Digitos=3, Intentos=3,
Comprobar[10], flag=0, flag1=0, flag2=0, flag3=0, Numero=1, flag4=0;
uint32_t Puerta_Principal=0, Ventana_Principal=0, Sala_Principal=0, Jardin=0,
Ventana_Cocina=0, Ventana_Cuarto=0, Cuarto_Invitado=0, Cuarto_Nino=0;

void config_portF(void)
{
    SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R5;           // activamos la señal de reloj
    del puerto F
    while( (SYSCTL_PRGPIO_R & SYSCTL_PRGPIO_R5)==0 ) {} // espera a que el puerto este
    listo para operar

    GPIO_PORTF_LOCK_R = 0x4C4F434B;
                    // Desbloquear puerto F
    GPIO_PORTF_CR_R |= 0x01;
                    // Se habilita PF0 (SW2)

    //SW1 está conectado al pin PF4, SW2 al pin PF0.

    GPIO_PORTF_DIR_R &= ~(0x11);
                    // Configura el pin 0 (bit0) y pin 4 (bit4) como entrada
    GPIO_PORTF_AFSEL_R &= ~(0x11);
                    // No utilizar funciones alternas
    GPIO_PORTF_PUR_R |= 0x11;
                    // Activa resistencias de pull-up en pin 0 (bit0) y pin
    4 (bit4)
    GPIO_PORTF_DEN_R |= 0x11;
                    // Habilita pines pin 0 y pin 4 para funcionar con
    niveles digitales.

    //Led Azul esta conectado al pin PF2, Led Rojo esta conectado al pin PF1, Led Verde
    esta conectado al pin PF3
    GPIO_PORTF_DIR_R |= 0x0E;
                    // Configura el pin 1, 2, 3 del puerto F como salida
    GPIO_PORTF_DR8R_R |= 0x0E;
                    // Activa el driver de 8 mA en el pin 1, 2, 3.
    GPIO_PORTF_DEN_R |= 0x0E;
                    // Habilita el pin 1, 2, 3; para funcionar con niveles
    digitales.
    GPIO_PORTF_DATA_R &= ~(0x0E);
                    // Apaga el led conectado a PF1, PF2, PF3 (Rojo, Azul, Verde)
} // fin config_portF

void config_portD(void){
    SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R3;           //
    Activamos el Reloj de Puerto D
    while( (SYSCTL_PRGPIO_R & SYSCTL_RCGCGPIO_R3)==0 ) { } // Se espera a que se active
    el reloj

    GPIO_PORTD_LOCK_R = 0x40007520;
                    // Desbloquear puerto D

```

```

GPIO_PORTD_CR_R |= 0x40007524;

// Se habilita PD3, PD2, PD1 y PD0

GPIO_PORTD_DIR_R &= ~(0x0E);
// Configura el pin 2 (bit3), pin 3 (bit4), pin 4
(bit5), pin 5 (bit6) como entrada
GPIO_PORTD_AFSEL_R &= ~(0x0E);
// No utilizar funciones alternas
GPIO_PORTD_PUR_R |= 0x0E;
// Activa resistencias de pull-up en pin 2 (bit3), pin 3
(bit4), pin 4 (bit5), pin 5 (bit6)
GPIO_PORTD_DEN_R |= 0x0E;
// Habilita pines pin 2 (bit3), pin 3 (bit4), pin
4 (bit5), pin 5 (bit6) para funcionar con niveles digitales.
}

void config_portE(void){
SYSCTL_RCGCGPIO_R |= SYSCTL_RCGCGPIO_R4;
Activamos el Reloj de Puerto D
while( (SYSCTL_PRGPIO_R & SYSCTL_RCGCGPIO_R4)==0) { } // Se espera a que se active
el reloj

GPIO_PORTE_LOCK_R = 0x40024520;
// Desbloquear puerto D
GPIO_PORTE_CR_R |= 0x40024524;

// Se habilita PD3, PD2, PD1 y PD0

GPIO_PORTE_DIR_R &= ~(0x0E);
// Configura el pin 2 (bit3), pin 3 (bit4), pin 4
(bit5), pin 5 (bit6) como entrada
GPIO_PORTE_AFSEL_R &= ~(0x0E);
// No utilizar funciones alternas
GPIO_PORTE_PUR_R |= 0x0E;
// Activa resistencias de pull-up en pin 2 (bit3), pin 3
(bit4), pin 4 (bit5), pin 5 (bit6)
GPIO_PORTE_DEN_R |= 0x0E;
// Habilita pines pin 2 (bit3), pin 3 (bit4), pin
4 (bit5), pin 5 (bit6) para funcionar con niveles digitales.
}

void ConfiguraUART0(void){
// Habilitamos reloj para el UART0
SYSCTL_RCGC1_R |= SYSCTL_RCGC1_UART0;
// Habilitamos reloj para GPIOA
SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOA;
// Inhabilitamos el UART0
UART0_CTL_R &= ~UART_CTL_UARTEN;
// Velocidad 9600bps (Fsysclk = 16MHz)
UART0_IBRD_R = (UART0_IBRD_R & 0xFFFF0000) | 104;
UART0_FBRD_R = (UART0_FBRD_R & 0xFFFFF000) | 11;
// 8, N, 1, FIFOs habilitados
UART0_LCRH_R = (UART0_LCRH_R & 0xFFFFF000) | 0x70;
// Habilitamos el UART0
UART0_CTL_R |= UART_CTL_UARTEN;
// Desactivamos modo analógico en PA0 y PA1
GPIO_PORTA_AMSEL_R &= ~(0x03);
// Conectamos UART0 a PA0 y PA1
GPIO_PORTA_PCTL_R = (GPIO_PORTA_PCTL_R & 0xFFFFF000) | 0x00000011;

```

```

// Activamos funciones alternas en PA0 y PA1
GPIO_PORTA_AFSEL_R |= 0x03;
// Activamos funciones digitales en PA0 y PA1
GPIO_PORTA_DEN_R |= 0x03;
}

void TxCar( uint8_t Car )
{
    while((UART0_FR_R & UART_FR_TXFF) != 0 ) ; //Espera mientras pila llena
    UART0_DR_R = Car;
}

uint8_t RxCar(void)
{
    //while((UART0_FR_R & UART_FR_RXFE) != 0) ; //Espera mientras pila vacía
    //return UART0_DR_R & 0xFF;

    if((UART0_FR_R & UART_FR_RXFE) == 0){
        return UART0_DR_R & 0xFF;
    }else{
        return 0;
    }
}

void TxCadena( uint8_t Cadena[] )
{
    uint8_t q;
    for( q = 0; Cadena[q] != '\0'; q++ )
        TxCar( Cadena[q] );
}

void Retardo(uint32_t NumCiclos){
    uint32_t i;
    for(i=0;i<=NumCiclos*2000000;i++) {}
    // Lazo de control FOR para generar un
    retardo.
}

void Ingresar_Contrasena_Inicial(void){
    uint32_t i=0;

    Nokia5110_SetCursor(5, 5);
    Nokia5110_OutChar(48+Numero);
    while(i<=Digitos){
        do{
            TxCar(0x1B);
            TxCadena("[");
            TxCadena("2");
            TxCadena("J");
            TxCadena("\r");
            TxCadena("Ingrese Contraseña Inicial");
            if((GPIO_PORTF_DATA_R & 0x10) == 0x00){

                while((GPIO_PORTF_DATA_R & 0x11) == 0x01){};
                Numero++;
            }
            if(RxCar()==119 | RxCar()==87){
                while(RxCar()==119 | RxCar()==87){};
                Numero++;
            }
        }while(1);
        i++;
    }
}

```

```

    }
    if(RxCar()== 83| RxCar()== 115){
        while(RxCar()== 83| RxCar()== 115){};
        Numero--;
    }
    if(Numero>9){
        Numero=1;
    }
    if(Numero<1){
        Numero=9;
    }
    Nokia5110_SetCursor(5, 5);
    Nokia5110_OutChar(48+Numero);
    TxCadena("\n\r");
    TxCar(Numero+48);
    Retardo(2);
}while((RxCar()!=13) && ((GPIO_PORTF_DATA_R&0x01)!=0x00));
while(RxCar()==13){};
while((GPIO_PORTF_DATA_R & 0x11) == 0x10){};
flag4=0;
TxCadena("$");
Nokia5110_SetCursor(3, 5);
Nokia5110_OutString("$");
Retardo(2);
Nokia5110_SetCursor(3, 5);
Nokia5110_OutString(" ");
Clave[i]=Numero;
i++;
}
}

uint32_t Validar_Contrasena(){
    uint32_t i=0,j=0,k=0;

    while(k<Intentos & flag1==0){
        flag2=0;
        Numero=1;
        while(i<=Digitos){
            Nokia5110_SetCursor(5, 5);
            Nokia5110_OutChar(48+Numero);
            do{
                TxCar(0x1B);
                TxCadena("[");
                TxCadena("2");
                TxCadena("J");
                TxCadena("\r");
                TxCadena("Ingrese Contraseña Inicial");
                if((GPIO_PORTF_DATA_R&0x10) == 0x00){

                    while((GPIO_PORTF_DATA_R & 0x11) == 0x01){};
                    Numero++;
                }
                if(RxCar()==119 | RxCar()==87){
                    while(RxCar()==119 | RxCar()==87){};
                    Numero++;
                }
            }
            if(RxCar()== 83| RxCar()== 115){
                while(RxCar()== 83 | RxCar()== 115){};

```

```

        Numero--;
    }
    if (Numero>9) {
        Numero=1;
    }
    if (Numero<1) {
        Numero=9;
    }
    Nokia5110_SetCursor(5, 5);
    Nokia5110_OutChar(48+Numero);
    TxCadena("\n\r");
    TxCar(Numero+48);
    Retardo(2);
}while((RxCar()!=13) && ((GPIO_PORTF_DATA_R&0x01)!=0x00));
while((GPIO_PORTF_DATA_R & 0x11) == 0x10){};
Nokia5110_SetCursor(3, 5);
Nokia5110_OutString("$");
TxCadena("$");
Retardo(2);
Nokia5110_SetCursor(3, 5);
Nokia5110_OutString(" ");
Comprobar[i]=Numero;
i++;

}

for(j=0; j<=Digitos;j++){
    if(Comprobar[j]!=Clave[j]){
        Nokia5110_DrawFullImage((char*)contrasena_incorrecta);
        Retardo(2);
        Nokia5110_DrawFullImage((char*)ingre_contra_d_sis);
        i=0;
        k++;
        flag2=1;
        break;
    }
}
if(flag2==0){
    flag1=1;
}
}
if(k==Intentos){
    flag1=0;
    return 0;
}else{
    flag1=0;
    return 1;
}
}

void Ingresar_Nueva_Contrasena(void){
    uint32_t i=0;

```


ARDUINO NOKIA 5110 TUTORIAL #2- DISPLAYING CUSTOMIZED GRAPHICS.

Consulta: 3 de noviembre.

<https://www.electronics-lab.com/project/arduino-nokia-5110-tutorial-2-displaying-customized-graphics/>

Instructables circuits

How to Use NOKIA 5110 LCD Screen With Arduino.

Consulta: 5 de noviembre.

<https://www.instructables.com/Getting-Started-With-NOKIA-5110-LCD-Screen-Using-A/>

Jonathan W. Valvano

Embedded Systems - Shape The World.

Consulta: 5 de noviembre.

<http://users.ece.utexas.edu/~valvano/Volume1/E-Book/>