

Sistema de alarma doméstica

Gino Bartolucci
Agosto 2018.
E.E.T.P. 483 “El industrial”.
Electrónica.
Proyecto y diseño

Abstract

En este documento se explica solución a un problema de una alarma para uso doméstico utilizando el microcontrolador atmega328 además de su programación, circuitos esquemáticos, circuitos impresos (Pcb), planos del gabinete para el sistema realizado.

| | |
|--|----|
| Capítulo 1 Introducción e información general..... | 1 |
| Explicación problema..... | 1 |
| Solución..... | 2 |
| Funcionamiento | 2 |
| Capítulo 2 Diseño | 3 |
| Código..... | 3 |
| Circuito eléctrico. | 4 |
| Pcb de los circuitos eléctricos | 5 |
| Gabinete..... | 6 |
| Capítulo 3 Gestión | 7 |
| Costos | 7 |
| Capítulo 4 Planificación y ejecución..... | 8 |
| Planificación y Ejecución..... | 8 |
| Capítulo 5 Evaluación y perfeccionamiento..... | 9 |
| Conclusión..... | 9 |
| Evaluación..... | 9 |
| Perfeccionamiento..... | 9 |
| Capítulo 6 Anexo..... | 10 |
| Bibliografía..... | 10 |
| Datasheet..... | 10 |
| Anexo..... | 11 |

Capítulo 1

Introducción e información general

Explicación del problema:

Un sistema de alarma es un elemento de seguridad pasiva. Esto significa que no evitan una situación anormal, pero sí son capaces de advertir de ella, cumpliendo así, una función disuasoria frente a posibles problemas.

Se necesitan tener controlados cuatro sectores de la vivienda independientes entre ellos para que mediante sensores magnéticos o de movimiento de tipo NC (normal cerrado) detectar el allanamiento de la vivienda y dar aviso para activar una bocina o algún otro instrumento sonoro.

El sistema de alarma tendrá 3 estados:

Estado Inactivo: La alarma no se activa si los sensores detectan un intruso en algunas de las zonas que se encuentran activas.

Estado Alerta: El sistema dará señal a la bocina si los sensores de las zonas activas detectan algún intruso en la vivienda, si esto último sucede se pasará al estado Activo.

Estado Activo: Este estado se activará cuando se el sistema de alarma mediante los sensores de las zonas activas hayan detectado algún intruso y se solicitará la contraseña para que el sistema pase al estado inactivo.

El sistema de alarma se activará o desactiva al introducir una contraseña de 6 dígitos mediante un teclado numérico, al introducir la contraseña de forma errónea 3 veces, el sistema se bloqueara y se solicitara un Puk (Personal Unlocking Code) de 6 digitos para desbloquear el sistema, el código puk no tiene límite de intentos.

En el caso de querer activar o desactivar zonas en específico se hará mediante el teclado numérico, esto solo se podrá realizar en el momento en que el sistema de alarma esté en estado Inactivo.

El sistema consta con conexiones de entrada, para los distintos tipos de sensores, y para la conexión de un sistema de control remoto, que servirá como reemplazo a la introducción de contraseña o pin mediante el teclado numérico. También consta con conexiones de salida, para activar dispositivos que se ocupan de hacer sonar la bocina.

Solución:

Se utilizará un microcontrolador atmega328 para la solución del problema. El microcontrolador se programa mediante la placa Arduino Uno y el IDE (entorno de desarrollo) de Arduino. los circuitos impresos se diseñarán con el programa Eagle.

Se diseñará un circuito impreso principal para la contención del atmega328 con sus respectivos pines para la conexión con el segundo PCB y pines para la conexión de teclado numérico y display. El PCB principal también constará una fuente de 5 voltios la cual se hará con un regulador de tensión 7805.

Se creará otro PCB independiente y conectado mediante cables al principal para las distintas conexiones de entradas y salidas para las zonas, bocinas, buzzer para indicar las pulsaciones del teclado y control remoto.

Se le colocará un display al gabinete del sistema de alarma para tener una comunicación visual con el usuario y indicar el estado actual de la alarma, la introducción de contraseña o puk y el estado activo o inactivo de las zonas.

El gabinete a utilizar tendrá un tamaño de 10x15x5cm y un grado de protección IP56. Estará sellado mediante 4 tornillos en cada una de sus esquinas frontales. En él se colocarán todas las partes del sistema de alarma, el teclado numérico y el display estarán a la vista del usuario. Por dentro se colocarán todos los demás elementos como los PCB. El gabinete consta con 2 orificios para la salida de cables como por ejemplo para la conexión de alimentación, la conexión de las zonas.

Funcionamiento:

El estado inicial de la alarma será inactivo en el cual presionando las teclas función del teclado numérico se podrán encender o apagar las zonas, el estado de las zonas y el estado del sistema se mostrará por el display, al introducir la contraseña se pasará al estado Alerta. En este estado el sistema pasa a monitorear las zonas que fueron activadas y en el caso de que los sensores detecten algún intruso se abrirá el circuito (0), si no se detecta ningún intruso el circuito permanece cerrado (1). Al detectar un intruso y abrirse el circuito de cualquiera de las zonas activas el estado del sistema pasa a Activo y se accionará la bocina, se solicitará la introducción de la contraseña para pasar al estado inicial Inactivo.

En cualquier caso de introducir la contraseña mal 3 veces se solicitará un código puk de la misma longitud (6 caracteres) que la contraseña. En caso de conectar un control remoto este reemplaza al puk y la contraseña. El número de veces que se introdujo mal la contraseña y la cantidad de caracteres que se ingresaron se mostrará en el display. Al introducir la contraseña o puk esta se debe confirmar presionando * en el teclado, si por algún motivo la contraseña o el puk se introdujo mal se puede borrar para introducirla de vuelta, presionando # en el teclado, esto antes de presionar * y confirmar.

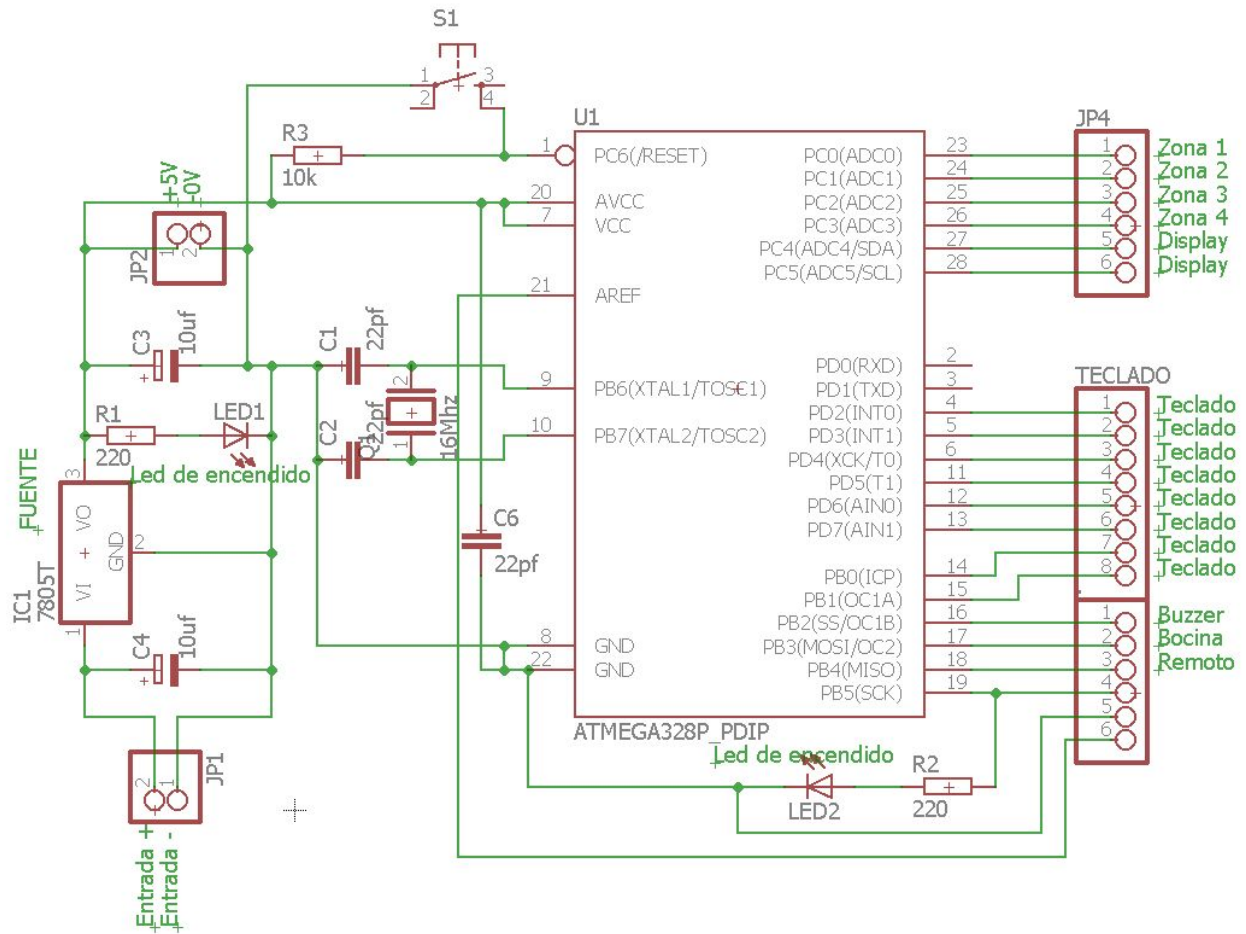
Capítulo 2 Diseño

Código utilizado:

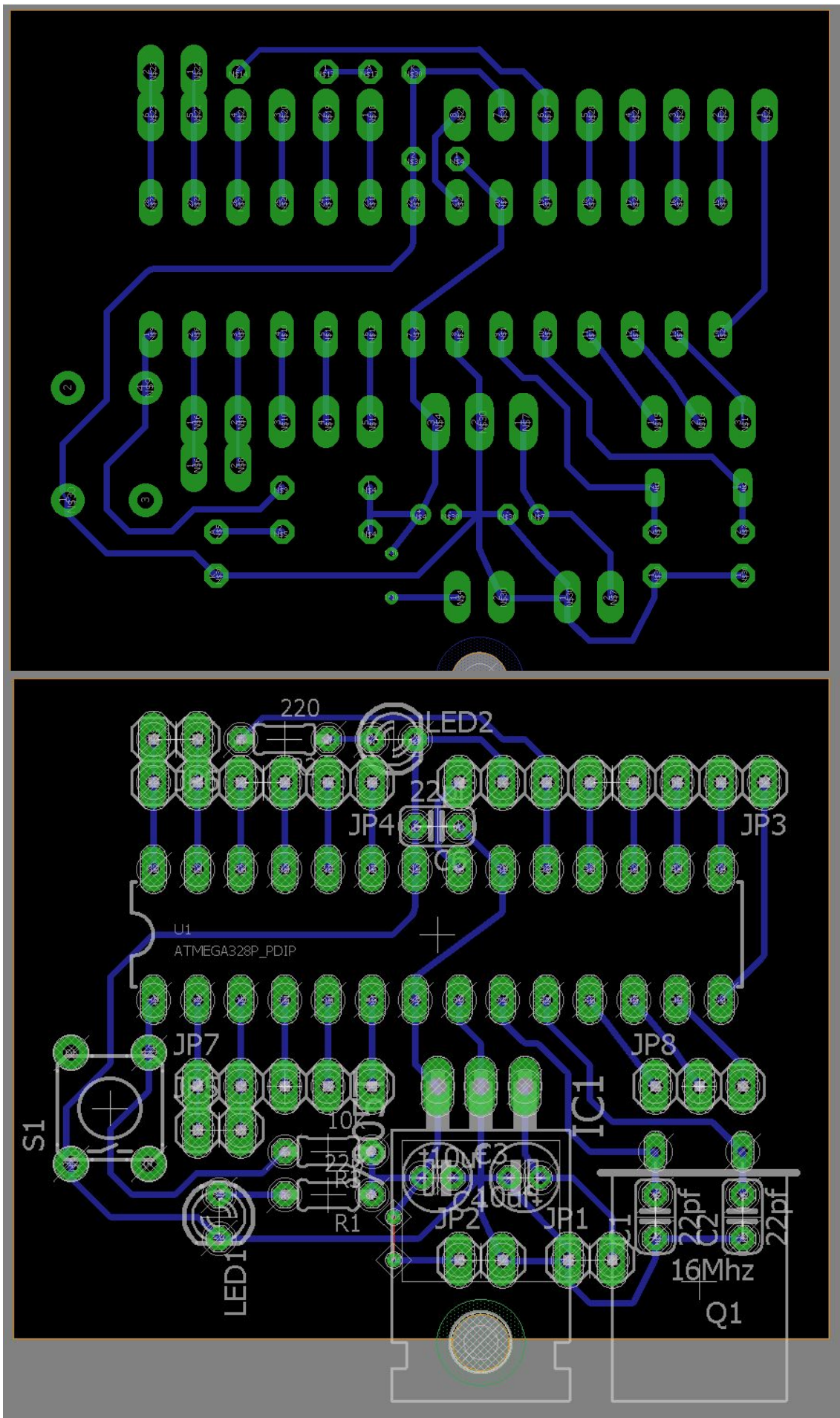
Para la programación del microcontrolador se utilizó la placa Arduino uno y el Ide Arduino. Se utilizaron las librerías Password.h para la creación de la contraseña y el código puk, Keypad.h para la utilización del teclado numérico, Wire.h LiquidCrystal_I2C.h para la utilización del display de 16x2 segmentos.

Puede encontrar el código completo y demás archivos en el siguiente link:
https://github.com/GinoBartolucci/sistema_alarma_arduino

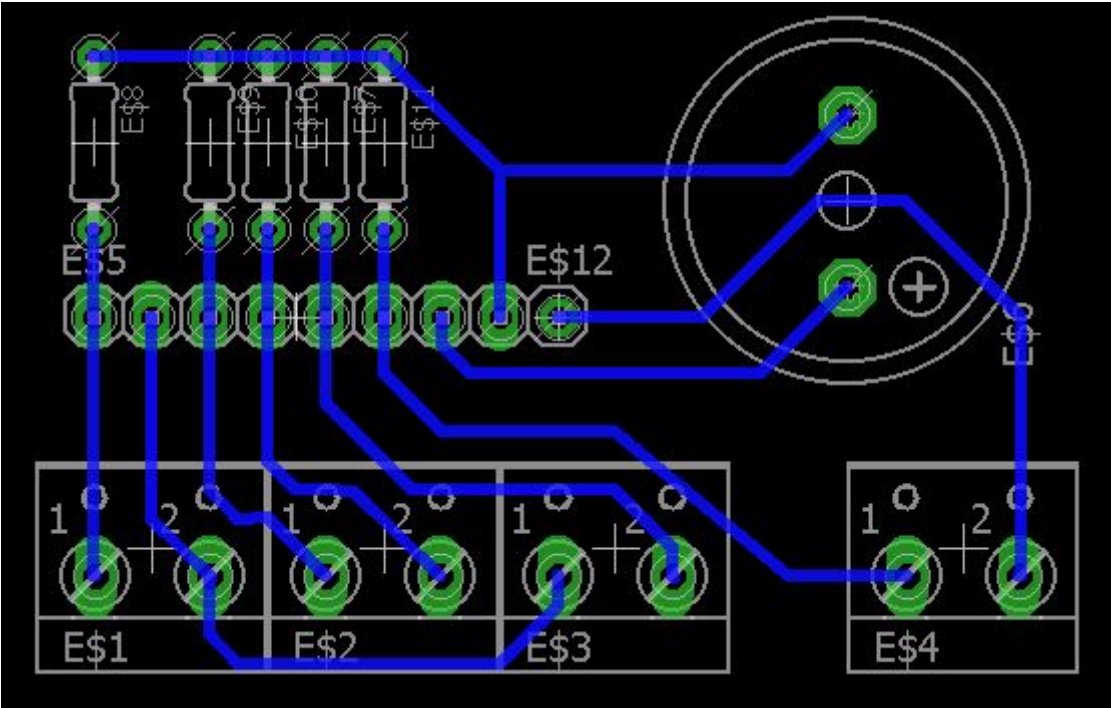
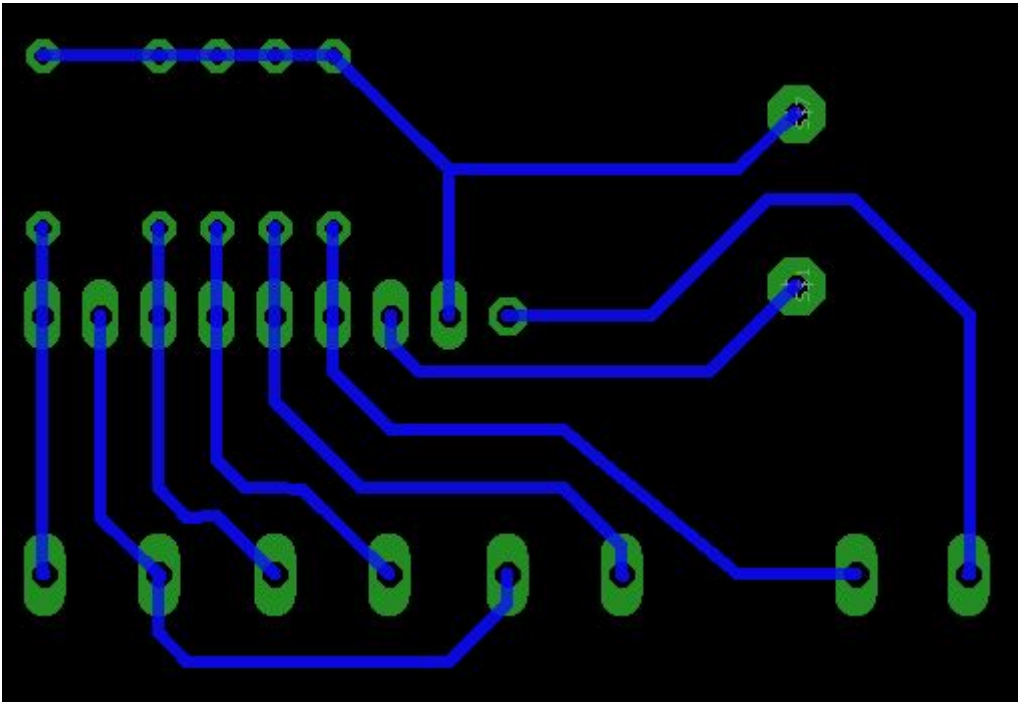
Circuito eléctrico:



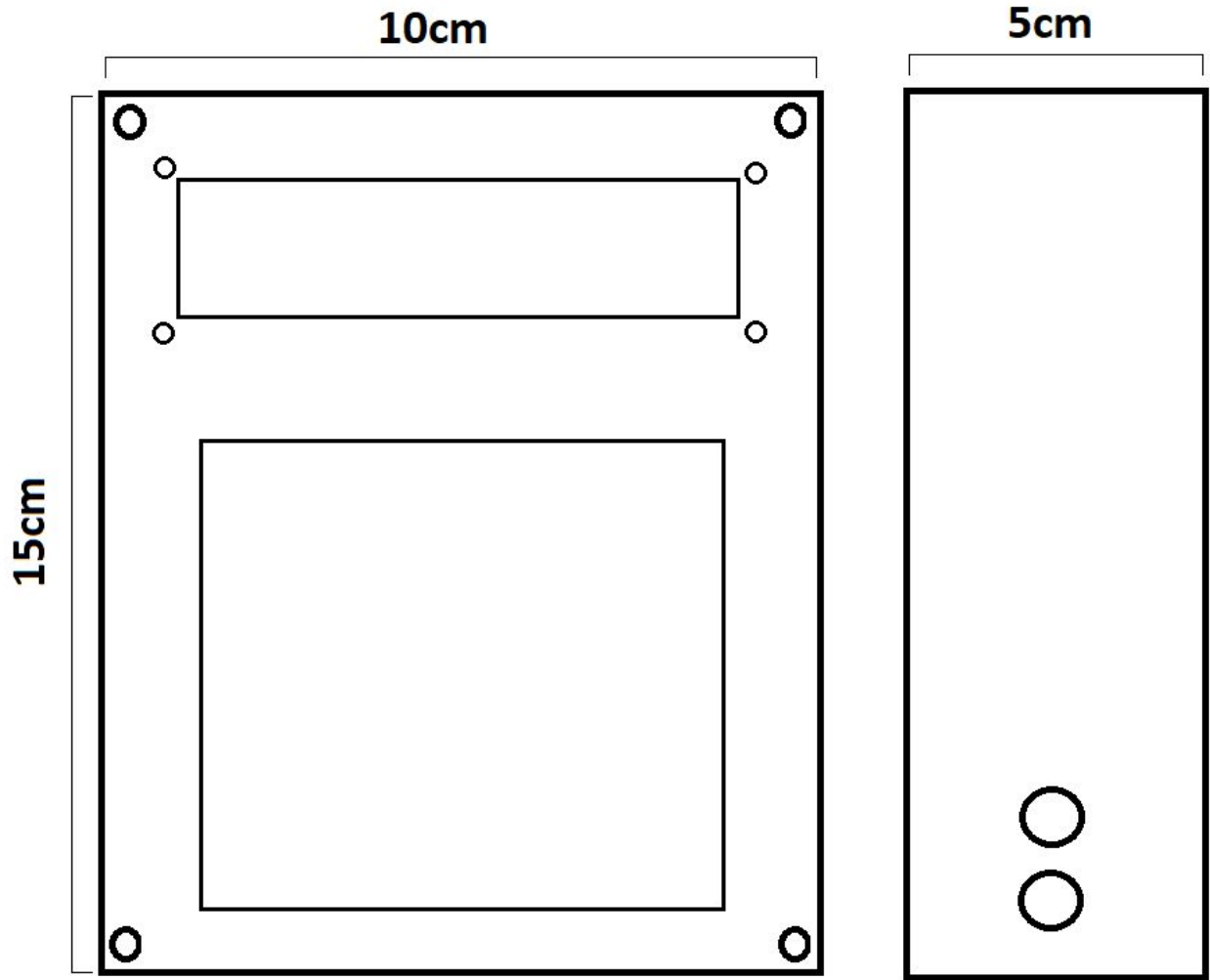
Circuitos impresos: PCB principal.



PCB secundario.



Gabinete:



Capítulo 3 Gestión

Costos:

| Producto | Cantidad | Costo (\$ pesos) |
|--------------------------------|-----------|------------------|
| Atmega328 | 1 | 90 |
| Cristal 16Mhz | 1 | 15 |
| Regulador de tensión 7805 | 1 | 25 |
| Capacitor electrolítico 10uf | 2 | 10 |
| Capacitor cerámico 22nf | 2 | 6 |
| Tira pines | 2 | 14 |
| Caja estanca (Gabinete) | 1 | 110 |
| Led rojo | 2 | 8 |
| Resistencias 1/4 | 3 | 3 |
| Display arduino 16x2 + i2c | 1 | 130 |
| Teclado matricial membrana 4x4 | 1 | 45 |
| Cargador | 1 | 0 |
| Cables varios | 18 | 0 |
| Borneras 1x2 | 4 | 16 |
| Buzzer | 1 | 15 |
| Total | 41 | 487 |

Capítulo 4

Planificación y ejecución

Planificación y ejecución:

- Evaluación de la problemática y solución
- Estudiar posibles herramientas para la solución
- Elegir el método y herramientas para la solución (Arduino)
- Investigación de posible código
- Utilizar librería keypad.h y realizar pruebas con el teclado numérico
- Añadir al código librería Password.h y realizar pruebas ingresando un pin por teclado
- Mejorar código anterior añadiendo número de intentos y puk
- Añadir el display al proyecto para poder mostrar caracteres de la contraseña
- Añadir al código las zonas del sistema, pudiendo activarlas o desactivarlas y mostrarlo en el display.
- Añadir la posibilidad de conectar un control remoto al sistema
- Prueba del sistema completo en protoboard
- Creación del circuito impreso principal para el microcontrolador en Eagle.
- Soldado de componentes al circuito impreso principal
- Prueba del sistema con código PCB principal y protoboard
- Adaptación de gabinete para el proyecto
- Creación de segundo PCB para las entradas y salidas
- Realización del informe

Capítulo 5

Evaluación y perfeccionamiento

Conclusión: Para la solución al problema aplique conocimientos en arduino que conocía y también algunos nuevos, el código fue hecho completamente por mí guiándome por los tutoriales de las librerías. Hice cosas nuevas como un pcb para el atmega328 con el que tuve problemas al principio y la utilización de una caja estanca como gabinete. También aprendí a usar la plataforma github.

Evaluación El proyecto terminado cumple dando una solución con la problemática. Se podría mejorar en el apartado de seguridad añadiendo elementos como una batería para cuando se corte el suministro eléctrico y alguna forma de evitar que el atmega328 se pueda reprogramar

Perfeccionamiento: Durante la realización del proyecto tuve varios problemas principalmente con el PCB principal y secundario en el que las referencias a GND están mal.

Capítulo 5
Anexo

Bibliografía:

Librerías arduino:

Display: <https://playground.arduino.cc/Main/LCDI2C>

Teclado: <https://playground.arduino.cc/Code/Keypad>

Contraseña: <https://playground.arduino.cc/Code/Password>

https://naylampmechatronics.com/blog/35_Tutorial--LCD-con-I2C-control-un-LCD-con-so.html

<https://www.taringa.net/posts/hazlo-tu-mismo/19566035/Construi-una-alarma-con-Arduino.html>

Datasheet:

Display: <http://www.cypress.com/file/127746/download>

Atmega328p:

http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

Teclado:

<https://www.parallax.com/sites/default/files/downloads/27899-4x4-Matrix-Membrane-Keypad-v1.2.pdf>

Oscilador 16Mhz: <https://www.nxp.com/docs/en/application-note/AN2500.pdf>

Buzzer: <http://www.farnell.com/datasheets/2171929.pdf>