

## Funciones necesarias/Subprogramas:

Para el ultrasonidos será necesaria una función para medir la distancia a los objetos. La función devolverá la distancia y recibirá los pines del Trigger, que manda el pulso de ultrasonidos, y del Echo, que los recibe:

```
int distancia (int TriggerPin, int EchoPin)
{
    long tiempo, distancia_cm;
    digitalWrite (TriggerPin, LOW);
    delayMicroseconds (4);
    digitalWrite (TriggerPin, HIGH);
    delayMicroseconds (10); //El Trigger envía una señal durante 10 microsegundos.
    digitalWrite (TriggerPin, LOW);

    tiempo = pulseIn (EchoPin HIGH); //Mide el tiempo que tarda el EchoPin en pasar
                                     de LOW a HIGH.
    distancia_cm = 343*tiempo*100; //Se calcula la distancia a partir del tiempo y la
                                   velocidad del sonido y se pasa a centímetros.
    return (distancia_cm);
}
```

Para los leds se pueden hacer funciones que los enciendan y los apaguen:

```
void encender (int LedPin)
{
    digitalWrite (LedPin, HIGH);
}

void apagar (int LedPin)
{
    digitalWrite (LedPin, LOW);
}
```

Para el teclado es necesaria una función que lea las letras que se presionan. Se utiliza la función `teclado.getKey()` que pertenece a la librería `Keypad.h`.

Para la pantalla LCD utilizaremos una función para escribir en la pantalla lo que se haya introducido por teclado y otra para borrar la pantalla cuando se introduzca una contraseña incorrecta o se pulse un botón determinado:

-Función para escribir en la pantalla:

La función tendrá como entrada una cadena de caracteres en la que se habrían guardado las teclas presionadas en el teclado por el usuario.

```
void escribir_en_lcd (char cadena[ ])
{
    int posicion_x=0, posicion_y=0; //Se añadiría posicion_y si la pantalla tuviera 2 filas.
    int i;

    for (i=0;i<N;i++) //N es el número de caracteres que tiene la contraseña.
    {
        lcd.setCursor (posicion_x, posicion_y); //Se coloca el cursor en el (0,0).
        lcd.print (cadena[i]); //Se imprime el caracter en la pantalla.
        posicion_x++; //Se aumenta la posición de las x para continuar escribiendo.
    }
}
```

-Función para borrar la pantalla:

```
void borrar_pantalla ()
{
    int i, posicion_x=0;
    for (i=0;i<N;i++)
    {
        lcd.print(' '); //Imprime un espacio en blanco en la posición en la que se
                          encuentre.
        posicion_x++;
    }
}
```

Finalmente, usaremos una función para comparar el código correcto con el introducido por el usuario. Esta función devolverá 1 o 0 en función de si la clave es correcta o no:

```
int comprobacion_clave (char cadena[ ])
{
    int resultado=1, i;
    char codigo[ ] = {}; //Se define el código correcto.

    for (i=0;i<N;i++)
    {
        if (cadena[i]!=codigo[i])
            resultado=0;
    }
    return (resultado);
}
```