This custom function will print a message that the alarm is activated and that we need to enter a password in order to stop the alarm. So using the next while loop we are constantly checking whether we have pressed a button on the keypad, and each button press is added to the tempPassword variable. If we enter more than 4 digits or press the sharp button the previously entered digits will be cleared so we can type them again from begin.

```
void enterPassword() {
int k=5;
tempPassword = "";
activated = true;
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" *** ALARM *** ");
lcd.setCursor(0,1);
lcd.print("Pass>");
while(activated) {
keypressed = myKeypad.getKey();
if (keypressed != NO_KEY){
if (keypressed == '0' || keypressed == '1' || keypressed == '2' || keypressed == '3' ||
keypressed == '4' || keypressed == '5' || keypressed == '6' || keypressed == '7' ||
keypressed == '8' || keypressed == '9' ) {
tempPassword += keypressed;
lcd.setCursor(k,1);
lcd.print("*");
k++;
}
}
if (k > 9 || keypressed == '#') {
tempPassword = "";
k=5;
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" *** ALARM *** ");
lcd.setCursor(0,1);
lcd.print("Pass>");
}
if ( keypressed == '*') {
if ( tempPassword == password ) {
activated = false;
alarmActivated = false;
noTone(buzzer);
screenOffMsg = 0;
}
else if (tempPassword != password) {
lcd.setCursor(0,1);
lcd.print("Wrong! Try Again");
delay(2000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" *** ALARM *** ");
lcd.setCursor(0,1);
lcd.print("Pass>");
```

```
47. }
48. }
49. }
50. }
```