

מדריך פרויקט מקיף: ערכת חממה חכמה והשקיה אוטומטית (IoT) עם ESP32

1. סקירת הפרויקט ומטרותיו

פרויקט זה מציע פתרון מקצה לקצה לבניית מערכת בקרת חממה והשקיה אוטומטית המבוססת על בקר ESP32, המשלבת ניטור אקלים סביבתי ושליטה מרחוק באמצעות ממשק רשת (Web Dashboard).

מטרת הפרויקט:

- ניטור בזמן אמת של טמפרטורה ולחות בחממה למניעת עומסי חום.
- הפעלת משאבת השקיה או שסתום חשמלי באמצעות ממסר לפי תזמון או ספי טמפרטורה.
- הצגת הנתונים ומצב המערכת על גבי מסך תצוגה LCD 16x2 בלחיצת כפתור או באופן רציף.
- אפשרות להפעלה ידנית של ההשקיה באמצעות כפתור פיזי לחוק עוקף (Manual Override).
- שרת אינטרנט פנימי (Web Server) המשודר על גבי ה-Wi-Fi של ה-ESP32 לצפייה בנתונים ושליטה מרחוק מכל מכשיר נייד או מחשב.

2. רשימת רכיבי הערכה ומפרט טכני

מק"ט	רכיב	תיאור ומפרט טכני	תפקיד במערכת
EL3011	ESP32 Development Board	מיקרו-בקר עוצמתי בעל מעבד כפול ליבה, תקשורת Wi-Fi 802.11 b/g/n ו-Bluetooth משולבים.	מוח המערכת – מעבד נתונים, מארח את השרת המקומי (Web Server), ומנהל את החיישנים והממסר.
EL3013	חיישן טמפרטורה ולחות DHT22	חיישן דיגיטלי בעל דיוק גבוה (טווח טמפרטורה -40°C עד 80°C, טווח לחות 0-100%).	דגימת התנאים הסביבתיים בחממה.

מק"ט	רכיב	תיאור ומפרט טכני	תפקיד במערכת
EL3029	מסך תצוגה LCD 16x2 עם מתאם I2C	מסך אלפא-נומרי (16 תווים ב-2 שורות) כולל מתאם I2C (PCF8574)) לחיבור ב-2 קווי תקשורת.	הצגת נתוני טמפרטורה, לחות וסטטוס השקיה במקום.
EL4023	מודול ממסר כפול (Channel Relay-2)	מודול מיתוג מתח עד 250VAC / 10A עם הפרדה גלוונית (Optocoupler).	הפעלת משאבת מים / שסתום השקיה (ערוץ 1) ומאוורר צינון (ערוץ 2).
EL4014	מודול כפתור רגעי (Keyestudio Button)	כפתור לחיצה בעל נגד משוך מובנה לקבלת אות דיגיטלי נקי.	לחצן עוקף (Override) להפעלה/כיבוי ידני מיידי של ההשקיה.
EL4003	ערכת חוטי גישור Dupont 20cm	חוטים גמישים בחיבורי זכר/נקבה שונים.	חיבור פיזי וחיווט בין ה-ESP32 לכל רכיבי הערכה.

3. תרשים חיבורים ומפת פינים (Wiring Diagram)

יש לבצע את החיווט לפי הפינים המפורטים בטבלה הבאה. שים לב לחיבורי המתח (5V / 3.3V) וההארקה (GND).

רכיב	פין ברכיב	חיבור בלוח ESP32 (EL3011))	הערות
חיישן DHT22 (EL3013))	+ / VCC	3.3V	מתח הפעלה לחיישן
	- / GND	GND	הארקה משותפת

רכיב	פין ברכיב	חיבור בלוח ESP32 ((EL3011	הערות
	OUT / DATA / S	GPIO 4	קו נתונים דיגיטלי
מסך I2C LCD ((EL3029	GND	GND	הארקה משותפת
	VCC	(5V (VIN	אספקת מתח לתאורה האחורית והמסך
	SDA	GPIO 21	קו נתונים I2C תקני ב-ESP32
	SCL	GPIO 22	קו שעון I2C תקני ב-ESP32
	VCC	(5V (VIN	מתח הפעלה לסלילי הממסר
מודול ממסר (EL4023)	GND	GND	הארקה משותפת
	IN1	GPIO 18	פין פיקוד משאבת השקיה (ערוץ 1)
	IN2	GPIO 19	פין פיקוד מאוורר / קירור (ערוץ 2)
מודול כפתור (EL4014)	V / VCC	3.3V	מתח הפעלה לכפתור
	G / GND	GND	הארקה משותפת
	S / OUT	GPIO 13	אות לחיצה דיגיטלי

רכיב	פין ברכיב	חיבור בלוח ESP32 ((EL3011	הערות
			(לחצן השקיה ידנית)

דגש בטיחות: בעת חיבור משאבת מים או שסתום הפועלים במתח רשת גבוה (220V), יש לנקוט באמצעי זהירות חמורים ולהסתייע בחשמלאי מוסמך. מומלץ להשתמש במורשי מתח נמוך (כגון משאבת 12V DC) עבור ניסויים והדגמות.

4. הגדרת סביבת העבודה (Software Setup)

1. פתחו את תוכנת **Arduino IDE**.
2. ודאו כי חבילת הלוחות של **ESP32** מותקנת ב-Boards Manager.
3. התקינו את הספריות הבאות דרך מנהל הספריות (Tools -> Manage Libraries):
 - **DHT sensor library** מאת Adafruit.
 - **LiquidCrystal_I2C** מאת Frank de Brabander.
 - **WiFi** ו-**WebServer** (משולבות כחלק מליבת ה-ESP32).

5. קוד המקור המלא (Arduino / ESP32 Code)

הקוד הבא מייצר נקודת גישה אלחוטית (Access Point) או מתחבר לרשת ה-Wi-Fi הביתית, מעלה שרת אינטרנט פנימי המציג נתוני לחות וטמפרטורה, מאפשר שליטה בלחיצת כפתור באתר או בלחיצה פיזית בכפתור הקיים בערכה, ומפעיל מנגנון קירור אוטומטי במידה והטמפרטורה בחממה עולה על 30°C.

```
#include <WiFi.h>
#include <WebServer.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "DHT.h"

// (Access Point-ניתן לשנות לחיבור הביתי או להשתמש ב) Wi-Fi הגדרות
const char* ssid = "Greenhouse_IoT";
const char* password = "SmartGreenhouse123";

// הגדרת פנים
#define DHTPIN 4
#define DHTTYPE DHT22
```

```

#define RELAY_PUMP 18      // ממסר 1 - משאבת השקיה
#define RELAY_FAN 19       // ממסר 2 - מאוורר קירור
#define BUTTON_PIN 13      // כפתור השקיה ידנית

DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27, 16, 2);
WebServer server(80);

bool pumpState = false;
bool fanState = false;

int lastButtonState = LOW;
unsigned long lastDebounceTime = 0;
unsigned long debounceDelay = 50;

unsigned long lastSensorRead = 0;
const long sensorInterval = 2000;

float temp = 0.0;
float hum = 0.0;

// של ממשק הניהול HTML-בניית עמוד ה
String sendHTML() {
    String ptr = "<!DOCTYPE html><html dir='rtl'><head><meta
charset='UTF-8'><meta name='viewport' content='width=device-width,
initial-scale=1.0'>";
    ptr += "<title>בקרת חממה חכמה</title>";
    ptr +=
"<style>body{font-family:Arial,sans-serif;text-align:center;background:
#f0f7f4;color:#333;margin:0;padding:20px;}</style>";
    ptr += "</head><body>";
    ptr += "<h1>🌱 והשקיה חממה בקרת מערכת</h1>";
    ptr += "<div
style='background:white;padding:20px;border-radius:10px;max-width:40
0px;margin:0 auto;box-shadow:0 2px 5px rgba(0,0,0,0.1);'>";
    ptr += "<h2>נתוני אקלים בחממה</h2>";
    ptr += "<p style='font-size:18px;'>טמפרטורה: <b>" + String(temp, 1)
+ " °C</b></p>";
    ptr += "<p style='font-size:18px;'>לחות אוויר: <b>" + String(hum, 1)

```

```

+ " %</b></p>";
ptr += "<hr>";
ptr += "<h2>עומסים בקרת</h2>";
ptr += "<p>משאבת השקיה: <b>" + String(pumpState ? "כבוי" : "פעילה") +
"</b></p>";
ptr += "<a href='/togglePump'><button style='padding:10px
20px;background:#2d6a4f;color:white;border:none;border-radius:5px;font-size:16px;cursor:pointer;'>משאבה מצב שינוי</button></a>";
ptr += "<p style='margin-top:20px;'>אוטומטי (קירור): <b>" +
String(fanState ? "כבוי" : "פעיל") + "</b></p>";
ptr += "</div></body></html>";
return ptr;
}

```

```

void handleRoot() {
  server.send(200, "text/html", sendHTML());
}

```

```

void handleTogglePump() {
  pumpState = !pumpState;
  digitalWrite(RELAY_PUMP, pumpState ? LOW : HIGH);
  server.sendHeader("Location", "/");
  server.send(303);
}

```

```

void setup() {
  Serial.begin(115200);

  pinMode(RELAY_PUMP, OUTPUT);
  pinMode(RELAY_FAN, OUTPUT);
  digitalWrite(RELAY_PUMP, HIGH); // כיבוי ממסר (Active LOW)
  digitalWrite(RELAY_FAN, HIGH);

  pinMode(BUTTON_PIN, INPUT);

  dht.begin();
  lcd.init();
  lcd.backlight();
}

```

```

// כנקודת גישה ESP32 הגדרת (SoftAP)
WiFi.softAP(ssid, password);
Serial.println("Access Point Started");
Serial.print("IP Address: ");
Serial.println(WiFi.softAPIP());

lcd.setCursor(0, 0);
lcd.print("Greenhouse System");
lcd.setCursor(0, 1);
lcd.print("IP: ");
lcd.print(WiFi.softAPIP());
delay(3000);
lcd.clear();

// הגדרת נחיבי שרת ה-Web
server.on("/", handleRoot);
server.on("/togglePump", handleTogglePump);
server.begin();
}

void loop() {
    server.handleClient();
    unsigned long currentMillis = millis();

    // --- קריאת חיישן ---
    if (currentMillis - lastSensorRead >= sensorInterval) {
        lastSensorRead = currentMillis;

        float h = dht.readHumidity();
        float t = dht.readTemperature();

        if (!isnan(h) && !isnan(t)) {
            temp = t;
            hum = h;

            // עדכון LCD
            lcd.setCursor(0, 0);
            lcd.print("T:" + String(temp, 1) + (char)223 + "C H:" +
String(hum, 0) + "% ");

```

```

    lcd.setCursor(0, 1);
    lcd.print("Pump:" + String(pumpState ? "ON " : "OFF") + "
Fan:" + String(fanState ? "ON " : "OFF"));

    // לוגיקה אוטומטית למאוורר קירור (מעל 30 מעלות)
    if (temp > 30.0) {
        fanState = true;
        digitalWrite(RELAY_FAN, LOW);
    } else {
        fanState = false;
        digitalWrite(RELAY_FAN, HIGH);
    }
}

// --- לחצן ידני עוקף להשקיה ---
int reading = digitalRead(BUTTON_PIN);
if (reading != lastButtonState) {
    lastDebounceTime = millis();
}
if ((millis() - lastDebounceTime) > debounceDelay) {
    static int buttonState = LOW;
    if (reading != buttonState) {
        buttonState = reading;
        if (buttonState == HIGH) {
            pumpState = !pumpState;
            digitalWrite(RELAY_PUMP, pumpState ? LOW : HIGH);
        }
    }
}
lastButtonState = reading;
}

```

6. שלבי בדיקה, הפעלה ודיבאג

1. חיבור והעלאת הקוד: חברו את ה-ESP32 למחשב והעלו את הקוד דרך Arduino IDE.
2. חיבור אלחוטי לממשק: במכשיר הנייד או במחשב, כפשו רשת Wi-Fi בשם Greenhouse_IoT

והתחברו עם הסיסמה SmartGreenhouse123.

3. **גישה לדאשבורד:** פתחו בדפדפן את הכתובת 192.168.4.1. תוכלו לראות את הנתונים בזמן אמת וללחוץ על הכפתור במסך להפעלת המשאבה.

4. **בדיקת לחצן פיזי:** לחצו על הכפתור הצהוב בערכה (EL4014) וודאו כי הממסר משנה את מצבו והנתון במסך ה-LCD מתעדכן בהתאם.