



Projektarbeit Mikroprozessortechnik
Pflichtenheft
Projekt: Raumklima Thermo- Hygrometer

Version 1.2

Francisco Stocker & Thomas Meier

Name	Datum	Unterschrift
Francisco Stocker	2023-01-30	
Thomas Meier	2023-01-30	

Änderungsgeschichte

<u>Datum</u>	<u>Version</u>	<u>Autor</u>	<u>Beschreibung</u>
2023-01-10	0.1	F.Stocker	1. Entwurf des Pflichtenhefts erstellt
2023-01-16	0.2	T.Meier	Elektronik-Schema eingefügt
2023-01-16	0.3	T. Meier	Fluss-Diagramm eingefügt
2023-01-17	0.4	F.Stocker	2. Entwurf des Pflichtenhefts geschrieben
2023-01-25	0.5	F.Stocker	IBN und Test SW & HW protokolliert
2023-01-26	1.0	F.Stocker	Version 1.0 erstellt
2023-01-28	1.1	T.Meier	Korrekturlesen
2023-01-30	1.2	F.Stocker & T.Meier	Abgabe Pflichtenheft

Inhaltsverzeichnis

1	EINLEITUNG.....	5
1.1	Aufgabestellung.....	5
1.2	Zweck	5
1.3	Terminplan	5
1.4	Team-Organisation	5
1.5	GitHub	5
2	ALLGEMEINE BESCHREIBUNG.....	6
2.1	Produktfunktionen	6
2.2	Benutzereigenschaften.....	6
2.3	Einschränkungen.....	6
2.4	Priorisierung der Anforderungen	6
3	ELEKTRONIK-SCHEMA.....	7
3.1	Liste der benutzten Elektronik-Elemente.....	7
4	SOFTWARE.....	8
4.1	Flussdiagramm	8
4.2	Software Regeln	8
4.3	Sensor Messung.....	9
4.4	LCD-Anzeige.....	9
5	INBETRIEBNAHME – UND TEST PROTOKOLL.....	10
6	SOLL-IST.....	11
A	TERMINPLAN	12
B	DATENBLÄTTER	13
C	SOFTWARE-CODE.....	14

Abbildungsverzeichnis

Abbildung 1: Ausschnitt aus Terminplan im MS-Project erstellt.....	5
Abbildung 2: Elektronik-Schema mit dem Programm «Fritzing» erstellt.....	7
Abbildung 3: Flussdiagramm der Software Regeln.....	8
Abbildung 4:Bsp. LCD-Anzeige	9

Tabellenverzeichnis

Tabelle 1: Bauteile.....	7
Tabelle 2: SW-Regeln Ampel.....	8
Tabelle 3:IBN und Testprotokoll.....	10

1 Einleitung

1.1 Aufgabestellung

Im Fach Mikrocomputertechnik an der Höhere Fachschule «Teko» in Zürich, ist der Auftrag erteilt worden ein Projekt zu realisieren und zu dokumentieren welches sich mit der Thematik der Projekt- Planung und Realisierung im Bereich der Software- und Hardware-Entwicklung den Schwerpunkt haben soll.

Die Studenten sollen dieses Projekt maximal zu zweit umsetzen.

1.2 Zweck

Im vorliegenden Dokument sind die Anforderungen definiert, welche im Projekt «Raumklima Thermo-Hygrometer» umgesetzt werden müssen. Es beschreibt den Auftrag zwischen Auftraggeber und Auftragnehmer. Der Ausdruck *Pflichtenheft* ist hier im Sinne der *IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE Std 830-1998* verwendet. Die dort definierte *Requirements Specification* beinhaltet sowohl die Benutzeranforderungen (*Lastenheft* gemäss DIN 69901-5) als auch Realisierungsvorgaben an die Entwicklungsgruppe (*Pflichtenheft* gemäss DIN 69901-5).

1.3 Terminplan

Mit diesem Terminplan wird der Ablauf der Teilaufgabe gemäss dem Bewertungsblatts dargestellt. Gleichzeitig hat der Terminplan die Funktion der Aufgabenteilung zwischen den Projektbeteiligten Studierenden.

Der Terminplan ist im Anhang «A» beigelegt.

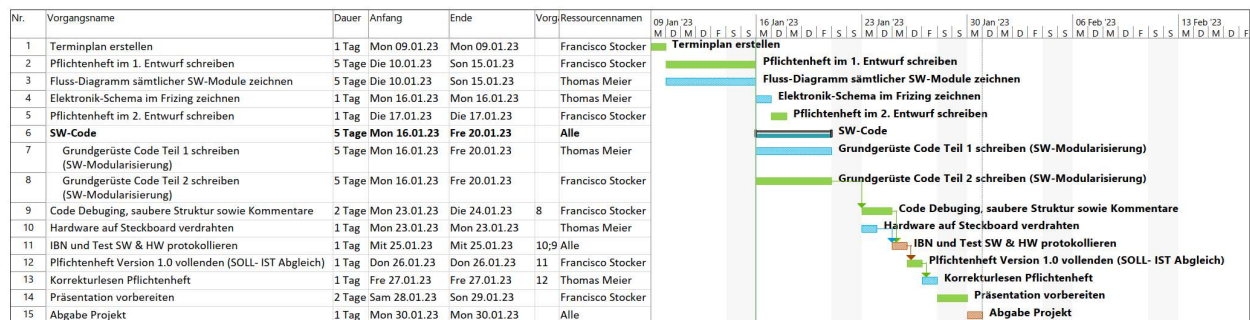


Abbildung 1: Ausschnitt aus Terminplan im MS-Project erstellt

1.4 Team-Organisation

Siehe Punkt 1.3 Terminplan.

1.5 GitHub

Das Projekt wurde auf github.com hochgeladen, um die Versionierung des Software-Code zu dokumentieren.

Unter folgendem Link kann auf das Projekt zugegriffen werden:

<https://github.com/Seff141/Projektarbeit.git>

Das Projekt ist als privat markiert und kann nur mit der entsprechenden Einladung der Projektverfasser geöffnet werden.

2 Allgemeine Beschreibung

2.1 Produktfunktionen

Die Hauptfunktionen dieses Produktes sind die Messung der Umgebungs-Temperatur und der Umgebungs-Luftfeuchtigkeit.

2.2 Benutzereigenschaften

Der Benutzer muss keine besonderen elektrotechnischen Kenntnisse haben. Festlegung der Benutzer (Benutzergruppen), welche mit dem System arbeiten sollen, z.B. Routinebenutzer, Servicetechniker, Administrator.

2.3 Einschränkungen

Da es sich hier um einen Aufbau ohne jeglichen IP-Schutz handelt, kann dieser nur in Witterungsgeschützten Innenräume benutzt werden.

Der eingesetzte Sensor hat für die Messungen folgende Range:

- Luftfeuchtigkeit: 20-80 Prozent
- Temperatur -Range: 0-50 Grad Celsius

2.4 Priorisierung der Anforderungen

2.4.1 Muss-Anforderung

Dies ist eine Anforderung, welche für das System essenziell und unabdingbar ist, bzw. das System würde keinen Sinn ergeben, wenn diese Anforderung nicht implementiert wäre.

Der Aufbau hat folgende Muss-Anforderungen:

- Temperatur messen
- Feuchtigkeit messen
- Temperatur in Grad Celsius auf Display anzeigen.
- Feuchtigkeit in Prozent auf Display anzeigen.

2.4.2 Soll-Anforderung

eine Soll-Anforderung ist nicht unabdingbar, trägt jedoch zur wesentlichen Verbesserung des Systems bei. Sie soll wenn möglich realisiert werden.

Es soll eine Anzeige der Luftfeuchtigkeit mittels einer Ampel implementiert werden.

3 Elektronik-Schema

Um den Hardwareaufbau des Projektes vorab zu visualisieren und um die entsprechenden Elektronik-Elemente zu bestellen, wurde dies mit dem Programm «Fritzing» umgesetzt.

Im Anschluss werden alle benutzten Elemente in einer Liste aufgeführt.

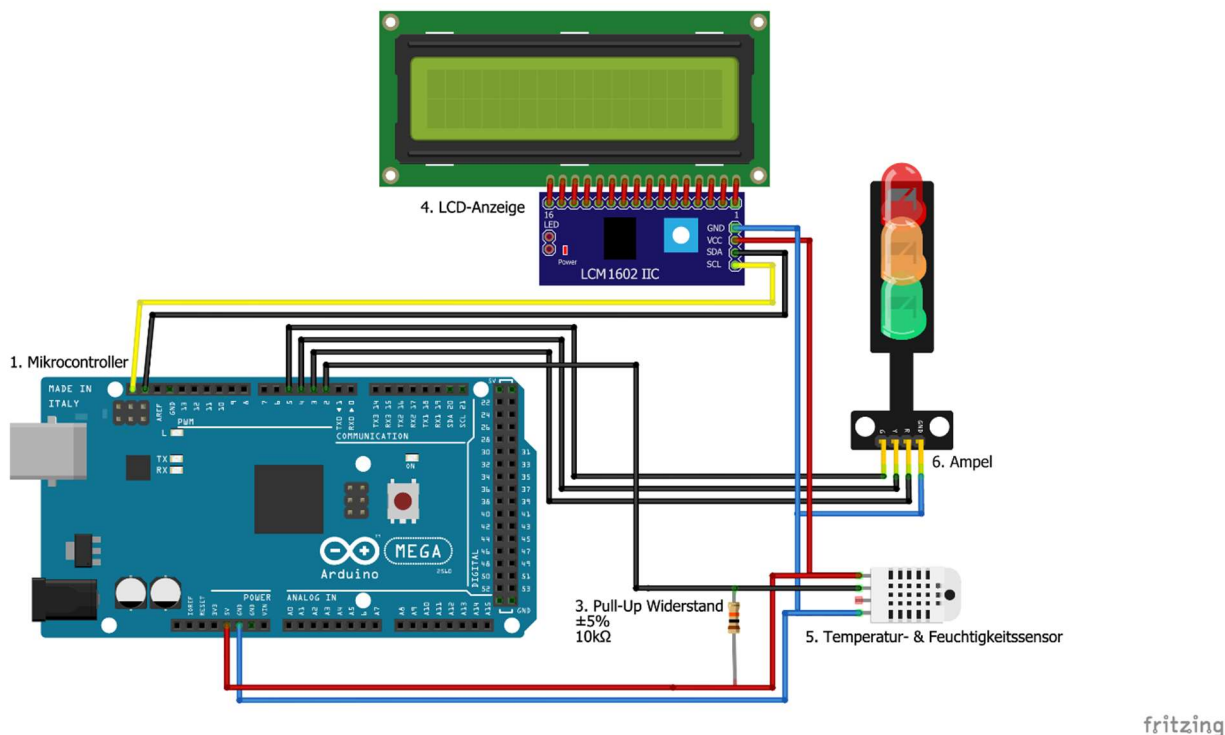


Abbildung 2: Elektronik-Schema mit dem Programm «Fritzing» erstellt

3.1 Liste der benutzten Elektronik-Elemente

Nr.	Name	Typ	Funktion
1	Mikrocontroller	Funduino-Mega 2560	Spannungsquelle 5V & umsetzt SW
2	Breadboard	Steckboard aus Arduino Kit	Steckübergang für Spannungsversorgung
3	Pull-Up Widerstand	10kOhm Widerstand	Digitaler Eingangsspannung sichern
4	LCD – Anzeige	I2C LCD	Darstellung der Ausgabe
5	Temperatur- & Feuchtigkeitssensor	DHT11	Messeinrichtung
6	Ampel	Ampel aus Arduino Kit	Darstellung Feuchtigkeit in Abstufungen nach Prozent

Tabelle 1: Bauteile

Die Datenblätter zu den einzelnen Bauteilen sind im Anhang «B» zu finden.

4 Software

4.1 Flussdiagramm

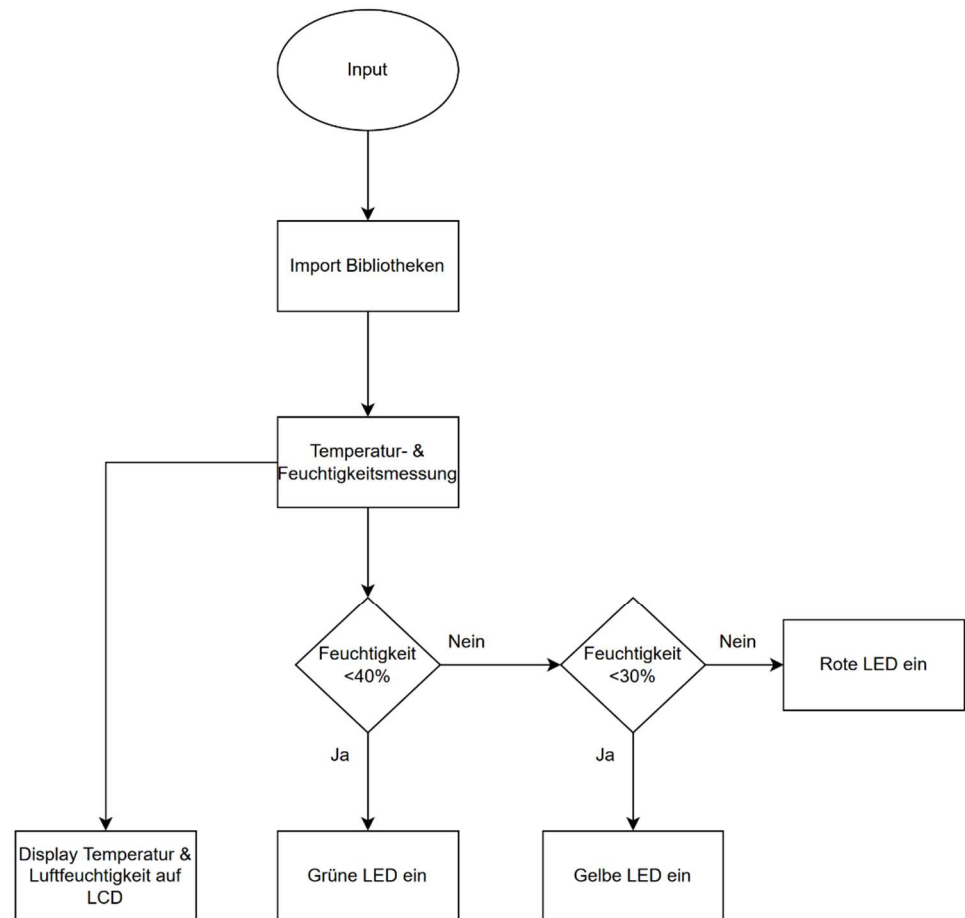


Abbildung 3: Flussdiagramm der Software Regeln

4.2 Software Regeln

An dieser Stelle werden die Regeln für die Anzeige der Messung der Feuchtigkeit durch das Ampelsystem definiert:

LED	Feuchtigkeit Range
Grün	$\leq 40\%$ - 80%
Gelb	<30 % - 39%
Rot	<20% - 29%

Tabelle 2: SW-Regeln Ampel

4.3 Sensor Messung

Die Messung mit dem Sensor DHT11 wird mit der Bibliothek «dht.h» von Adafruit gemacht. Der Typ des Sensors ist zu definieren und im «setup» des SW-Codes zu starten.

Die Temperatur wird mit dem Datentyp «float» definiert und aus der Bibliothek «dht.h» ausgelesen.

Die Luftfeuchtigkeit wird mit dem Datentyp «float» definiert und aus der Bibliothek «dht.h» ausgelesen.

Da der Sensor eine Vorlaufzeit zum Auslesen braucht ist diese im «Loop» als «delay» von 2 Sekunden zu definieren.

4.4 LCD-Anzeige

Die LCD-Anzeige ist mit folgenden Bibliotheken zu verwenden:

- Wire.h
- LiquidCrystal_I2C.h

Die LCD-Anzeige hat zwei Zeilen, in welche die Messungen angezeigt werden.

In der oberen Zeile wird die Temperatur mit dem Befehl «lcd.print» dargestellt.

Die untere Zeile ist für die Darstellung der Luftfeuchtigkeit mit demselben Befehl vorgesehen.

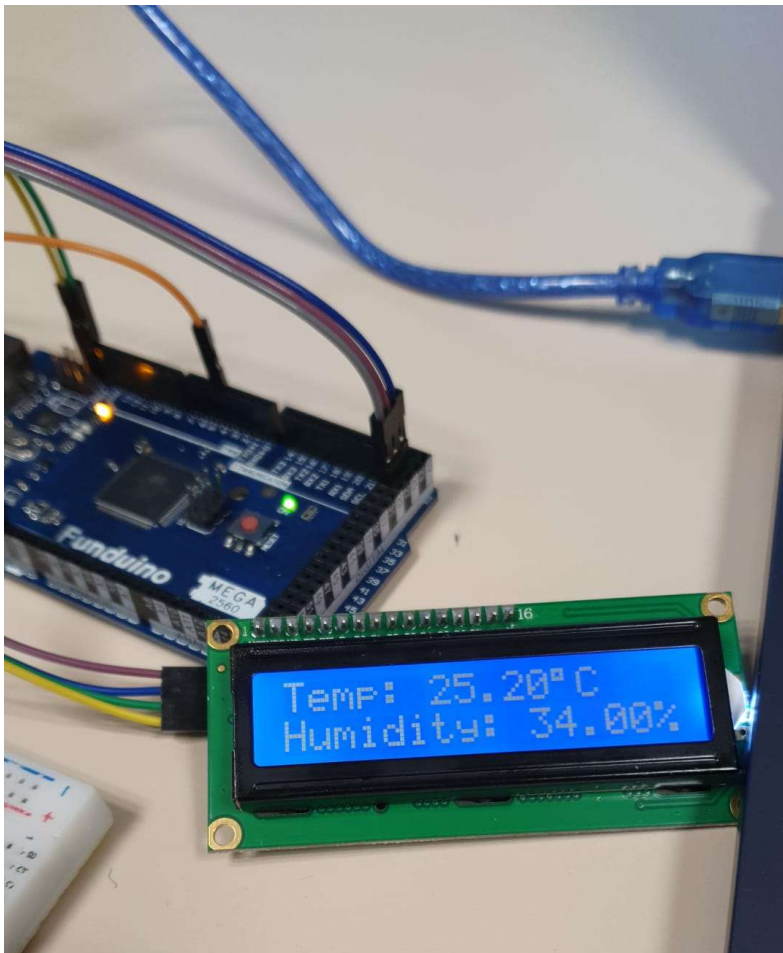


Abbildung 4:Bsp. LCD-Anzeige

5 Inbetriebnahme – und Testprotokoll

Element	Test-Art	Messinstrument	Resultat
Spannungsversorgung	Spannungsmessung am Funduino Mikrocontroller	Multimeter	+/- 5V
Kabelverbindungen	Sichtvergleich mit Elektronik-Schema		Korrekt verdrahtet
Temperatur-Messung	Vergleich mit handelsüblichem Thermo-Hygrometer	Raumtemperatur Messeinrichtung «Boneco X200»	Vertretbare Abweichung von $\pm 2^{\circ}\text{C}$
Feuchtigkeit-Messung	Vergleich mit handelsüblichem Thermo-Hygrometer	Raumtemperatur Messeinrichtung «Boneco X200»	Vertretbare Abweichung von $\pm 5\%$
LCD Anzeige	Sichtkontrolle		Helligkeit ist je nach Umgebung am Potenziometer anzupassen. Anzeige wird in korrekter Reihenfolge gemäss Vorgabe dargestellt.
Ampel	Sichtkontrolle und Anhauchen des Sensors		Ampel war defekt und wurde durch LED's ersetzt. LED Leuchtstärk ist sehr minim. Siehe Kapitel 6.

Tabelle 3: IBN und Testprotokoll

6 SOLL-IST

Das Projekt wurde gemäss den oberen Kapiteln realisiert und getestet. Das Ampelsystem konnte nicht gemäss Elektronik-Schema (Kapitel 3) realisiert werden, da das vorhandene Elektronik-Element beim Zeitpunkt der Inbetriebnahme nicht funktionstüchtig war. Als alternative wurde auf handelsübliche LEDs ausgewichen, welche in den Farben einer Ampel ausgewählt wurden.

Weiter ist aufgefallen, dass die Spannungsversorgung von Mikrocontroller nicht ausreichend ist, um den Fühler und die LEDs gleichzeitig zu betreiben. Dies hat sichtliche Auswirkungen auf die Leuchtstärke der LEDs der Ampel.

Anhang A – Terminplan

A Terminplan









Nr.	Vorgangsname	Dauer	Anfang	Ende	Vorg	Ressourcennamen	09 Jan '23	16 Jan '23	23 Jan '23	30 Jan '23	06 Feb '23	13 Feb '23	20 Feb '23	27 Feb
							M D M D F S S	M D M D F S S	M D M D F S S	M D M D F S S	M D M D F S S	M D M D F S S	M D M D F S S	M D M
1	Terminplan erstellen	1 Tag	Mon 09.01.23	Mon 09.01.23		Francisco Stocker	■	■						
2	Pflichtenheft im 1. Entwurf schreiben	5 Tage	Die 10.01.23	Son 15.01.23		Francisco Stocker	■	■						
3	Fluss-Diagramm sämtlicher SW-Module zeichnen	5 Tage	Die 10.01.23	Son 15.01.23		Thomas Meier	■	■						
4	Elektronik-Schema im Frizing zeichnen	1 Tag	Mon 16.01.23	Mon 16.01.23		Thomas Meier		■						
5	Pflichtenheft im 2. Entwurf schreiben	1 Tag	Die 17.01.23	Die 17.01.23		Francisco Stocker		■						
6	SW-Code	5 Tage	Mon 16.01.23	Fre 20.01.23		Alle		■						
7	Grundgerüste Code Teil 1 schreiben (SW-Modularisierung)	5 Tage	Mon 16.01.23	Fre 20.01.23		Thomas Meier		■						
8	Grundgerüste Code Teil 2 schreiben (SW-Modularisierung)	5 Tage	Mon 16.01.23	Fre 20.01.23		Francisco Stocker		■						
9	Code Debuging, saubere Struktur sowie Kommentare	2 Tage	Mon 23.01.23	Die 24.01.23	8	Francisco Stocker			■					
10	Hardware auf Steckboard verdrahten	1 Tag	Mon 23.01.23	Mon 23.01.23		Thomas Meier			■					
11	IBN und Test SW & HW protokollieren	1 Tag	Mit 25.01.23	Mit 25.01.23	10;9	Alle			■					
12	Plfichtenheft Version 1.0 vollenden (SOLL- IST Abgleich)	1 Tag	Don 26.01.23	Don 26.01.23	11	Francisco Stocker			■					
13	Korrekturlesen Pflichtenheft	1 Tag	Fre 27.01.23	Fre 27.01.23	12	Thomas Meier			■					
14	Präsentation vorbereiten	2 Tage	Sam 28.01.23	Son 29.01.23		Francisco Stocker			■					
15	Abgabe Projekt	1 Tag	Mon 30.01.23	Mon 30.01.23		Alle			■					



Projekt: 230111_Terminplan Projekt-MP	Manueller Sammelrollup	■	Francisco Stocker	■	Alle	■
	Manueller Sammelvorgang	■	Thomas Meier	■		

Anhang B – Datenblätter

B Datenblätter

     (https://www.bastelgarage.ch/index.php?route=account/account)
 (https://www.bastelgarage.ch/index.php?route=account/wishlist)
 (https://www.bastelgarage.ch/index.php?route=checkout/cart)
 (https://www.bastelgarage.ch/index.php?route=checkout/checkout)



(https://www.bastelgarage.ch)

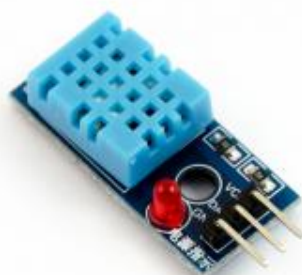
Suche



 0 Artikel - 0,00CHF

 (https://www.bastelgarage.ch)

DHT11 Temperatur und Luftfeuchtigkeitssensor (https://www.bastelgarage.ch/dht11-temperatur-und-luftfeuchtigkeitssensor)



(https://www.bastelgarage.ch/image/cache/catalog/Artikel/420081-420090/420082-800x800.jpg)



(https://www.bastelgarage.ch/image/cache/catalog/Artikel/420081-420090/420082-3-800x800.jpg)
(https://www.bastelgarage.ch/image/cache/catalog/Artikel/420081-420090/420082-4-800x800.jpg)

Beschreibung

Bewertungen (1)

Der DHT11 ist ein lowcost Luftfeuchtigkeit und Temperatur Sensor mit digitaler Schnittstelle, mit einem 1-Wire Anschluss. Der Sensor ist bereits werkseitig kalibriert, hat jedoch nicht so eine hohe Auflösung. Wenn du eine höhere Auflösung und bessere Genauigkeit willst, empfehlen wir dir auch den DHT22. Eine fertige Arduino Library für die Ansteuerung findest du im Library

Manager unter „DHT Sensor Library“. Auf dem Board sind bereits die Widerstände verbaut die zur Ansteuerung benötigt werden. Somit kannst du das Board einfach an dein Arduino anstecken und loslegen.

Falls du Fragen oder Probleme beim Ansteuern hast oder Anregungen zu der Produktbeschreibung, schreib uns einfach eine Mail oder nutze die Bewertungsfunktion. Wir helfen dir gerne weiter.

Anschlüsse:

DHT11: => Arduino UNO:
VCC => 5V oder 3.3V
Data => 2
GND => GND

Technische Details:

- Modell: DHT11
 - Betriebsspannung: 3.3V – 5.5V
 - Benötigt nur ein Pin (1-Wire)
 - Messbereich Luftfeuchtigkeit: 20% - 80% (±5% Genauigkeit)
 - Messbereich Temperatur: 0°C – 50°C (±2°C Genauigkeit)
 - Auflösung: 1 % rH/1 °C
 - Antwortzeit: <5s
 - Größe: 32 x 15 x 8mm
 - Gewicht: 5g
-

Lieferumfang:

1x DHT11 Breakout Board



DHT11 Temperatur und Luftfeuchtigkeitssensor

Artikelnr.: 420082

Verfügbarkeit: Lagernd

3,50CHF

Netto: 3,25CHF

Menge

+ Warenkorb

★★★★★ 1 Bewertungen () / + Bewertung ()

Anhang C – Software-Code

C Software-Code

```

1
2  /*=====
3      Projektarbeit
4  =====*/
5
6  /*=====
7  Titel: Projektarbeit
8  Description: Temperatur- und Feuchtigkeitsmessung mit LCD Display
9  Author: Thomas Meier & Francisco Stocker
10 Date: 28.01.2023
11 Version: V01.00.00
12 =====*/
13
14 #include <DHT.h> // "DHT sensor library" by Adafruit
15 #include <Wire.h> // Library für LCD
16 #include <LiquidCrystal_I2C.h> // Library für LCD
17 #include "arduino_config.h" // Config-Datei mit Sensor Pinbelegung
18
19 #define DHTTYPE DHT11 // Es handelt sich um den DHT11 Sensor
20
21 DHT dht(DHTPIN, DHTTYPE); //Der Sensor wird ab jetzt mit"dth" angesprochen
22 LiquidCrystal_I2C lcd(0x27, 16, 2); // Pins des LCD festlegen
23
24 void setup()
25 {
26     dht.begin(); //DHT11 Sensor starten
27     lcd.init(); //LCD einschalten
28     lcd.backlight(); //LCD Rücklicht einschalten
29 }
30
31 void loop()
32 {
33     delay(2000); //Zwei Sekunden Vorlaufzeit (Sensor ist etwas träge)
34
35     float Luftfeuchtigkeit = dht.readHumidity(); //Luftfeuchtigkeit auslesen
36     float Temperatur = dht.readTemperature(); //Temperatur auslesen
37
38     // Ausgabe auf LCD
39     lcd.setCursor(0,0); //Setzt Cursor auf die 1. Zeile & die 1. Position
40     lcd.print("Temp: "); //Schreibt "Temp:"
41     lcd.print(Temperatur); //Zeigt die aktuelle Temperatur
42     lcd.print("\337C"); // \337 = Grad(°)
43     lcd.setCursor(0,1); //Setzt Cursor auf die 2. Zeile & die 1. Position
44     lcd.print("Humidity: ") //Schreibt "Humidity:"
45     ;lcd.print(Luftfeuchtigkeit); //Zeigt die aktuelle Luchtfeuchtigkeit
46     lcd.print("%"); //Zeigt die einheit der Luftfeuchtigkeit (%)
47
48     // Grüne LED wird eingeschalten sobald die Luftfeuchtigkeit grösser/gleich 40% ist.
49     if (Luftfeuchtigkeit >= 40) {
50         digitalWrite(LED_G, HIGH);
51     } else {
52         digitalWrite(LED_G, LOW);
53     }
54
55     // Gelbe LED wird eingeschalten sobald die Luftfeuchtigkeit kleiner als 40% und
56     // grösser/gleich 30% ist.
57     if (Luftfeuchtigkeit < 40 and Luftfeuchtigkeit >= 30) {
58         digitalWrite(LED_Y, HIGH);
59     } else {
60         digitalWrite(LED_Y, LOW);
61     }
62
63     // Rote LED wird eingeschalten sobald die Luftfeuchtigkeit kleiner als 30% ist.
64     if (Luftfeuchtigkeit < 30) {
65         digitalWrite(LED_R, HIGH);
66     } else {
67         digitalWrite(LED_R, LOW);
68     }
69 }

```

```
1  /*=====
2                                     config-file
3  =====*/
4
5  /*=====
6  Description: Stammdaten
7  Author: Thomas Meier & Francisco Stocker
8  Date: 28.01.2023
9  Version: V01.00.00
10 =====*/
11
12 const int DHTPIN = 2; // Der Sensor wird an PIN 2 angeschlossen
13 const int LED_R = 3; // Rote LED
14 const int LED_Y = 4; // Gelbe LED
15 const int LED_G = 5; // Grüne LED
```