



HOCHSCHULE OSNABRÜCK

UNIVERSITY OF APPLIED SCIENCES

Praktikum im Fach

Internet of Things (IoT) / Industrie 4.0

Versuch 4: Überwachung der Bodenfeuchtigkeit

Autoren: Marco Schaarschmidt (m.schaarschmidt@hs-osnabrueck.de)
Nicolas Lampe (n.lampe@hs-osnabrueck.de)

Version: 2.1

Wichtige Hinweise:

- Zu dieser Aufgabenstellung gehört ein Dokument mit Basisinformation.
- Bei Fragen oder Rückmeldungen zum Praktikumsversuch stellen Sie diese bitte primär zu den Praktikumszeiten (dienstags 14:30 bis 16:00). Bei dringenden Fällen schreiben Sie eine E-Mail an oder Nicolas Lampe (n.lampe@hs-osnabrueck.de) oder Sebastian Böhm (sebastian.boehm@hs-osnabrueck.de).

1 Einleitung

Innerhalb dieses Versuches soll der M5Stack mit einen Bodenfeuchtigkeitssensor (engl. *Earth Moisture Sensor*) verbunden werden. Ziel ist es, das Feuchtigkeitslevel der Erde in einem Blumentopf zu messen und dadurch entscheiden zu können, ob die Pflanze mit Wasser versorgt werden muss, oder ob die Erde noch feucht genug ist. Mittels der eingebauten *RGB-LED* Leisten im Akku-Modul soll auf der einen Seite mittels LEDs der Akkuladestand dargestellt werden und auf der anderen Seite ein rotes Blinken erzeugt werden, wenn die Bodenfeuchtigkeit unter ein kritisches Level fällt. Die Informationen über das Feuchtigkeitslevel und den Akkuladestand sollen innerhalb einer Weboberfläche bereitgestellt werden und dort einsehbar sein. Zusätzlich sollen die Werte auf dem Display angezeigt und zusätzlich via *MQTT* anderen Teilnehmern bereitgestellt werden. Für diesen Versuch werden aus dem IoT/I4.0-Box die folgenden Hardwarekomponenten benötigt:

- M5Stack Core
- Earth Moisture Sensor

Machen Sie sich vor Beginn des Praktikums mit der Funktionsweise des Earth Moisture Sensors, der RGB-LED-Leisten, dem MQTT-Protokoll sowie die im Praktikum eingesetzten Arduino-Libraries vertraut. Generell sollten Sie vor Beginn dieses Versuches die Basisinformationen gelesen und ebenfalls verstanden haben, damit Sie die Grundlagen zur Bearbeitung des Versuches besitzen.

2 Versuchsdurchführung

In den nachfolgenden Schritten soll zunächst der *Earth Moisture Sensor* in Betrieb genommen und kalibriert werden, sodass Sie in der Lage sind, Daten aufzuzeichnen und auszuwerten. Die Daten des Sensors sollen auf dem Bildschirm angezeigt werden und in weiteren Teilaufgaben über einen Webserver für andere Nutzer sowie über *MQTT* für andere Systeme bereitgestellt werden. Die Analyse der Daten ist nicht Aufgabe des Versuches und soll nur exemplarisch betrachtet werden. Als Versuchsprotokoll sind die nachfolgenden Fragen in schriftlicher Form zu beantworten und mit dem entwickelten Programmcode in OSCA vor dem Testat zu hinterlegen. Zusätzlich muss Ihr Versuchsprotokoll eine Beschreibung sowie Screenshots der Website und des Displays (Daten anderer Teilnehmer oder Gruppen) beinhalten.

- Erläutern Sie die generelle Funktionsweise des *Earth Moisture Sensors* und beschreiben Sie woran erkannt werden kann, ob die Bodenfeuchtigkeit gerade einen hohen oder niedrigen Wert hat.
- Wofür steht HTML? Erläutern Sie den grundlegenden Aufbau.
- Was ist der Unterschied zwischen HTTP und MQTT?
- Erläutern Sie, welche Vorteile das Bereitstellen der Informationen über Protokolle wie MQTT gegenüber HTML-Seiten aus Sicht von IoT-Geräten hat.
- Benennen Sie die Akteure, die an einer Kommunikation über MQTT beteiligt sind und beschreiben Sie in eigenen Worten die allgemeine Funktionsweise des MQTT-Protokolls.
- Beschreiben Sie, was *Quality of Service* (QoS) bedeutet und wie MQTT QoS umsetzt.
- Erläutern Sie beispielhaft, welche Typen von *Wildcards* MQTT unterstützt und zeigen Sie anhand eines Beispiels, wie diese eingesetzt werden können.
- Auf welche Schwierigkeiten sind Sie bei der Versuchsdurchführung gestoßen?

2.1 Inbetriebnahme und Kalibrierung des Earth Moisture Sensors

Der Bodenfeuchtigkeitssensor ist mit einem entsprechenden Grove-Kabel an Port B des M5Stacks anzuschließen. Für die Kalibrierung sind zunächst maximale und minimale Werte zu ermitteln. Entwickeln Sie ein Programm, mit dessen Hilfe Sie die Kalibrierung durchführen können. Betreiben Sie daher den Sensor an trockener Luft und anschließend in einem Glas mit Wasser. Lassen Sie sich die minimalen Werte auf dem Display oder auf der Konsole ausgeben und notieren Sie sich diese Werte, da sie für die spätere Programmierung benötigt werden.

Der Sensor sollte nicht zu weit in den Boden gesteckt werden. Maximal bis zur schwarzen Markierung an den Stäben!

Im Anschluss an die Kalibrierung schreiben Sie ein Programm, dass den relativen Anteil der Bodenfeuchtigkeit basierend auf den ermittelten minimalen und maximalen Werten in Prozent auf dem Bildschirm des M5Stacks ausgibt. Lassen Sie eine LED-Leiste dauerhaft rot aufblin-ken, wenn der Wert unter dem von Ihnen gewählten kritischen Punkt fällt.

2.2 Ermitteln und Ausgeben des Akkuladestandes

Programmieren Sie die Abfrage des Akkuladestands und geben diesen auf der jeweiligen ande-ren LED-Leiste in Form eines Art Strichbalkens aus. Zeigen Sie zusätzlich die Prozentzahl des Ladestandes auf dem Display an.

2.3 Accesspoint und Webserver auf dem M5Stack

Wie bereits im Praktikumsversuch *Smart Home* bei der *M5Stack Timer Camera*, kann auch der M5 Stack als Access Point betrieben werden. Dabei spannt der M5Stack ein eigenes Wireless Local Area Network (WLAN) auf, mit dem sich wiederum WLAN-Fähige Geräte wie Smart-phones, Tablets und Laptops verbinden können. Für die Bereitstellung eines Webserver nutzen Sie die Bibliothek *WebServer*. Um auf einer Website Text darzustellen, wird Hypertext Markup Language (HTML) verwendet. Ein Beispiel für eine einfache HelloWorld HTML-Seite sieht wie folgt aus:

```
<!DOCTYPE html>
<html>
<head><title>HelloWorld-Title</title></head>
<body><p>HelloWorld</p></body>
</html>
```

Der Body-Teil enthält den Text der Website, in dem später die bestimmten Werte stehen sollen. Der HTML-Code wird innerhalb des M5Stack Programms mit `print`-Befehlen ausgegeben.

Betreiben Sie den M5Stack im Access Point Modus, sodass Sie sich via Smartphone, Tablet oder Laptop entsprechend verbinden können. Programmieren Sie nun eine simple HTML-Web-seite auf der die Ausgabe des relativen Anteils in Prozent der Bodenfeuchtigkeit angezeigt wird.

2.4 Integrieren und Nutzen eines MQTT-Brokers

Als MQTT-Client nutzen wir im Rahmen des Praktikums die Bibliothek *PubSubClient*¹ in der Version 2.8.0, welche Sie über die Bibliotheksverwaltung in der Arduino IDE hinzufügen

¹ <https://github.com/knolleary/pubsubclient>

können. Es empfiehlt es sich einen der öffentlichen MQTT-Broker² zu verwenden, sofern Sie Ihre Lösung nicht Vorort testen. Welche MQTT-Broker Sie verwenden ist Ihnen freigestellt. Sprechen Sie sich jedoch mit Ihren Partnern ab, sodass Sie sich mit dem gleichen MQTT-Broker verbinden. Im folgenden finden Sie eine allgemeine Struktur für den Umgang mit der Bibliothek *PubSubClient*:

```
//Angabe des verwendeten Brokers
const char* mqtt_server = <BROKER_ADDRESS>;

//Initialisierung des PubSubClients
WiFiClient wClient;
PubSubClient client(wClient);

void setup() {
  //Setzen des Servers und Callbacks
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}

void callback(char* topic, byte* message, unsigned int length) {
  ...
}

void loop() {
  if (!client.connected()) {
    while (!client.connected()) {
      //Aufbauen einer Verbindung
      if (client.connect(<CLIENT_ID>)) {
        //Subscribe
        client.subscribe(<TOPIC>);
      }
    }
  }
  client.loop();
  //Publish
  client.publish(<TOPIC>, <MESSAGE>);
}
```

Erweitern Sie den Programmcode so, dass Sie die eigenen gemessenen Werte publishen und ebenfalls die Werte einer anderen Gruppe abrufen. Weiterhin soll die in 2.3 entwickelte HTML-Seite so angepasst werden, dass neben Ihren eigenen Datenwerten auch Datenwerte Ihres Gruppenpartners oder die Werte einer anderen Gruppe auf der Seite angezeigt werden.

² Bekannte öffentliche MQTT-Broker: HiveMQ (<http://www.mqtt-dashboard.com/> <http://www.hivemq.com/demos/websocket-client/>) sowie Mosquitto (<http://test.mosquitto.org/>)