

# Messuhr WLAN/MQTT Modul

## Installation und Betrieb



**rAAaware UG (haftungsbeschränkt)**  
**Heidelberg**

# Inhaltsverzeichnis

|  |    |
|--|----|
| 1. Einleitung.....                                 | 3  |
| 1.1. Bestandteile.....                             | 4  |
| 1.1.1. Model M1 (drehbares Gehäuse).....           | 4  |
| 1.1.2. Model M3 (feststehend).....                 | 4  |
| 2. Normalbetrieb.....                              | 5  |
| 2.1. Senden des Messwertes.....                    | 5  |
| 2.2. Messpausen / Stromsparfunktion (StandBy)..... | 5  |
| 3. Konfiguration.....                              | 6  |
| 3.1. Installation von curl.....                    | 6  |
| 3.2. Konfigurationsdaten in config.json.....       | 7  |
| 3.3. Datenübertragung.....                         | 8  |
| 4. Servicefunktionen.....                          | 8  |
| 5. Aktualisieren der Firmware.....                 | 9  |
| 6. MQTT.....                                       | 9  |
| 6.1. MQTT Broker.....                              | 9  |
| 6.2. Topics.....                                   | 10 |
| 6.2.1. Messwerte auslesen.....                     | 10 |
| 6.2.2. Eingangswerte.....                          | 11 |
| 6.2.3. Status und Diagnose.....                    | 11 |
| 7. Technische Informationen.....                   | 12 |
| 7.1. Aufbau.....                                   | 12 |
| 7.2. Montage.....                                  | 12 |
| 7.3. Ladestation.....                              | 13 |
| 7.4. Spannungsversorgung.....                      | 13 |
| 7.5. Stromverbrauch.....                           | 14 |
| 7.6. Messfrequenz.....                             | 14 |
| 7.7. Technische Daten.....                         | 14 |
| 7.8. Zertifikate.....                              | 15 |
| 8. Sonstiges.....                                  | 15 |
| 8.1. Rechtliches.....                              | 15 |
| 8.2. Kontaktdaten.....                             | 15 |
| 9. Anhang.....                                     | 16 |
| 9.1. Referenz config.json.....                     | 16 |

Version 1.0.5.0 // 05.12.2018

# 1. Einleitung

Dieses Dokument beschreibt die Installation und den Betrieb des Messuhr WLAN Moduls für Messuhren mit Digimatic Schnittstelle.

Das Modul führt diese Aufgaben durch:

- Verarbeitung und Wandlung der Binärwerte der Digimatic Schnittstelle.
- Steuerung des Displays.
- Steuerung des WLAN Kommunikationsflusses.
- Implementation des MQTT Datenprotokolls.
- Powermanagement des Systems (Standby Funktion).
- Möglichkeit zum Flashen der Software über die WLAN Verbindung.

Das Modul kennt 2 Betriebsarten:

- Konfiguration
- Normalbetrieb

In der Betriebsart [Konfiguration] funktioniert das Modul als WLAN Server und stellt einen WLAN Access Point bereit.

Somit kann über einen beliebigen Client eine Verbindung zum Modul hergestellt werden - z.B. wenn ein bereits konfigurierter Zugang zu einem bestehenden Netzwerk nicht mehr möglich ist. Über die Verbindung zu diesem Access-Point kann eine Konfiguration in das Modul geladen werden.

Der Zweck der Konfiguration besteht darin, das Modul für den Zugriff auf ein bestimmtes Netzwerk und einen bestimmten MQTT Server vorzubereiten.

Die Konfiguration wird idealerweise über einen CURL Upload einer JSON Konfigurationsdatei durchgeführt (siehe Kapitel "Konfiguration").

Nach erfolgreichem Laden einer Konfiguration startet das Modul im MQTT Betrieb. Der eigene WLAN Accesspoint wird deaktiviert.

Stattdessen versucht das Modul sich mit dem konfigurierten WLAN Accesspoint zu verbinden. Nach erfolgreicher Verbindung mit dem Accesspoint wird versucht eine Verbindung mit dem konfigurierten MQTT Server herzustellen.

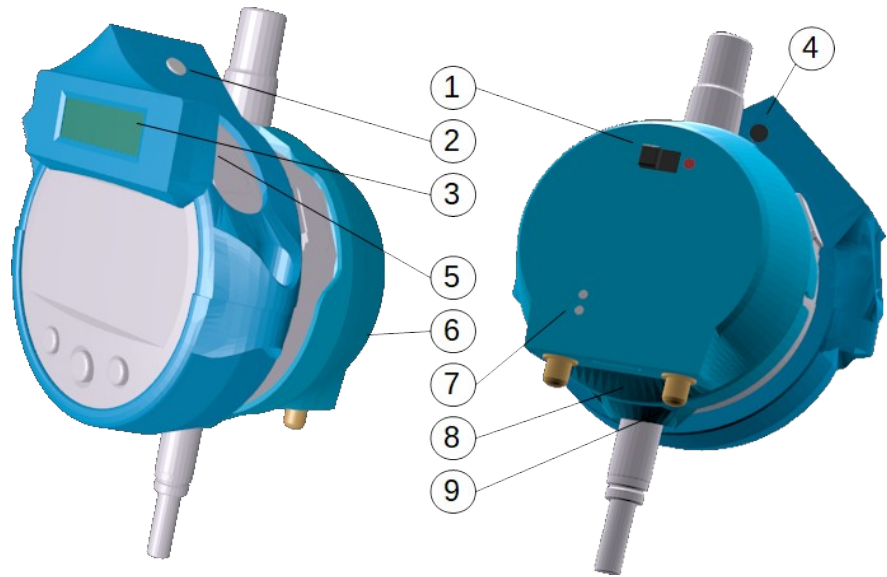
Ist diese Verbindung erfolgreich kann das Messgerät direkt über MQTT Botschaften versenden und empfangen.

Über eine Servicefunktion lässt sich das Modul wieder in den Konfigurationsmodus versetzen.

## 1.1. Bestandteile

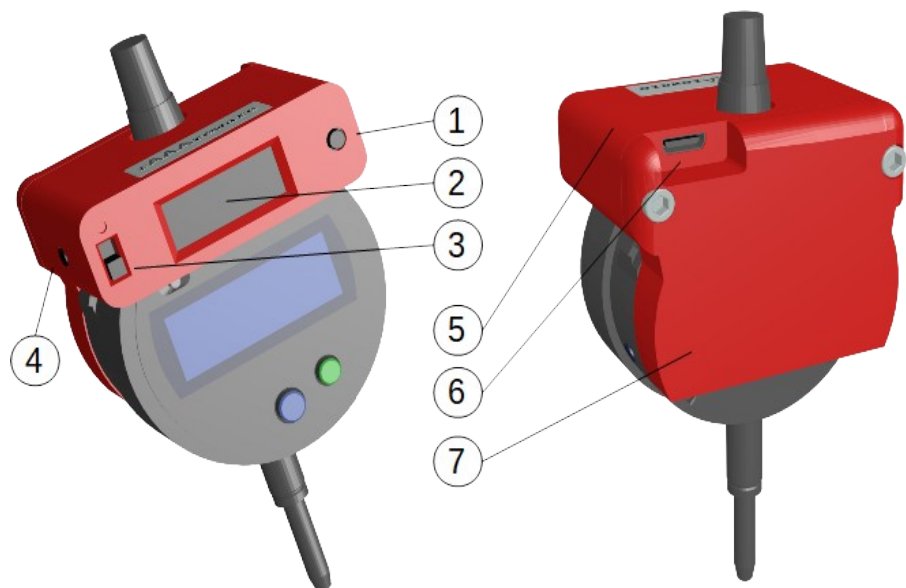
### 1.1.1. Model M1 (drehbares Gehäuse)

1. Hauptschalter (Schiebeschalter)
2. "Data" Taste (optional, wenn nicht an der Messuhr)
3. OLED Display
4. Servicetaste
5. Batterieöffnung Messuhr
6. Akku-Pack Messuhr-Modul
7. Ladekontrollanzeige
8. USB Ladebuchse
9. Ladekontakte für Ladestation



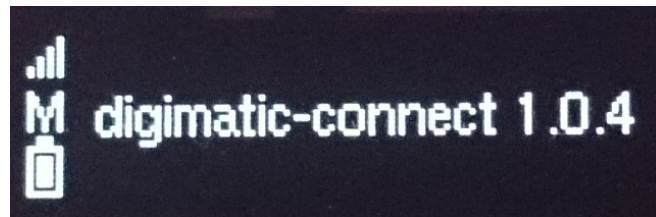
### 1.1.2. Model M3 (feststehend)

1. "Data" Taste
2. OLED Display
3. Hauptschalter (Schiebeschalter)
4. Servicetaste
5. Ladekontrollanzeige
6. USB Ladebuchse
7. Akku



## 2. Normalbetrieb

Das Modul wird über den Schiebeschalter am Akkupack eingeschaltet.  
Nach dem Einschalten wird auf dem Display "digimatic connect" zusammen mit der auf dem Modul vorhandenen Versionsnummer angezeigt.



*Sollte dies nicht angezeigt werden ist das Gerät noch nicht konfiguriert:  
Siehe Kapitel "Konfiguration" um das Modul zu konfigurieren.*

Zusätzlich werden auf der linken Seite 3 Mögliche Symbole eingeblendet:

Oben: WLAN Status

Mitte: MQTT Status

Unten: Status der Versorgungsspannung

Bei korrekter Konfiguration sollte nach kurzer Zeit das durchgestrichene WLAN Symbol verschwinden und der WLAN Signalstatus angezeigt werden.

Wenn eine Verbindung zum MQTT Broker hergestellt werden konnte, erlischt das mittlere "M" Symbol.

Über MQTT können nun Werte gesendet und empfangen werden.

Empfangene "Aufgaben" oder "Werkbank" Informationen (siehe MQTT Referenz) werden auf dem Display angezeigt. Ist der Text breiter als das Display wird der Text langsam gescrollt.

### 2.1. Senden des Messwertes

Bei jeder durchgeführten Messung wird der Messwert gesendet.

Eine Messung wird ausgelöst durch

- Drücken der "Data" Taste an der Messuhr.
- Drücken der "Data" Taste am Messuhr-Modul.
- Empfangen einer "digimatic/datarequest/set" Botschaft.
- Einen Zeitpunkt, wenn ein zyklisches Messen konfiguriert ist.

### 2.2. Messpausen / Stromsparfunktion (StandBy)

Nach einem über MQTT einstellbaren Zeitpunkt geht das Modul in den StandBy Modus.

Das Display wird hierbei ausgeschaltet und die WLAN Verbindung abgeschaltet.

Zum Aufwecken des Moduls wird einfach die "Data" Taste an der Messuhr oder am Modul kurz gedrückt. Auf dem Display erscheint dann nach wenigen Sekunden "Power on".

Der Zähler für den Wechsel in den Stromsparmmodus (StandBy/Sleep) wird zurückgesetzt durch:

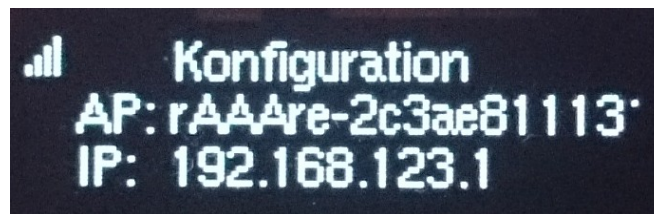
- Durchführen einer Messung.
- Empfang einer beliebigen gültigen MQTT Botschaft.  
(z.B. zum Einstellen des Displaytextes (Task/Workbench).)
- Eine der beiden Tasten am Modul gedrückt wird.

In den Messpausen sollte das Modul mit der Messuhr immer in der Ladestation abgelegt werden. Bei längerer Nicht-Benutzung oder beim Transport muss die Messuhr am Rückseitigen Schiebeschalter ausgeschaltet werden.

### 3. Konfiguration

Vor dem ersten Gebrauch oder nach einem Wechsel der WLAN/MQTT Umgebung muss das Modul konfiguriert werden.

Die Konfiguration dient dazu, dem Modul zu sagen, in welches WLAN es sich einwählen kann und mit welchem MQTT Server es sich verbinden soll.



Auf dem Display wird "Konfiguration" in der 1. Zeile angezeigt.

In der 2. Zeile wird ein WLAN Accesspoint angezeigt, unter welchem das Modul erreichbar ist. In der 3. Zeile wird "192.168.123.1" angezeigt: Dies ist die IP Adresse unter welcher das Modul innerhalb des Accesspoints erreichbar ist.

Um die Konfiguration zu senden wird von einem Rechner aus zunächst eine Verbindung zu dem Accesspoint hergestellt. Hierzu ist kein Passwort notwendig.

Wenn die WLAN Verbindung steht kann durch ausführen der Datei "curl.cmd" die Konfiguration in der Datei "config.json" auf das Modul übertragen werden.

Beispielinhalt der Datei curl.cmd:

```
curl -X PUT http://192.168.123.1/config --header "Content-Type: application/json" -d @config.json
```

Die Adresse [http://192.168.123.1/config] ist hierbei fest vorgegeben.

Die [config.json] muss die Konfigurationsdaten für das Modul enthalten.

#### 3.1. Installation von curl

"curl" ist ein freies Programm zur Datenübertragung. Es ist sehr verbreitet und auf vielen Rechnern schon installiert. Im Netz existieren zudem eine Vielzahl an Anleitungen und Beschreibungen zur Einsatz von curl.

Wenn curl noch nicht installiert ist oder nicht bekannt ist wo es installiert ist kann es einfach nochmal in ein Verzeichnis geladen werden.

Der Download erfolgt z.B. unter "https://curl.haxx.se/download.html".  
Es finden sich dort curl Versionen für viele verschiedene Rechnerarchitekturen.  
Für einen Windows PC:  
Unter "Packages" ist recht weit unten "Win32 - Generic" gelistet.  
Alle hier angebotenen Versionen und Varianten sind verwendbar.  
Es wird also einfach eine Version angeklickt und auf dem lokalen Rechner gespeichert.  
Nach dem Download werden die curl Dateien einfach in ein Verzeichnis entpackt, z.B.  
c:\prog\curl\  
Es wird hier bewusst nicht das Windows-Programmverzeichnis gewählt, da in diesem  
eventuell keine Schreibberechtigungen vorliegen.

Wenn die curl.cmd und curl selbst nicht im selben Verzeichnis liegen muss in der curl.cmd  
der Pfad zu curl mit angegeben werden. z.B.

```
c:\download\curl -X PUT http://192.168.123.1/config --header "Content-Type: application/json" -d @config.json
```

### 3.2. Konfigurationsdaten in config.json

Die Konfigurationsdaten beschreiben den Accesspoint mit welchem sich das Modul  
künftig verbinden soll. Weiter wird angegeben, bei welchem MQTT Server sich das Modul  
anmelden soll.

Nicht alle Inhalte der Datei müssen angepasst werden.  
Wichtig sind die Einträge:

"name": Klartextname der Messuhr, kann frei vergeben werden und kann dann über  
MQTT ausgelesen werden.

"ssid": Name des Accesspoints, mit welchem sich die Messuhr verbinden soll.

"password": Passwort des WLAN Zugangs (WPA2 Sicherheit).

"base\_topic": MQTT Topic unter welchem die Daten der Messuhr gesendet werden. Kann  
beliebig angegeben werden.

"host": Rechneradresse des Rechners auf welchem der MQTT Server/Broker im Netzwerk  
läuft.

"port": Port, auf welchem der MQTT Server läuft (normalerweise 1883).

Wenn das Modul kein DHCP Dienst verwenden soll um eine eigene IP Adresse zu  
beziehen kann alternativ auch eine feste IP Adresse für das Modul festgelegt werden (im  
Abschnitt "wifi"):

z.B. "ip": "192.168.1.5"

Beispiel einer config.json Datei:

```
{
  "name": "Messuhr1",
  "wifi": {
    "ssid": "WLAN_ACCESSPOINT_TEST",
    "password": "geheim"
  },
  "mqtt": {
    "base_topic": "topic/",
    "host": "192.168.1.44",
    "port": 1883,
    "auth": false
  }
}
```

```

    },
    "ota": {
        "enabled": false
    },
    "settings": {}
}

```

Eine Liste aller möglichen Konfigurationen der config.json ist im Anhang zu finden.

### 3.3. Datenübertragung

Nach der Installation von curl und dem Erstellen oder Anpassen der Dateien curl.cmd und config.json wird durch Aufruf der curl.cmd die Konfiguration an das Modul gesendet. Der Aufruf erfolgt über ein Konsolenfenster (Windows: "cmd") oder durch Doppelklick auf die curl.cmd im Dateifexplorer.

```

C:\WINDOWS\system32\cmd.exe
C:\dev\Messuhr\doc>D:\Prog\curl\I386\curl -X PUT http://192.168.123.1/config --header "Content-Type: application/json" -d @config_pe.json
{"success":true}
C:\dev\Messuhr\doc>pause
Drücken Sie eine beliebige Taste . . .

```

Als Antwort des curl Befehls erscheint in der Konsole ein "[...success]".

Nach erfolgreicher Konfiguration und einem durchgeführten Reset (z.B. durch Ein-/Ausschalten) startet das Modul im normalen Betriebsmodus ("Normalbetrieb").

### 4. Servicefunktionen

Über die Servicetaste an der Rückseite oder Seite des Modulkörpers können Sonderaufgaben an dem Messuhrmodul vorgenommen werden. Die Taste kann je nach Ausführung direkt betätigt werden oder wird mit einem kleinen Gegenstand (Bedienstift) durch die Gehäuseöffnung hindurch betätigt.

Bei jedem Druck auf die Taste leuchtet die blaue Status-LED des Moduls kurz auf und auf dem Display erscheint die zugehörige Information. Ist zu der angezeigten Information eine Funktion hinterlegt, kann diese Funktion ausgeführt werden in dem die Taste für über 3 Sekunden gedrückt wird. Ein kurzes Drücken der Taste schaltet zur nächsten Information oder Funktion weiter. Es können bis zu 9 Informationen oder Funktionen hinterlegt werden. Nach der letzten Funktion wird wieder mit der ersten Funktion begonnen. Wenn für über 2 Sekunden keine Taste gedrückt wird geht die Funktionswahl wieder in den Ausgangszustand über.

| Nr. | angezeigte Information   | Beschreibung                                     |
|-----|--------------------------|--|
| 1   | 1 - Spannung 3,22 Volt   | Aktuelle Versorgungsspannung.                    |
| 2   | 2 - IP 192.168.1.201     | Aktuelle lokale IP Adresse (DHCP oder statisch). |
| 3   | 3 - ID 23AB459000889933  | Geräte-ID (MAC-Adresse).                         |
| 4   | 4 - Sleep Timer: 60 sek. | Aktuell eingestellte Standby-Zeit.               |



| Nr. | angezeigte Information | Beschreibung  |
|-----|------------------------|---|
| 5   | 5 - Neu starten        | Modul neu starten (Reset).                                      |
| 6   | 6 - Firmware update    | Aktualisieren der Firmware.                                     |
| 7   | 7 -                    |   |
| 8   | 8 -                    |   |
| 9   | 9 - Factory reset      | Konfiguration löschen und wechseln in den Konfigurations-Modus. |

## 5. Aktualisieren der Firmware

Die Firmware der Messgeräte-Erweiterung kann über WLAN aktualisiert werden (OTA-Update/Over The Air Update). Nach der Freischaltung über die Servicetaste kann die neue Firmware gesendet werden.

Der Aktualisierungsvorgang wird auf dem Display angezeigt.

Die Aktualisierung wird nur dann aktiviert wenn die gesamte Firmware fehlerfrei über das WLAN in das Modul übertragen werden konnte.

Die neue Firmware-Versionsnummer wird dann nach dem nächsten Start des Moduls auf dem Display angezeigt.

Zum Aktualisieren der Firmware existiert ein eigenes Programm mit einer eigenen Anleitung.

Soll eine angeforderte OTA Aktualisierung doch nicht ausgeführt werden, kann das Modul einfach ausgeschaltet und wieder eingeschaltet werden. Anschliessend befindet es sich wieder im normalen Modus.

## 6. MQTT

Der Datenaustausch im Betriebsmodus erfolgt über einen MQTT Datenbroker.

Daten gesendet oder empfangen werden nur, wenn das Modul nicht im StandBy Modus ist. Dies ist am eingeschalteten Display erkennbar.

Um für Testzwecke MQTT Topics auslesen oder zu setzen kann das Programm MQTT.fx verwendet werden. Allerdings zeigt MQTT.fx nicht die gelieferten Werte welche auf ein SUBSCRIBE direkt geliefert werden an.

### 6.1. MQTT Broker

Der MQTT Datenverkehr wird über einen MQTT Server/Broker abgewickelt.

Im Internet sind viele Informationen zu MQTT zu finden, auch wie ein Server installiert und betrieben wird.

Siehe z.B. <https://mqtt.org/>.

Damit diese Dokumentation in sich geschlossen ist hier eine sehr schnelle und einfache Anleitung zur Installation des weit verbreiteten Brokers "Mosquitto" (<https://mosquitto.org/>) unter MS Windows (32 Bit). Wenn andere Produkte oder Versionen gewünscht sind sollten die Informationen im Internet verwendet werden.

- a)  
Download des Windows-Installers von  
<https://www.eclipse.org/downloads/download.php>  
(z.B. Datei `mosquitto-1.5.4-install-windows-x86.exe`)
- b)  
Download von "Win32 OpenSSL" unter  
<https://slproweb.com/products/Win32OpenSSL.html>  
(Win32 OpenSSL - EXE)
- c)  
Installation beider Pakete.
- d)  
Kopieren der Dateien "libssl-1\_1.dll", "libcrypto-1\_1.dll" vom Win32 OpenSSL Verzeichnis  
in das Mosquitto Programmverzeichnis.
- e)  
Damit der Broker als Windows-Dienst ausgeführt wird:  
Öffnen eines Kommandozeilenfensters (`cmd.exe`) im Admin-Modus.  
Ausführen von  
`mosquitto install`  
im Mosquitto Programmverzeichnis.

## 6.2. Topics

Die Topics sind hierarchisch aufgebaut:

`[base_topic]/[modul-id]/[sub-topic]`

Die Modul-ID ist die MAC Adresse des WLAN Moduls ohne Trennzeichen.

Beispiel:

`rAAAre/a020a61a53f2/$name`

Alle Werte werden als ASCII Zeichenkette gesendet.

Im Modul sind diese `[sub-topic]` definiert:

### 6.2.1. Messwerte auslesen

| Sub-Topic                         | Beschreibung           | Wertebeispiel |
|-----------------------------------|------------------------|---------------|
| <code>digimatic/value</code>      | Aktueller Messwert     | -2.303        |
| <code>digimatic/value/unit</code> | Einheit des Messwerts  | mm            |
| <code>digimatic/task</code>       | Aktuelle Messaufgabe   | Axialspiel    |
| <code>digimatic/workbench</code>  | Aktueller Arbeitsplatz | WB01          |
| <code>adc/voltage</code>          | Versorgungsspannung    | 3.23          |
| <code>adc/voltage/unit</code>     | Einheit Spannung       | V             |

### 6.2.2. Eingangswerte

| Sub-Topic               | Beschreibung         | Wertebeispiel   |
|-------------------------|----------------------|---|
| digimatic/request/set   | Messwert anfordern   | 0 (beliebig, wird nicht ausgewertet)  |
| intervalms/set          | Zyklisch messen      | 5000 (millisekunden)<br>Der minimale Wert liegt hier bei ca. 200 ms. Siehe auch unter "Technische Informationen".<br>Der Wert "0" schaltet den Intervall wieder ab. |
| digimatic/sleepsec/set  | Zeit bis zum StandBy | 60 (sekunden)<br>Minimal: 10 sek.<br>Maximal: 14 Tage<br>(14*24*60*60) = 1209600  |
| digimatic/task/set      | Messaufgabe setzen   | Axialspiel  |
| digimatic/workbench/set | Arbeitsplatz setzen  | WB01  |

Diese Werte müssen nach einem Power-OFF erneut gesetzt werden.

### 6.2.3. Status und Diagnose

| Sub-Topic        | Beschreibung                      | Wertebeispiel     |
|------------------|-----------------------------------|-------------------|
| \$fw/version     | Aktuelle Firmware Version         | 1.0.2             |
| \$localip        | IP Adresse des Moduls             | 192.168.1.201     |
| \$mac            | MAC Adresse des Moduls            | A0:20:A6:1A:53:F2 |
| \$name           | Name des Moduls                   | Messuhr           |
| \$online         | Zustand des Moduls                | true / false      |
| \$stats/interval | Zeitintervall für Statusbotschaft | 60 (sekunden)     |
| \$stats/signal   | WLAN Signalstärke                 | 100 (%)           |
| \$stats/uptime   | Aktuelle Laufzeit                 | 55 (sekunden)     |

Jeder dieser Werte wird einmalig geliefert, sobald das Topic abonniert wird. Wenn z.B. die WLAN Signalstärke erneut gelesen werden soll, muss der Wert neu angefordert werden (SUBSCRIBE). Die Werte sind hierbei mit dem RETAINED Flag versehen.

## 7. Technische Informationen

### 7.1. Aufbau

Das Modul wird direkt an bzw. um die Messuhr angebracht.  
2 gesicherte Schrauben verbinden das Modul stabil mit der Messuhr.

Im Modul verbaut ist der Microcontroller zur Steuerung der Einheit.  
Über ein WLAN Modul kann sich das Subsystem direkt mit einem WLAN Netzwerk verbinden. Das Modul wird intern über eine Kabelverbindung an die Digimatic Schnittstelle des Messgeräts angeschlossen. Alternativ kann das Modul auch direkt mit der Messuhr verlötet werden.

Bei einer externen Stromversorgung wird das Elektronikmodul über ein Verbindungskabel mit dem Stromversorgungsmodul verbunden.

Durch diese flexible Verbindung bleibt die Möglichkeit der Drehung des MitoToyo Display erhalten und das interne OLED Display des Moduls folgt der Drehung (Modul M1).

Das hochauflösende Display zeigt programmierbare Informationen an. Zusätzlich werden Geräteinformationen sowie der WLAN- u. Akku-Status angezeigt.

Ein programmierbarer Taster ermöglicht es zusätzliche Eingaben vorzunehmen oder das Modul in einen bestimmten Zustand zu versetzen.

Die autonome Stromversorgungseinheit mit Ladevorrichtung ist an der Rückseite der Messuhr aufgesteckt (Modul M1).

Das Gehäuse ist in Kunststoff (ABS oder PLA) gefertigt.

### 7.2. Montage

Anleitung zur Montage/Demontage des Moduls M1 an einer MitoToyo Messuhr.

Zur Montage des Moduls wird zunächst sichergestellt, dass das Messuhr-Modul am Hauptschalter des Akku-Moduls ausgeschaltet ist bzw. die Steckverbindung zu einer externen Spannungsversorgung getrennt ist (Die Einheit muss spannungsfrei sein).

Der 5polige Stecker wird dann in die Schnittstellenöffnung der Messuhr gesteckt.

Anschliessend wird die Messuhr eingeschaltet und das Modul mit Strom versorgt. Auf dem Modul-Display sollte eine erste Meldung erscheinen.

Durch auslösen einer Messung an der Messuhr muss das Messergebnis auf dem Display des Moduls angezeigt werden.

Alternativ kann das Modul mit der Messuhr fest verlötet werden. Dies garantiert eine sehr dauerhafte und stabile Verbindung des Moduls mit der Messuhr. Diese Montage wird durch uns durchgeführt.

Bei manchen Messuhren kann eine Messung nur extern ausgelöst werden: Durch die Messtaste am Modul kann hier eine Messung manuell angefordert werden.

Nach erfolgreichem Test wird das Modul über die Messuhr gestülpt und durch den unteren Haltebügel gesichert. Der Haltebügel wird mit 2 Schrauben V2A M3x12 mit den dazugehörigen Muttern verschraubt.

Achtung: Die Schrauben hierbei nur soweit anziehen, dass das Display der Messuhr noch leicht gedreht werden kann.

Das Akku-Modul wird Rückseitig auf die Messuhr gesteckt.

Die Demontage erfolgt in umgekehrter Reihenfolge.

### 7.3. Ladestation

Die Ladestation dient zum Aufladen des Akkus des Messuhr-Moduls (Modul M1). Wird die Messuhr gerade nicht verwendet sollte die Einheit Messuhr-Modul immer in der Ladestation aufgeladen werden.

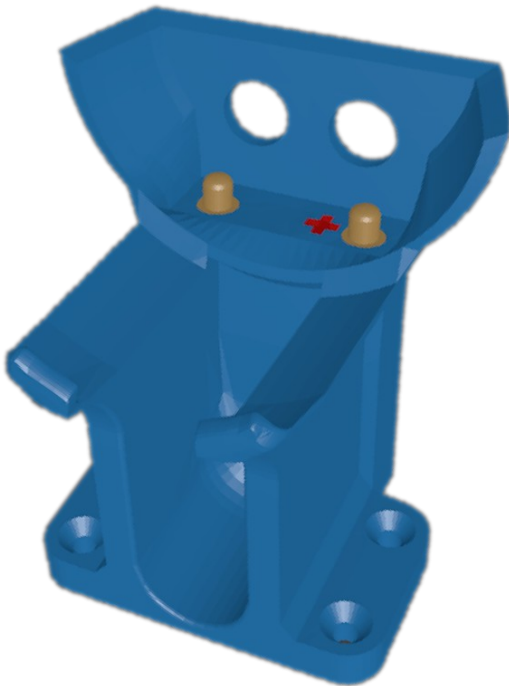


Abbildung 1: Ansicht Vorderseite



Abbildung 2: Ansicht Rückseite

Die Ladestation wird an einem festen Gegenstand verschraubt. Die Bohrabstände der Ladestation sind Breite = 44 mm, Tiefe = 20 mm. Die Bohrungen sind für M4 Senkkopfschrauben ausgelegt.

Die Stromversorgung erfolgt über die an der Rückseite angebrachte Hohlbuchse (5.5mm/2.1mm). Das mitgelieferte Netzteil liefert die benötigten 5V bei max. 500mA. Optional kann über ein Adapterkabel auch ein Laden an einer USB Buchse erfolgen.

### 7.4. Spannungsversorgung

Die externe Spannungsversorgung des Moduls beträgt 5 Volt. Die interne Spannungsversorgung beträgt 3.3V für das Modul und 1.5V für die Messuhr. Der Hauptschalter trennt nur das Modul von der Spannungsquelle. Die Stromversorgung der Messuhr wird immer aufrechterhalten, damit der gesetzte Nullpunkt der Messuhr erhalten bleibt.

## 7.5. Stromverbrauch

Der Stromverbrauch der IoT Messuhr Erweiterung beträgt im aktiven Betrieb ca. 100mA Strom bei 3.3 Volt Spannung.

Direkt während der sehr kurzen Sendephase des Messwerts kann der Stromverbrauch kurz auf bis zu 200mA ansteigen.

Bei fortlaufender Messung und Übertragung alle 10 sek. wird ein mittlerer Stromverbrauch von 110mA ermittelt.

Bei ununterbrochener Messung mit Übertragung des Messwerte wird ein 1000mA Akku somit ca. 9 Stunden die erforderliche Leistung bereitstellen.

Wenn im praktischen Betrieb nicht kontinuierlich gemessen wird verlängert sich die Akku-Laufzeit entsprechend:

Wenn z.B. alle 10 min. eine Messung über eine Minute durchgeführt wird ist von einer ca. 10 mal längeren Akku-Betriebszeit auszugehen. Eine volle Akku-Ladung sollte also für ca. 90 Stunden genügend Energie liefern.

Der Stromverbrauch im StandBy Modus ist mit einigen wenigen mA fast vernachlässigbar. Für eine z.B. längere Lagerung des Moduls sollte die Stromversorgung über den Hauptschalter am Akku-Modul vollständig getrennt werden.

## 7.6. Messfrequenz

Mit der Standard-Firmware dauert ein Messintervall ca. 200 ms.

Die minimale Zeit für den MQTT-Topic "intervalms/set" sollte also 200 ms nicht unterschreiten. Wird der Wert unterschritten werden die Werte schneller abgerufen, es ist jedoch nicht sichergestellt, dass die Werte auch tatsächlich schneller übertragen werden. Wird in einer Messanwendung eine schnellere Messdatenübertragung benötigt kann mit einer speziellen Firmware auch eine sehr hohe Messfrequenz umgesetzt werden. Zudem kann der synchronisierte Zeitstempel der Messung als zusätzliches Topic übertragen werden. Darüber wäre es dann auch möglich Messwerte zu interpolieren um eine Quasi-Echtzeit Messung durchzuführen.

## 7.7. Technische Daten

|                     | M1                                 | M3       | Einheit |
|---------------------|------------------------------------|----------|---------|
| Maße (BxHxT)        | 75x75x28                           | 61x50x44 | mm      |
| Gewicht             | 70                                 | 55       | g       |
| Temperaturbereich   | 0 - 50                             |          | °C      |
| Versorgungsspannung | 5                                  |          | V       |
| WLAN Frequenz       | 2.4 (IEEE802.11b/g/n; Kanäle 8-14) |          | GHz     |

## **7.8. Zertifikate**

Das eingesetzte Funkmodul besitzt CE Zeichen und erfüllt die Anforderungen in Bezug auf Sicherheit, Gesundheit, EMC (Elektromagnetische Verträglichkeit) und Funkstandards.

Die eingehaltenen Standards entsprechen den EN Standards EN60950: 2006+A11:2009+A1:2010+A12:2011+A2:2013. EN62311: 2008. EN301 489-1 V1.9.2 (2011-09). EN301 489-17 V2.2.1 (2012-09). EN200 328 V1.8.1 (2012-06).

Das Modul erfüllt die FCC Standards und ist geprüft unter FCC ID: 2AHMR-ESP12S.

## **8. Sonstiges**

### **8.1. Rechtliches**

MituToyo und Digimatic sind vermutlich eingetragene Warenzeichen von MituToyo. Wir verwenden diese Bezeichnungen hier zur Erklärung des IoT Moduls (der Hardware und Software) und stehen in keiner Verbindung zu MituToyo.

### **8.2. Kontaktdaten**

rAAaware UG (haftungsbeschränkt)  
Steigerweg 49  
69115 Heidelberg  
info@raaaware.de

## 9. Anhang

### 9.1. Referenz config.json

Weitere mögliche Konfigurationen in der config.json Datei:

```
{
  "name": "Modul 3", "device_id": "module3",
  "wifi": {
    "ssid": "wlan_network_1",
    "password": "secret",
    "bssid": "DE:AD:BE:EF:BA:BE",
    "channel": 1,
    "ip": "192.168.1.5",
    "mask": "255.255.255.0",
    "gw": "192.168.1.1",
    "dns1": "8.8.8.8",
    "dns2": "8.8.4.4"
  },
  "mqtt": {
    "host": "192.168.1.10",
    "port": 1883,
    "base_topic": "m3modules/",
    "auth": true,
    "username": "user",
    "password": "pass"
  },
  "ota": {
    "enabled": false
  }
}
```

Wenn aus dem Namen nicht bekannt ist was eine Einstellung bedeutet sollte sie auch nicht verwendet werden.

OTA wird normalerweise über unser eigenes externe Tool durchgeführt und sollte hier nicht aktiviert werden. Wenn es aktiviert ist kann ein Firmware Update auch über MQTT durchgeführt werden.