

# Implementierung eines Tracking Systems für Produkte auf einer Fertigungskette

## Kilian Röper (Embedded Systems)

**KEYWORDS** — NFC, MQTT, RITZ

### Bandumlaufsystem & Tracking Technologien

Die Lernfabrik fertigt teilautomatisch Produkte über eine Fertigungskette, die als Bandumlaufsystem abgebildet ist (Figure 1). Die Produkte werden simulativ als Raspberry Pi in einer 3D-gedruckten Schachtel über das Band transportiert. Es gibt vier Stationen: Ein Werkerassistent zur Produktvorbereitung, ein Schraubroboter, zum Befestigen der Schachtel, ein Hochregal, in dem fertige Produkte eingelagert werden und eine End-of-Line-Station, an der Informationen vom Raspberry Pi abgenommen werden.

Für die Steuerung und Nachverfolgbarkeit wurde ein System implementiert, das die Position und den Zustand der Produkte mittels NFC-Technologie registriert und diese an ein Manufacturing-Execution-System (MES) weitergibt.



Figure 1: Bandumlaufsystem im RITZ

Um das zu ermöglichen wurde an jeweils einer der Stationen ein Node-MCU-Controller angebracht und mit einem NFC-Sensor verbunden. Diese kommunizieren per MQTT mit einem auf einem Raspberry Pi laufenden MQTT-Broker, der dann die Informationen an ein zentrales MES-System mit Node-Red Visualisierung weiterleitet. Außerdem ist eine Schnittstelle zu einer SPS vorgesehen mit der das Band in Abhängigkeit vom Zustand der Tags gesteuert werden kann.

### MQTT & Systemarchitektur

MQTT (Message Queueing Telemetry Transport) ist ein leichtgewichtiges, offenes Netzwerkprotokoll für die asynchrone Kommunikation zwischen Geräten – besonders in IoT-Systemen (Internet of Things).

Es basiert auf dem Publisher/Subscriber-Modell:

1. Publisher senden Nachrichten zu einem Thema (Topic)
2. Subscriber abonnieren diese Themen und erhalten Nachrichten vom Broker

Die Node-MCU-Controller agieren sowohl als Publisher als auch als Subscriber, um Nachrichten sowohl zu senden als auch vor dem Beschreiben eines Tags zu empfangen (Figure 2).

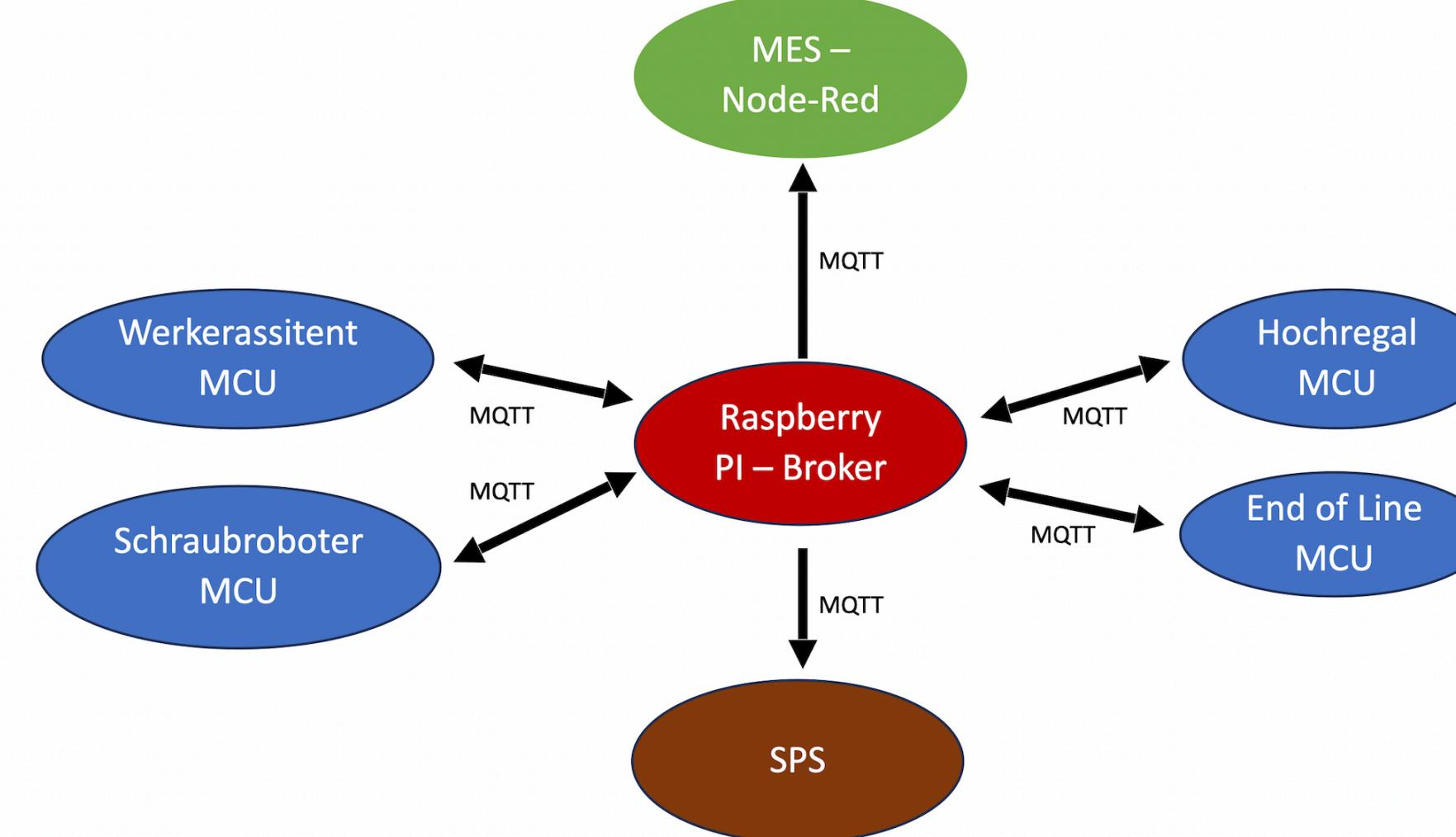


Figure 2: Systemarchitektur bestehend aus Raspberry Pi - Broker, Node-MCU, SPS und Manufacturing Execution System (MES)

Variable	Typ	Erklärung
DEBUG	define	ermöglicht das Ein- und Ausschalten von Debug-Messages
ssid	char-pointer	Netzwerk-SSID des Raspberry Pi
password	char-pointer	Passwort zum Raspberry Pi
topic_send	char-array	topic auf dem ein MCU-Controller sendet
topic_receive	char-array	topic das ein MCU-Controller zum hören abonniert

Table 1: Generische Softwarevariablen der Node-MCUs

Die Software auf den MCU-Controllern wurde einzeln entsprechend der Anforderungen einer Station implementiert und bietet generische Konfigurationsvariablen. Table 1 listet davon beispielhaft einige auf. Ein besonderes Merkmal der Systemarchitektur ist die Schnittstelle zu einer SPS von Siemens, mit das Bandumlaufsystem gesteuert werden kann (Figure 2). Diese abonniert Topics vom MQTT-Broker, wodurch das Band bei empfangener Nachricht beispielsweise gestoppt oder gestartet werden kann.

### NFC-Technologie

#### NFC – Near Field Communication

Smartphone / NFC-Leser • Aktives Gerät • Sendet + empfängt • 13.56 MHz Trägerfrequenz • Energiequelle

NFC-Tag / Chip • Passives Gerät • Empfängt Energie vom Leser • Reflektiert Signal (Backscatter) • Keine eigene Stromversorgung

Technische Merkmale: • Reichweite: < 10 cm • Frequenz: 13.56 MHz • Kommunikation: Peer-to-Peer, Lesen/Schreiben und Kartenemulation • Standard: ISO/IEC 14443 & 18092

### Sensor-Polling

Um Informationen erfolgreich auf einen NFC-Tag zu schreiben, muss sich der Tag für eine definierte Dauer im Erfassungsbereich des NFC-Sensors befinden und zuverlässig erkannt werden. Dazu wird ein Polling-Verfahren eingesetzt: Der Sensor überprüft kontinuierlich, ob ein Tag in Reichweite ist. Solange der Tag erkannt wird, ist ein Schreibvorgang möglich. Überschreitet die Erkennungsdauer eine durch die Variable INSCOPE\_TIME festgelegte Schwelle, wird der Vorgang als erfolgreich gewertet. In diesem Fall blockiert der Controller kurzzeitig die weitere Verarbeitung, um die Nachricht sicher an den MQTT-Broker zu übermitteln.

Das folgende Codebeispiel verdeutlicht das Prinzip des Pollings: Der NFC-Sensor wird in einer Schleife kontinuierlich abgefragt, um zu prüfen, ob sich ein Tag im Erfassungsbereich befindet.

```
void loop(void) {
    if (read_NFC_sensor()) {
        Serial.println("Tag gefunden");
    } else {
        Serial.println("auf einen Tag warten");
    }
    // Kurze Pause (1ms)
    delay(1);
}
```

### Visualisierung in Node-Red

Node-RED ist eine webbasierte, visuelle Entwicklungsumgebung für die verdrahtungsfreie Programmierung von Datenflüssen. Dieses Tool wurde verwendet, um ein graphisches Live-Tracking der Produkte auf dem Band zu ermöglichen.

- Anforderungen an das Dashboard (Figure 3):
  - Visuelle Abbildung der Stationen und des Bandumlaufsystems
  - Aktualisierung der Positionen der Produkte entlang des Bandes
  - tabellarische Auflistung der Produkte
- Gehalt der Tabelle:
  - UID: Identifikationsnummer eines Tags
  - tracking: Position auf dem Band
  - state: Zustand an der letzten Station
  - finished: im Hochregal eingelagert?

Das erstellte Dashboard bietet außerdem die Möglichkeit testweise Nachrichten an die MCU-Controller zu senden. Das wird über die integrierten MQTT-Blöcke in Node-Red ermöglicht.

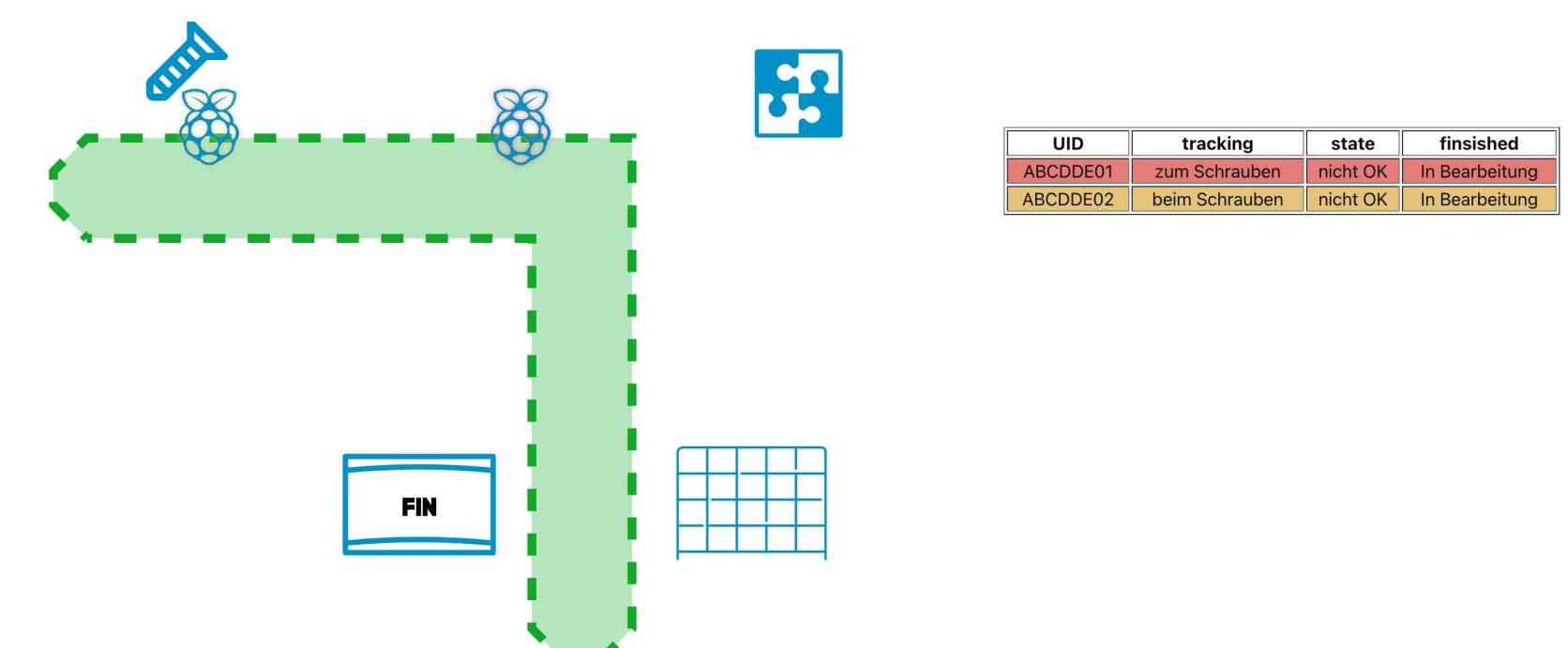


Figure 3: Visualisierung des Bandumlaufsystems in Node-Red