



Fakultät Informatik

Praktikumsbericht

im Studiengang Informatik

vorgelegt von

Jan-Eric Gedicke

Matrikelnummer 3653446

Erstgutachter: Prof. Dr. Trommler

Betreuer Un- Etling, Michael; Conrad, Maximili-
ternehmen: an; Baron, Dominik

© 2025

Inhaltsverzeichnis

1	Einleitung (evtl noch etwas umschreiben)	1
2	Vorstellung der Siemens AG (noch umschreiben)	2
3	Vorstellung meiner Fachabteilung	3
4	Vorstellung der SmartFactory@EH-Anlage	4
4.1	Station 1: Modulares Produktionssystem	6
4.2	Station 2: Abfüllung	6
4.3	Station 3: Quality Gate	8
4.4	Station 4: Kommissionierbahnen	8
4.5	Station 5: Eckstation	9
4.6	Station 6: Recycling	10
4.7	Netzwerkverbund	11
4.8	Begriffsklärungen	12
5	Projekt SmartFactory@EH-Anlage	15
5.1	Projekt Smart Factory-Anlage: Aufgabenstellung	15
5.2	Projekt Smart Factory-Anlage: Zielsetzung	15
5.3	Projekt SmartFactory-Anlage: Projektorganisation (geschrieben)	16
5.4	Projekt SmartFactory-Anlage: Projektverlauf	18
5.4.1	Auftaktwoche:	18
5.4.2	Hands-On SmartFactory-Anlage:	19
5.4.3	Bibliotheksverantwortlicher (erweiternder Aufgabenbereich):	19
5.4.4	Flaschenmanagment in der Unit:	20
5.4.5	Drehteller:	22
5.4.6	Kommunikation:	23
5.5	Projekt SmartFactory-Anlage: Ergebnis	23
6	Persönliches Fazit zum Praktikum (umschreiben)	25

1 Einleitung (evtl noch etwas umschreiben)

Dieser Praktikumsbericht dokumentiert mein Praxissemester bei der Siemens AG in der Fachabteilung DI CS SD EH. Als dualer Student mit vertiefter Praxis bei Siemens bin ich sowohl meiner Fachabteilung (DI CS SD EH FA 2), als auch der SPE (Siemens Professional Education), also einer Ausbildungsabteilung, zugewiesen. Mein Praxissemester umfasst daher zwei praktische Projekte, eines in meiner Abteilung und eines der Ausbildungsabteilung, welche ich begleiten durfte. In meiner Fachabteilung "Expert House" (DI CS SD EH), war ich Teil der Weiterentwicklung der Smart Factory-Anlage, einer für Studierende konzipierten Trainings- und Automatisierungsanlage zur Simulation einer Flaschenabfüllungsproduktion. Während meiner Zeit in der Ausbildungsabteilung habe ich das Projekt "BersiBot", die Programmierung eines Roboters mit KI-Bilderkennung und Webseitensteuerung, begleitet. Die spezifischen Aufgabenstellungen, Ziele, das Projektmanagement, der Projektverlauf und die erreichten Ergebnisse des Projekts „SmartFactory- Anlage“ sind Hauptgegenstand dieses Berichts. Auf das Projekt "BersiBot" werde ich in diesem Bericht nicht gesondert eingehen, da es gerade im Hinblick auf das Projektmanagement und die allgemeine Vorgehensweise sehr starke Überschneidungen beider Projekte gibt und die SmartFactory-Anlage das deutlich umfangreichere Projekt ist.

2 Vorstellung der Siemens AG (noch umschreiben)

Die Siemens AG wurde 1847 von Werner Siemens und Georg Halske als „Telegraphen Bau-Anstalt von Siemens und Halske“ in Berlin gegründet.¹ Anfangs konzentrierte sich das Unternehmen auf die Entwicklung und den Vertrieb von Telegrafengeräten, wodurch Siemens schnell zu einem bedeutenden Akteur in der Elektroindustrie wurde. In den darauffolgenden Jahren expandierte das Unternehmen nach Russland und England² und erschloss neue Geschäftsfelder wie die Elektrifizierung der Eisenbahn.³

In den 1930er Jahren erweiterte Siemens seine Produktpalette um elektrische Haushaltsgeräte, Wärme- und Heizungskonzepte sowie Medizintechnik.⁴ Heute ist Siemens in vielen Branchen vertreten, darunter Industrie, Gebäudetechnik, Energietechnik, Mobilität, Gesundheitstechnik, Business Administration und Human Resource Management. Die Schwerpunkte des Unternehmens liegen auf der Digitalisierung, insbesondere der digitalen Transformation von Fertigungsprozessen, Transportnetzwerken, Gebäuden und Energiesystemen.

Die Siemens AG beschäftigt weltweit rund 320.000 Mitarbeiter und erzielte im Geschäftsjahr 2022/23 einen Umsatz von 74,8 Milliarden Euro sowie einen Nettogewinn von 8,5 Milliarden Euro.⁵ Ein Jahr später, im Jahr 2024, lag der Umsatz bei 75,9 Milliarden Euro, was die anhaltende Stärke des Unternehmens unterstreicht.

¹Vgl. Ernst Klett Verlag: Infoblatt Siemens AG, 2004, <https://www.klett.de/alias/1036900>. Zugriff: 5. Januar 2025

²Vgl. Merkur: Siemens – Geschichte, Aktie und Tätigkeitsfelder, 2022, <https://www.merkur.de/wirtschaft/siemens-geschichte-aktie-und-taetigkeitsfelder-91268844.html>. Zugriff: 19. Januar 2024

³Vgl. Ernst Klett Verlag, 2004

⁴Vgl. Merkur, 2022

⁵Vgl. Siemens, 2023 zitiert nach de.statista.com, <https://de-statista-com.thn.idm.oclc.org/statistik/daten/studie/73827/umfrage/umsatz-von-siemens-seit-2005/>. Zugriff: 5. Januar 2025

3 Vorstellung meiner Fachabteilung

Ich bin in der Abteilung **Digital Industries Customer Service Service Delivery Expert House (DI CS SD EH)** tätig, die Teil des Kundenservices der Sparte Digital Industries der Siemens AG ist. Das Expert House bietet Siemens-Kunden Zugang zu fachlichen Spezialisten, die bei der Umsetzung spezifischer Projekte unterstützen. Innerhalb des Expert House existieren verschiedene Fachbereiche, darunter die Bereiche **Process Automation** und **Factory Automation**, letzterem bin ich zugeordnet.

Die **Factory Automation** umfasst ein breites Spektrum an Tätigkeiten wie die Programmierung und Inbetriebnahme von Automatisierungsanlagen für Kunden sowie die Entwicklung maßgeschneiderter Softwarelösungen. Aufgrund des hohen fachlichen Spezialisierungsgrads ist der Bereich in mehrere nummerierte Gruppen unterteilt. Meine Gruppe, **Factory Automation 2 (FA2)**, konzentriert sich auf Themen wie **Safety**, **Reliability** und **Technology Services**.

Wir stellen sicher, dass die Anlagen nicht nur den höchsten Sicherheitsstandards entsprechen, sondern auch zuverlässig und technologisch auf dem neuesten Stand sind. Dazu gehört die Überprüfung und Optimierung sicherheitsrelevanter Prozesse, die Analyse der Anlagenzuverlässigkeit sowie die Bereitstellung technologischer Beratungsdienste. Diese Expertise ermöglicht es uns, für Kunden individuelle Lösungen zu entwickeln, die höchsten Anforderungen an Sicherheit und Effizienz gerecht werden.

4 Vorstellung der SmartFactory@EH-Anlage

Die **SmartFactory-Anlage** ist eine speziell für Schulungszwecke konzipierte Automatisierungsanlage, die von der Abteilung **Expert House** der Siemens AG entwickelt wurde. Sie dient als Trainingsplattform für Auszubildende und Studierende, um praxisnahe Erfahrungen in der Automatisierungstechnik zu sammeln und mit einer Vielzahl von Produkten aus dem Siemens-Katalog vertraut zu werden. Die Anlage simuliert eine Flaschenabfüllungsproduktion im Kleinformat und wird zur Ausbildung sowie zum Testen von Prototypen genutzt.

Der modulare Aufbau der Anlage umfasst sieben verschiedene Teilmodule: die **Abfüllstation**, die **Qualitätskontrolle**, die **Kommissionierbahn**, die **Eckstation**, die **Recyclingstation**, den **Delta-Picker** und das **MPS-Modul**. Diese Stationen sind durch ein Kreislautsystem miteinander verbunden, wobei die Flaschen von der Abfüllstation über die verschiedenen Stationen transportiert werden. Im Kreislautsystem werden die Flaschen am Ende des Prozesses, nach dem Recycling, wieder zur erneuten Verarbeitung bereitgestellt. Dazu gehören das Befüllen von Flaschen und das Entleeren von Transportbändern.

Jedes Modul der Anlage hat einen spezifischen Schwerpunkt und ist mit ähnlicher Hardware ausgestattet, einschließlich einer **Siemens-Simatic S7-1500 Speicherprogrammierbaren Steuerung** (SPS), außer dem Quality-Gate, bei dem ältere Steuerungstechnik mit einer ET200Sp verbaut ist, einem **Siemens HMI** (Human-Machine Interface) für Statusmeldungen und Benutzereingaben sowie Transportbändern. Zudem sind an einigen Stationen pneumatisch betriebene Greifer und RFID-Schreib-Lesegeräte verbaut, die eine präzise Steuerung und Überwachung der Anlage ermöglichen. Die Anlage läuft iterativ, was bedeutet, dass nach Abschluss des Kreislauts die Flaschen, Deckel und Kugeln für die erneute Abfüllung bereitgestellt werden.

Die Anlage kann über drei Betriebsmodi gesteuert werden: **Einrichtungsbetrieb**, **Handbetrieb** und **Automatikbetrieb**, sowie zwei Betriebsarten: **Modularbetrieb** und **Inselbetrieb**. Im **Einrichtungsbetrieb** kann der Bediener alle Anlagenteile uneingeschränkt bewegen, auch wenn dies zu Beschädigungen führen könnte. Der **Handbetrieb** ermöglicht es, einzelne Funktionen auszuführen, wie das Befüllen einer Flasche oder das Entleeren eines Bandes. Im **Automatikbetrieb** läuft die Anlage vollständig automatisiert und erfordert nur im Fehlerfall, zum Anlegen eines Auftrags oder bei einer Bedieneranforderung, einen Bediener. Eine Bedieneranforderung wäre hier zum Beispiel das manuelle Entleeren der Kommissionierbahn oder das Entleeren eines Deckelausschussbands.

Die Funktionsweise der einzelnen Stationen im **Automatikbetrieb** ist wie folgt: In der **Abfüllstation** werden die Flaschen mit Kugeln befüllt, bevor sie in die nächsten Stationen weitergeleitet werden. In der **Qualitätskontrolle** wird die Qualität der befüllten Flaschen geprüft. Danach erfolgt in der **Kommissionierbahn** und der **Eckstation** die Sortierung der Flaschen sowie eine weitere Qualitätsprüfung. Der Kreislauf endet in der **Recyclingstation**, in der die Flaschen entleert und die Kugeln in die richtigen Behälter sortiert zurückgeführt werden.

Durch diese modulare und vielseitige Struktur bietet die **SmartFactory-Anlage** den Auszubildenden eine umfassende Ausbildung und ein realistisches Trainingsumfeld für die Arbeit mit modernen Automatisierungssystemen.

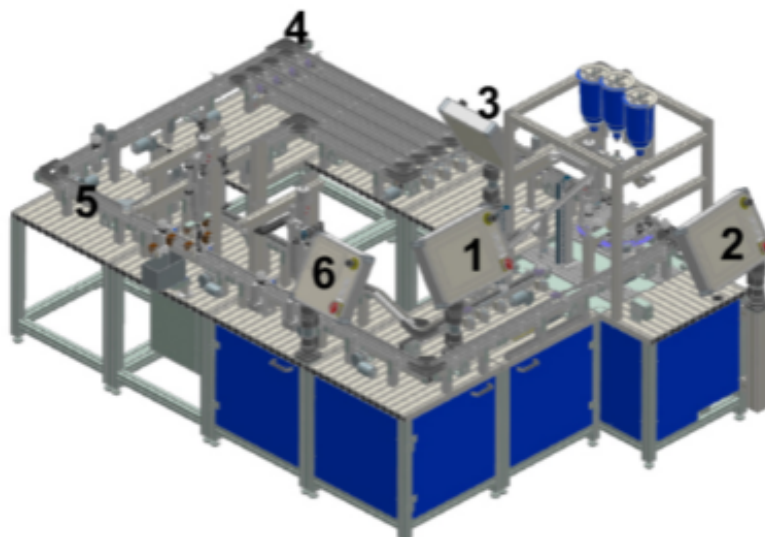


Abbildung 4.1: Stationen der SmartFactory-Anlage[2]

Abbildung 1 zeigt ein Modellbild der gesamten Anlage und veranschaulicht die räumliche Anordnung sowie die Verknüpfung der einzelnen Stationen. In den folgenden Abschnitten werden die Funktionsweisen der jeweiligen Module detailliert beschrieben.

4.1 Station 1: Modulares Produktionssystem

Das Modulare Produktionssystem (MPS) ist eine zentrale Komponente der SmartFactory-Anlage und dient der Bereitstellung von Flaschen und Deckeln für die Abfüllstation. Es nimmt Deckel von der Recycling-Anlage entgegen, überprüft diese mithilfe eines RFID-Lesegeräts und eines Farbsensors auf Lesbarkeit und Farbe und sortiert fehlerhafte oder andersfarbige Deckel, wie rote, aus.

Auf Anforderung der Abfüllstation liefert das MPS die benötigte Anzahl an Flaschen und setzt mithilfe eines Greifers die passenden weißen Deckel auf die befüllten Flaschen. Fehlerhafte Flaschen oder Deckel werden aussortiert, um die Produktionsqualität zu sichern. Das System überwacht kontinuierlich den Materialvorrat, um Unterbrechungen zu vermeiden, und stellt relevante Daten über ein HMI bereit.

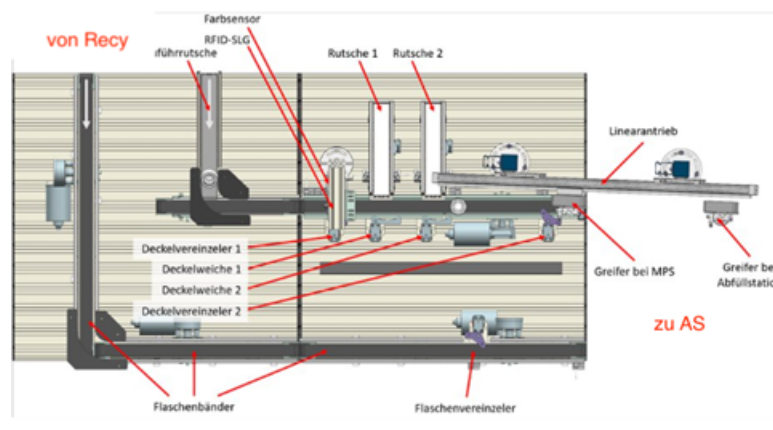


Abbildung 4.2: Modulares Produktionssystem[2]

4.2 Station 2: Abfüllung

Die Abfüllstation ist der Start des Kreislaufs der Anlage und hat die Aufgabe, Flaschen mit einer vorgegebenen Anzahl von Kugeln zu befüllen. Diese Anzahl

wird durch die Erstellung von Aufträgen am Bedienpanel (HMI) oder Remote über Open Platform Communications Unified Architecture (OPC UA) auf einem Webinterface festgelegt. Hier kann zum Beispiel der Prozentsatz der Kugelfarbe (bei einem Maximum von 250 Kugeln) oder ein Absolutwert für die Anzahl der Kugeln festgelegt werden. Für das Abfüllen, hat die Abfüllung drei Container mit jeweils, roten, blauen und gelben Kugeln. Außerdem kann die Anzahl der abzufüllenden Flaschen angegeben werden. Diese Anzahl wird per “On-Demand”-TCP-Verbindung an das modulare Produktionssystem gesendet. Die danach von dem modulare Produktionssystem gesendeten Flaschen werden durch einen Drehteller an verschiedenen Verarbeitungsstationen bewegt. Als erstes wird die Flasche, mit der des Auftrages angegebenen Anzahl an Kugeln, an der ersten Verarbeitungsstationen befüllt (Rote 1, 4.3). Dann wird in Verarbeitungsstationen 2 (Rote 2, 4.3) ein Deckel von dem modulare Produktionssystem angefordert und von diesem auf die Flasche gesetzt. Der Deckel wird danach in Verarbeitungsstationen 3 (Rote 3, 4.3) mit einen drehbaren Greifer verschraubt und an der vierten Verarbeitungsstationen (Rote 4, 4.3) wird der Deckel mit einem RFID-Tag beschrieben. Der RFID-Tag enthält Datum und Uhrzeit der Abfüllung, sowie welcher Auftrag abgefüllt wurde. Jeder dieser Schritte soll parallel ausgeführt werden, um die Produktionszeit zu minimieren.

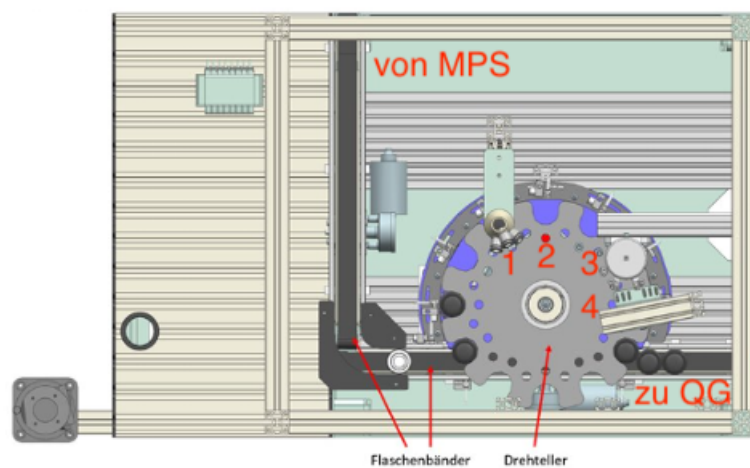


Abbildung 4.3: Abfüllstation[2]

4.3 Station 3: Quality Gate

Das Quality Gate soll die Auftragsdaten auf dem RFID-Tag mit dem Inhalt der Flasche abgleichen. Hierzu wird an der ersten Verarbeitungsstationen der RFID-Tag der Flasche eingelesen und an die zweite Verarbeitungsstationen soll ein Farbsensor die farbliche Zusammensetzung prüfen. Sollte die farbliche Zusammensetzung übereinstimmen, wird an der dritten Verarbeitungsstationen der RFID-Tag der Flasche mit „Gut“ beschrieben, sollte sie aber nicht übereinstimmen wird der RFID-Tag mit „Ausschuss“ beschrieben. Jede Verarbeitungsstationen hat dafür einen eigenen Vereinzelter, um die richtige Verarbeitung der Flaschen zu gewährleisten. Der Anlagenteil dient also zur Qualitätssicherung der abgefüllten Flaschen.

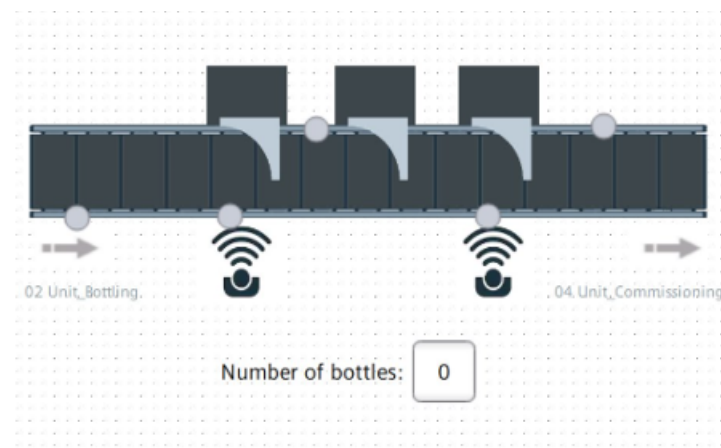


Abbildung 4.4: Quality Gate[2]

4.4 Station 4: Kommissionierbahnen

Die Kommissionierbahnen sollen die vom Quality-Gate ankommenden Flaschen nach Auftragsnummer sortieren, hierzu wird zu Beginn der RFID-Tag gelesen und die Flasche auf eine von vier Bahnen sortiert. Durch das Quality-Gate als fehlerhaft markierte Flaschen oder Flaschen ohne lesbaren RFID-Chip werden direkt über ein Ausschussband an die nächste Station weitergegeben. Sollte die Flasche „In Ordnung“ sein, wird sie auf eine von vier Bändern sortiert. Die sortierten Flaschen sollen so lange in der Station gehalten werden, bis der entsprechende Auftrag abgeschlossen wurde oder der Bediener ein manuelles Entleeren der Bänder anfordert.

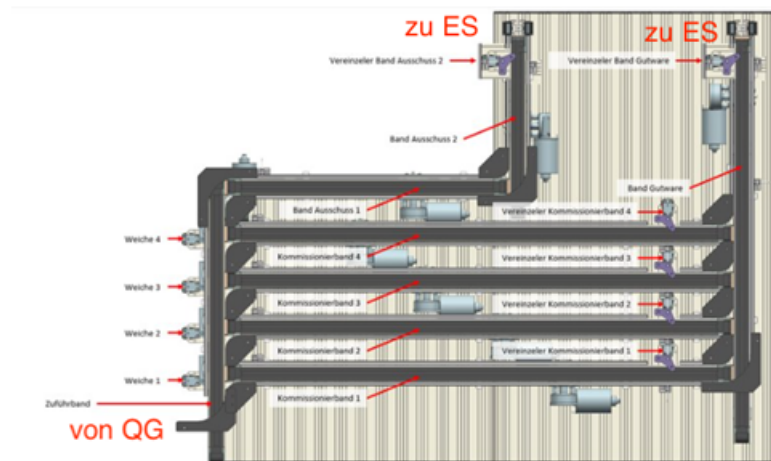


Abbildung 4.5: Kommissionierbahnen[2]

4.5 Station 5: Eckstation

Die Eckstation dient als Übergangspunkt, an dem die Flaschen von der Kommissionierbahn zur Recyclingstation weitergeleitet werden. Die Besonderheit der Eckstation ist, zwei Laufbänder, die wie hochklappbare Brücken fungieren, eine für die Abhandlung von Aufträgen und eine für aussortierte Flaschen. Das Hochklappen ermöglicht es, einen Schaltschrank in der Mitte der Anlage zu erreichen. Das Hochklappen der Laufbänder soll unterbunden werden, wenn noch Flaschen auf dem Laufband sind. Dies erfordert das Tracken der Flaschenanzahl und einen Stopper, um den Zulauf der Flaschen bei hochgeklappten Laufbändern zu verhindern und das Hochklappen bei vollem Laufband zu verhindern. Ist das Laufband voll, wird der Zulauf durch den Stopper verhindert. Des Weiteren ist es möglich, ein entsprechendes Anforderungssignal an die Eckstation, zum Anheben der Brücke, zu senden, auf dessen Quittierung Flaschen warten müssen. Die Station übernimmt keine weiteren Funktionen und dient ausschließlich dazu, die Flaschen zwischen den beiden Stationen zu verfahren und Zugang zum Einschaltschrank zu gewährleisten.

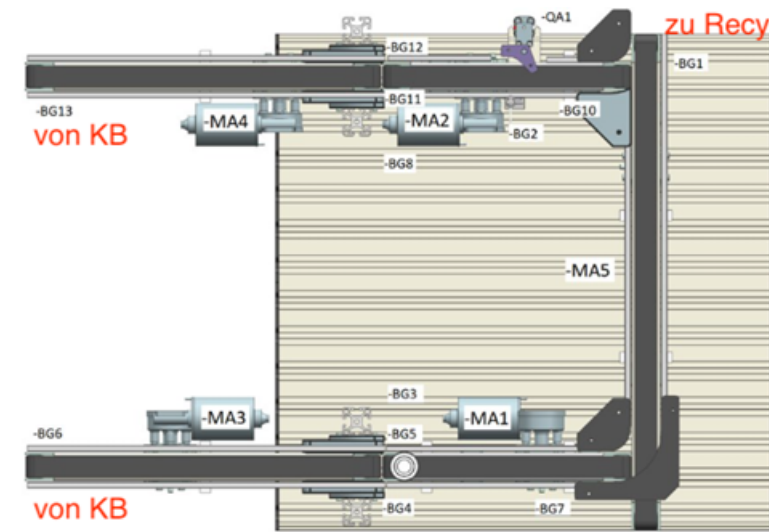


Abbildung 4.6: Eckstation[2]

Abbildung 5 zeigt den Aufbau der Eckstation. Die Kontaktpunkte für die versenkbaren Brücken befinden sich an den Sensoren BG4, BG5, BG11 und BG12. Diese Sensoren detektieren, ob die Brücken richtig in die ursprüngliche Position verfahren wurden.

4.6 Station 6: Recycling

Die Recyclingstation übernimmt eine entscheidende Rolle im geschlossenen Produktionskreislauf der SmartFactory-Anlage. Sie verarbeitet alle ankommenden Flaschen, unabhängig davon, ob diese aus erfolgreich abgeschlossenen Produktionsaufträgen stammen oder aufgrund von Fehlern aussortiert wurden.

Zu Beginn wird jede Flasche von einem **RFID-Reader** erfasst, der ihre Identität und den Zustand überprüft. Diese Informationen werden an die Steuerung weitergeleitet, um den weiteren Verarbeitungsprozess zu optimieren. Anschließend gelangt die Flasche zum **Deckelentferner**, wo ein präziser Greifer den Flaschendeckel entfernt. Der Deckel wird auf ein separates Laufband transportiert, das die Deckel für eine spätere Verarbeitung zur Verfügung stellt (siehe Station 1 in Abbildung 4.7).

Nach der Entfernung des Deckels fährt die Flasche weiter zum **Entleerer**. Hier wird sie von einem Greifer angehoben und über einen Trichter positioniert, wo die

Kugeln entleert werden (siehe Station 2 in Abbildung 4.7). Der Trichter ist mit einem **Farbsensor** ausgestattet, der die Farbe der Kugeln erkennt. Diese Information wird an das Steuerungssystem weitergegeben, um die Kugeln entsprechend zu sortieren.

Das Rückführungssystem für die Kugeln wird **pneumatisch** betrieben. Je nach erkannter Farbe wird der passende Abfüllschlauch aktiviert, durch den die Kugeln in den entsprechenden Container an der Abfüllstation geschossen werden. Dies gewährleistet eine exakte farbliche Trennung und ermöglicht die Wiederverwendung der Kugeln im nächsten Produktionszyklus.

Während die erste Flasche entleert wird, bereitet das System die nächste Flasche für den Deckelentfernungsprozess vor, um eine hohe Effizienz sicherzustellen. Nach dem Entleeren wird die Flasche wieder auf das Laufband gesetzt und zur **modularen Produktionsstation (MPS)** weitergeleitet. Die MPS macht die Flasche anschließend für den nächsten Produktionsprozess verfügbar.

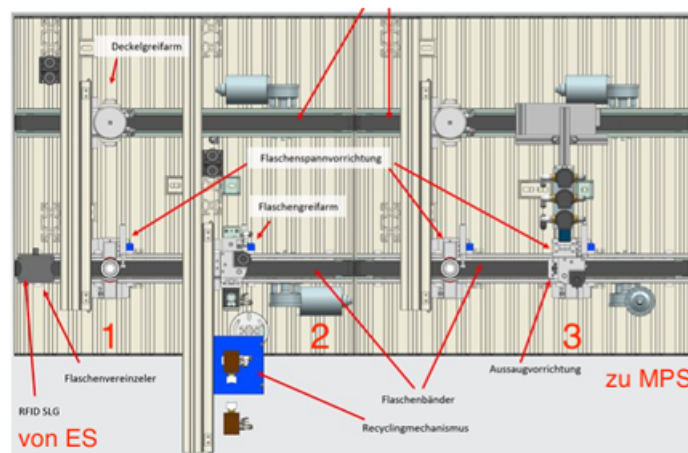


Abbildung 4.7: Recyclingstation[2]

4.7 Netzwerkverbund

Alle Anlagenteile sind mit Ethernet mit der Einspeisung modular verbunden und bilden damit ein Subnetz. Dies ermöglicht eine effiziente und stabile Kommunikation zwischen den verschiedenen Stationen der Anlage und ermöglicht das senden

und empfangen von Daten. Die gesamte Kommunikation der Anlage erfolgte “On-Demand” und passiert dadurch nur wenn sie von einer anderen Station angefordert wird.

4.8 Begriffsklärungen

Speicherprogrammierbaren Steuerungen (SPS):

Im nachfolgenden Abschnitt wird das programmieren von speicherprogrammierbaren Steuerungen (SPS) mehrfach thematisiert. Dabei handelt es sich um die Entwicklungsumgebung für alle Automatisierungsprodukte von Siemens [1].

Eine speicherprogrammierbare Steuerung (SPS) ist ein digitales Gerät, das zur Steuerung von Maschinen und Anlagen eingesetzt wird. Die Prozesslogik für speicherprogrammierbare Steuerung (SPS) lässt sich in verschiedenen Programmiersprachen umsetzen, darunter FUP (Funktionsplan), AWL (Anweisungsliste) und SCL (Structured Control Language). Für die SmartFactory-Anlage wurde festgelegt, die Programmierung bedarfsgerecht in FUP und SCL durchzuführen.

Da eine PLC (Programmable Logic Controller) als Echtzeitsystem zyklisch arbeitet, sie führt Berechnungen durch und aktualisiert Ausgaben in festen Intervallen, sind bei der Programmierung spezielle Anforderungen zu beachten. Um diesen Anforderungen gerecht zu werden, stehen verschiedene Konzepte zur Verfügung:

- **Funktion:** Ähnlich wie eine Methode in C#, ermöglicht eine Funktion die Verarbeitung von Übergabeparametern und die Berechnung entsprechender Ausgabewerte. Allerdings sind diese Ausgabewerte nicht persistent, da sie in jedem Programmzyklus neu berechnet werden.
- **Funktionsbaustein:** Ein Funktionsbaustein bietet dieselben Möglichkeiten wie eine Funktion, jedoch mit der zusätzlichen Fähigkeit, Daten über mehrere Zyklen hinweg zu speichern.
- **Datenbaustein:** Ein Datenbaustein dient der persistenten Speicherung von für das Programm relevanten Daten. Wird ein Funktionsbaustein erstellt, wird automatisch ein zugehöriger Datenbaustein zur Speicherung seiner Daten angelegt.

- **User Defined Datatype (UDT):** Ein UDT ist ein benutzerdefinierter Datentyp, der individuell vom Programmierer erstellt werden kann.

Konzeptionierung nach ISA-88:

Die Software der Anlage wurde gemäß dem ISA-88-Standard konzipiert. Dieser Standard legt fest, dass die Software in wiederverwendbare Module für Bauteile und Baugruppen strukturiert wird. Die Module wurden in die folgenden Typen unterteilt:

- **Control Modul:** Ein Control Modul repräsentiert einzelne Bauteile, die an jeder Station der Anlage vorkommen. Ein typisches Beispiel ist ein Sensor. Obwohl an der Anlage verschiedene Sensortypen verbaut sind, teilen sie alle die grundlegende Funktion, ein Bit umzuschalten. Das Control Modul `CM_BinarySensor` soll so entwickelt werden, dass es universell für alle Sensortypen einsetzbar ist, wobei lediglich die Hardwareadresse des jeweiligen Sensors angegeben werden muss.
- **Equipment Modul:** Equipment Module bestehen aus einer Kombination mehrerer Control Module und bilden komplexere Baugruppen ab. Beispielsweise sind an den Förderbändern der SmartFactory-Anlage neben Sensoren auch Pneumatikzylinder verbaut. Da die Anzahl der Control Module pro Förderband variieren kann, wurde das Equipment Modul `EM_Conveyor` so gestaltet, dass es eine beliebige Anzahl an Control Modulen unterstützt und flexibel konfigurierbar ist.
- **Unit:** Die Unit bildet die spezifische Prozesslogik eines Anlagenmoduls ab und implementiert die erforderlichen Programmabläufe.

Die Struktur eines nach ISA-88 entworfenen Systems ist hierarchisch organisiert. In diesem Aufbau kommuniziert eine Unit ausschließlich mit Equipment Modulen, während Equipment Module sowohl mit Units als auch mit Control Modulen interagieren. Control Module hingegen stehen nur in Verbindung mit Equipment Modulen.

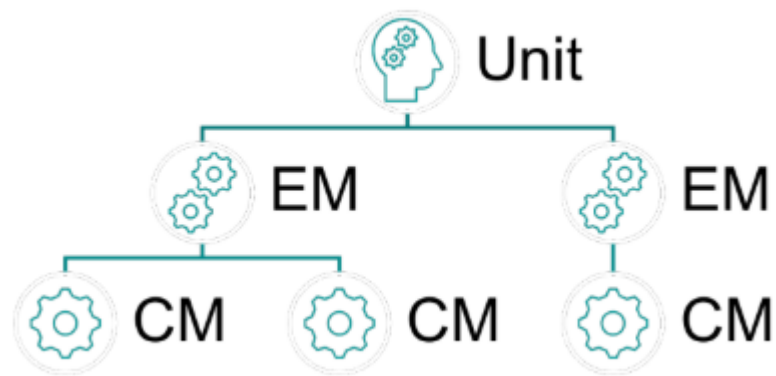


Abbildung 4.8: ISA-88 Modulhierarchie[2]

5 Projekt SmartFactory@EH-Anlage

5.1 Projekt Smart Factory-Anlage: Aufgabenstellung

Die SmartFactory-Anlage ist eine speziell für Schulungszwecke konzipierte Automatisierungsanlage, die von der Abteilung Expert House entwickelt wurde. Sie beinhaltet verschiedene Produkte aus dem Siemens-Produktkatalog wie zum Beispiel speicherprogrammierbare Steuerungen, Lichtsensoren und Motoren. Meine Aufgabe bestand darin, zusammen mit 18 weiteren Studierenden die Weiterentwicklung für diese Anlage durchzuführen. Dies umfasste sowohl die vollständige Ausbesserung von und Behebung bereits bekannter Mängel, welche in einer List of Open Points (LOP) festgehalten wurden, als auch das Testen der Robustheit der Software. Die Software wurde nach dem ISA-88-Standard umgesetzt und alle Erweiterungen sollen auch nach diesem hinzugefügt werden. Zur Programmierung war die Siemens-Automatisierungssoftware „TIA-Portal“ zur Verwendung vorgegeben. Eine weitere Anforderung war es, die umfassenden Dokumentationsunterlagen in Form von Word-Dokumenten für nachfolgende Studierende auszubessern und gegebenenfalls mit den neuen Funktionen zu erweitern. Die Nutzung des ISA-88-Standard soll gewährleisten, dass die Anlage ohne einen Ansprechpartner aus unserem Projektteam weiterentwickelbar ist.

5.2 Projekt Smart Factory-Anlage: Zielsetzung

Das Ziel am Ende des Praxissemesters war es, die funktionsfähige SmartFactory-Anlage weiterzuentwickeln, ihre Robustheit zu erhöhen und uns eine Grundlage für Ausbildungs- und Weiterbildungsmodule zu schaffen. Dafür müssen für jedes Anlagenmodul bekannte Fehler und offene Punkte aus der List of Open Points (LOP) abgearbeitet werden, sowie bei der Erkennung neuer Fehler, diese zur List

of Open Points (LOP) hinzugefügt werden. Des Weiteren müssen die Bedingungen wie eine einheitliche HMI-Schnittstelle und eine leichte Bedienung der Anlage umgesetzt werden.

5.3 Projekt SmartFactory-Anlage: Projektorganisation (geschrieben)

Das Projektteam besteht aus ... Informatikstudenten und ... Elektro- und Informationstechnikstudenten sowie drei Fachbetreuern. Diese haben uns als “Product Owner” im Projekt unterstützt und geholfen.

Das gesamte Projekt wurde auf der agilen Methode **Scrum** basierend durchgeführt. Dabei wurden jedoch Anpassungen vorgenommen, die dem Projektablauf und den spezifischen Umständen gerecht wurden. Wie bereits in Abschnitt 1, „Einleitung“, erläutert, wurde die Arbeitsphase im Expert House sowohl bei uns Informatikern als auch bei den Elektro- und Informationstechnikern durch die **SPE-Phase** unterbrochen, allerdings zu unterschiedlichen Zeitpunkten. Dies führte zu einem geteilten Arbeitszeitplan: In den ersten acht Wochen arbeiteten alle gemeinsam am Projekt, in den darauffolgenden acht Wochen nur die Elektro- und Informationstechniker, und in den abschließenden acht Wochen waren ausschließlich wir Informatiker beteiligt.

Trotz der Abweichungen von den klassischen Scrum-Praktiken war dieser Ansatz sinnvoll. Durch die iterative Arbeitsweise und die klare Struktur von Scrum konnten wir den Überblick über die aktuellen Aufgaben besser behalten. Zur Organisation der Aufgaben und Projektmeilensteine half uns insbesondere ein **Kanban-Board**, das die Aufgaben nach den Kategorien strukturierte. Dieses visuelle Hilfsmittel erleichterte die Nachverfolgbarkeit von Fortschritten erheblich und förderte die Verständlichkeit während der regelmäßigen Meetings.

Die Kombination aus Scrum-Elementen und der Verwendung des Kanban-Boards sorgte für eine verbesserte Teamkommunikation und einen reibungsloseren Workflow. So konnte jeder Beteiligte stets nachvollziehen, welche Aufgaben anstanden, bearbeitet wurden oder bereits abgeschlossen waren. Besonders in der finalen Phase, als die Teams unabhängig voneinander arbeiteten, erwies sich diese Arbeits-

weise als vorteilhaft, da sie die Selbstorganisation förderte und klare Prioritäten setzte.

Dies führte dazu, dass an den Übergabepunkten große und kleine Änderungen klar an das andere Projektteam übergeben werden mussten. Gleichzeitig war es essenziell, die Dokumentation stets auf dem aktuellen Stand zu halten, um eine reibungslose Weiterarbeit zu ermöglichen.

Dadurch stimmten wir uns gemeinsam, Betreuer und Auszubildende, in Daily-Meetings ab. Wir arbeiteten in Zwei-Wochen-Sprints, die jeweils mit einer Vorstellung der Ergebnisse an die Stakeholder endeten. Diese Präsentation ist ein wichtiges Mittel um unseren Fortschritt an der Anlage zu zeigen und Punkte für die nächste Bearbeitung festzulegen. Eine zentrale Rolle spielte die "List of Open Points"(LOP), die zur Dokumentation von bestehenden Fehlern und Mängeln genutzt wurde.

Die LOP war in Dringlichkeitsstufen unterteilt:

- **Sehr wichtig:** Prozessbeendende Fehler oder solche, die mechanische Schäden verursachen könnten.
- **Mittel:** Fehler, die den Prozess nicht vollständig stoppen, aber zu Fehlern in der Verarbeitung von Flaschen führen.
- **Niedrig:** Fehler mit geringeren Auswirkungen, z. B. das Herausfallen einer Kugel bei der Initialisierung der Abfüllstation.

Die Stakeholder legten diese Prioritäten fest. Für jeden Sprint suchten sich die Verantwortlichen der jeweiligen Station die wichtigsten Aufgaben mit der höchsten Priorität heraus und arbeiteten diese ab.

Gerade zu bearbeitende Aufgaben und bereits abgeschlossene wurden in ein Kanban-Board eingetragen. Dies half, den Fortschritt zu visualisieren und machte in den Daily-Meetings deutlich, welche Aufgaben eventuell vom Plan abwichen. Dadurch konnten notwendige Anpassungen schnell besprochen und vorgenommen werden, um weiterhin das Projektziel in der gegebenen Zeit zu erreichen.

Das Kanban-Board wurde in folgende Bereiche unterteilt:

- **Aufgaben:** Hier befinden sich alle Aufgaben, die im aktuellen Sprint erledigt werden müssten, mit denen sich aber noch niemand befasst hat.

- **In Bearbeitung:** Hier wurden die aktuell bearbeiteten Aufgaben gesammelt.
- **Internes Review:** Wurde eine Aufgabe abgeschlossen, wurde das Ergebnis zunächst durch einen weiteren Studierenden korrekturgelesen und auf Verständlichkeit überprüft.
- **Ready for Review:** Eine von einem Studierenden als in Ordnung befundene Aufgabe wird noch ein weiteres Mal durch einen Betreuer geprüft.
- **Nacharbeiten:** Wurde eine durchgeführte Aufgabe entweder von einem Betreuer oder einem Studierenden als nicht in Ordnung befunden, oder es traten im Projektverlauf Probleme damit auf, ist sie in diesem Bereich zu finden.
- **Erledigt:** In diesem Bereich befinden sich schließlich komplett abgeschlossene Aufgaben.

Des Weiteren haben wir mit dem Siemens-Multiuser-Server gearbeitet, eine speziell angepasste Lösung für die Zusammenarbeit an Automatisierungsprojekten. Das bedeutet, dass jeder Studierende eine lokale Instanz des Projekts hatte, diese bearbeiten und an der Anlage testen konnte. Große Änderungen konnten dann auf einen GIT-ähnlichen Server hochgeladen werden, von dem alle anderen Instanzen diese Änderungen wieder herunterladen konnten. Außerdem konnte man über ein File-Locking-System bei Dateien, an denen man gerade arbeitete, festlegen, ob ein Konflikt besteht, wenn zwei Studierende gleichzeitig an derselben Datei arbeiten. Dies hat "Merge-Konflikte" und den Verlust von Code verhindert, die Zusammenarbeit erleichtert und ermöglicht eine grundlegende Versionskontrolle.

5.4 Projekt SmartFactory-Anlage: Projektverlauf

5.4.1 Auftaktwoche:

Noch vor dem offiziellen Start des Projekts wurde in meiner Abteilung eine Auftaktwoche organisiert, die dazu diente, uns auf die bevorstehenden Aufgaben vorzubereiten. Der Schwerpunkt lag darauf, unser Wissen und unseren Umgang mit dem TIA Portal aufzufrischen. Obwohl wir als dual Studierende bereits einen

zweiwöchigen Kurs zur Nutzung der Software absolviert hatten, fehlte uns durch ein Semester ohne praktische Anwendung die nötige Routine. In dieser Woche bestand unsere Aufgabe darin, die Steuerung eines Aufzugs mit fünf Stockwerken zu programmieren. Dabei wurden uns auch Programmierkonzepte wie die „State Machine“ vermittelt, die später eine wichtige Rolle bei der Entwicklung der SmartFactory-Anlage spielten.

5.4.2 Hands-On SmartFactory-Anlage:

Nach der Auftaktwoche wurden wir zufällig in Teams von zwei bis drei Personen eingeteilt, um die SmartFactory-Anlage in Betrieb zu nehmen. Ziel war es, am Ende der Woche eine Flasche, mit dem bereits bestehenden Code, vollständig durch die Anlage zu führen. Jedem Team wurde dabei eine spezifische Station zugewiesen. Ich war für die Abfüllstation zuständig. Die Aufgabenstellung war bewusst vage gehalten, um uns die Möglichkeit zu geben, die Anlage eigenständig kennenzulernen. Das Ergebnis bei der Abfüllstation bestand darin, das angefragte Flaschen und Deckel richtig für erstellte Aufträge verwendet wurden und ein Betriebsbeendender des ersten Laufbandes behoben wurden. Allerdings wurden auch einige neue Punkte, wie die fehlerhafte Kalibrierung des Drehtellers oder die fehlende Initialisierungsfahrt, in die LOP (List of Open Points) nachgetragen. Nach Abschluss dieser Aufgabe führten wir ein „Lessons Learned“ durch. Die wichtigste Erkenntnis hierbei war, dass die Anlage ohne einen sorgfältig erarbeiteten Plan und ein durchdachtes Konzept nicht effektiv in Betrieb genommen werden kann.

5.4.3 Bibliotheksverantwortlicher (erweiternder Aufgabenbereich):

Mir wurde ein erweiterter Aufgabenbereich in Form des Bibliotheksverantwortlichen zugewiesen. Meine Aufgabe bestand darin, die Projektbibliothek des Multiuser-Servers konsistent zu halten. Dies ist notwendig, da Programmbausteine gemäß dem ISA-88-Standard, wie beispielsweise *Control Modules* oder *Equipment Modules*, von mehreren Instanzen im Projekt genutzt werden können.

Ein Lichtsensor kann beispielsweise Teil eines Förderbands oder eines Drehtellers sein. Um doppelten Code zu vermeiden und die Erweiterbarkeit so hoch wie möglich zu halten, wird im Code, wenn ein Lichtsensor benötigt wird, der Baustein

aus der Projektbibliothek verwendet. Dieser Baustein wird im Projekt als neue Instanz des Bausteins aus der Projektbibliothek eingefügt. Möchte man den genannten Lichtsensor aktualisieren, werden alle Instanzen im Projekt sowie in der Projektbibliothek aktualisiert.

Da die Projektbibliothek versionsabhängig arbeitet, kann es vorkommen, dass jemand am Förderband eine Änderung vornimmt, die nicht mit der neuen Version des Lichtsensors kompatibel ist. Dies würde zu einem Ausfall der Lichtsensoren am Förderband führen. Ebenso könnte es passieren, dass auf dem Förderband noch eine ältere Version des Lichtsensors ohne die aktuellen Änderungen verwendet wird. Dies führt zu Inkonsistenzen, die von mir bereinigt werden mussten.

5.4.4 Flaschenmanagement in der Unit:

Das Flaschenmanagement der Abfüllstation wurde so umgesetzt, dass für jeden angeforderten Auftrag die benötigte Anzahl an Flaschen vom modularen Produktionssystem bereitgestellt wird. Diese Flaschen werden anschließend befüllt und mit Deckeln verschlossen. Ein Problem trat jedoch auf, wenn nicht genügend Flaschen nachgeliefert wurden. Beispielsweise konnten von 12 angeforderten Flaschen nur 9 ankommen, wodurch die Abfüllung anhielt und auf die Ankunft einer neuen Flasche an Position 1 des Drehtellers wartete (siehe Abbildung 5.1).

Um den Arbeitsfluss kontinuierlich aufrechtzuerhalten, wurde in Absprache mit den Betreuern eine Anpassung in der Unit der Abfüllstation vorgenommen. Diese Änderung sorgt dafür, dass, falls innerhalb von sieben Sekunden keine neue Flasche geliefert wird, die verbliebenen Flaschen, die sich bereits im Drehteller befinden und einem Auftrag zugewiesen wurden, dennoch abgefüllt und verschlossen werden. Dies gewährleistet eine höhere Effizienz und verhindert lange Stillstandszeiten der Abfüllstation.

Ein weiteres Problem betraf die Speicherung der Auftragsdaten. Diese wurden ursprünglich nicht remanent gespeichert, also nicht über einen PLC-Stop hinaus erhalten. Dies führte dazu, dass nach jedem Neustart oder Kaltstart der Steuerung alle Flaschen aus dem Drehteller entfernt werden mussten, bevor ein neuer Auftrag angelegt werden konnte. Dieses Verhalten widersprach dem Ziel, die Anlage auch für unerfahrene Bediener einfach und benutzerfreundlich zu gestalten.

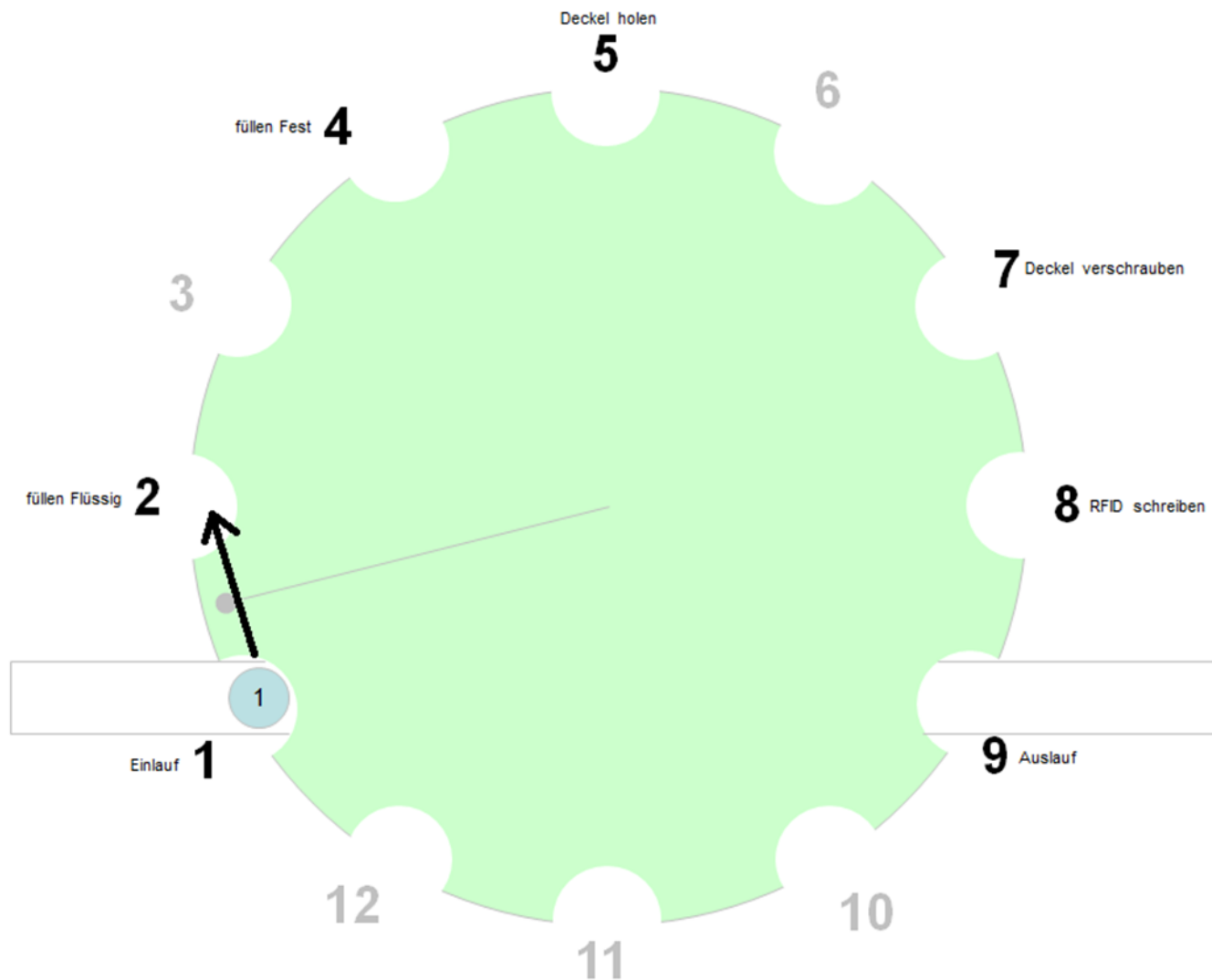


Abbildung 5.1: Drehteller der Abfüllanlage[2]

Der Drehteller erkannte nach einem Neustart weiterhin, dass sich Flaschen an den einzelnen Positionen (1–9) befanden. Dies führte dazu, dass der Drehteller in einen ungültigen Zustand geriet, da er für jede Position, an der eine Aufgabe auszuführen war, fehlerhafte Zustandsinformationen erhielt. Beispielsweise müssen an Position 4 die Flaschen befüllt, an Position 5 die Deckel angefordert, an Position 7 die Deckel verschraubt und an Position 8 die RFID-Tags beschrieben werden. Ohne korrekte Zustandsinformationen konnte der Drehteller diese Aufgaben nicht ausführen, was die gesamte Anlage blockierte. Nach der vorgenommenen Änderung wird der Drehteller darauf hingewiesen, dass sich unbekannte Flaschen, also Flaschen ohne Auftrag, im Drehteller befinden. Diese Flaschen werden nun automatisch aus dem Drehteller entfernt und an das Quality Gate weitergeleitet.

Außerdem erlaubt es die Abfüllstation, Aufträge zu bearbeiten oder gar zu löschen, da eine Liste aller angelegten Aufträge angelegt ist. Allerdings war es nicht möglich, einen Auftrag zu löschen, wenn dieser bereits in Bearbeitung war. Nun ist es möglich auch aktuelle Aufträge zu löschen.

5.4.5 Drehteller:

subsectionDrehteller: **Problem:** Die Knöpfe zur Steuerung des Drehtellers waren nicht korrekt eingestellt. Dies führte dazu, dass bei der Kalibrierung des Drehtellers ein Wechsel zwischen „drehe rechts“ und „drehe links“ nicht möglich war und zu einem nicht quittierbaren Fehler an der Abfüllstation führte. Beispielsweise drehte sich der Drehteller bei Betätigung des Knopfes „drehe rechts“ zwar in die gewünschte Richtung, setzte seine Bewegung jedoch auch nach dem Loslassen des Knopfes fort. Dies widerspricht den Anforderungen an Sicherheit, Benutzerfreundlichkeit und Robustheit der Anlage.

Herausforderung: Eine große Herausforderung bei dieser Aufgabe war es, die Fehlerursache zu identifizieren. Zunächst vermuteten wir, dass die Kommandos zum Anhalten des Drehtellers nicht korrekt gesendet wurden. Allerdings stellte sich heraus, dass die Kommandos nicht nur fehlerhaft gesendet wurden, sondern gar nicht implementiert waren. Stattdessen wurde eine Nebenwirkung des Drehtellers als Absolutwertgeber ausgenutzt: Der Befehl, einen neuen Absolutwert an der aktuellen Position des Drehtellers zu setzen, wurde mit seiner höheren Priorität verwendet, um das Drehen nach links oder rechts zu stoppen.

Lösung: Es musste ein neuer Befehl auf der untersten Steuerungsebene des Drehtellers implementiert werden, der das Stoppen des Drehtellers ermöglicht. Hierfür wurde im Control-Modul `CM_PositioningDrive` das UDT (User Defined Type) um ein neues Kommando erweitert. Dies war erforderlich, damit im Equipment-Modul `EM_Drehteller` ein entsprechender neuer Befehl hinzugefügt werden konnte.

Als Technologieobjekt besitzt der Drehteller bereits vorgeschriebene Funktionen. Durch das Einfügen dieser neuen Funktionalität konnte der Halte-Befehl korrekt in die Steuerung der Knöpfe integriert werden. Mit der korrekten Implementierung des Halte-Befehls war es außerdem möglich, in der Anlagenlogik den zuvor

missbräuchlich genutzten Befehl zum Setzen eines neuen Absolutwerts zu ersetzen. Dadurch konnte auch die Kalibrierung des Drehtellers ohne Benutzereingaben beim Kaltstart ermöglicht werden.

5.4.6 Kommunikation:

zyklisch und unitlogik zwischen mps und Abfüllung (mit mps counter auf abfüllseite/ Einführung deckel auf flasche im udt des ordermanagments) zwischen abfüllung und quality gate

Problem: Die Kommunikation zwischen den einzelnen Stationen der Anlage war auf „On-Demand“-Basis ausgelegt. Das bedeutet, dass jede Station nur zu einem bestimmten Zeitpunkt mit einer anderen Station kommuniziert hat. Wenn jedoch eine Station nicht bereit war, die Kommunikation zu empfangen, ging diese „verloren“. Dies führte dazu, dass Anlagenteile nicht mehr synchron arbeiteten, die Anlage in einen fehlerhaften Zustand geriet und die Produktion gestoppt wurde.

Herausforderung: Hierfür musste ein neuer Baustein mit einem Watchdog-Timer eingeführt werden, der die Kommunikation zwischen den Stationen überwacht.

Lösung: Der neue Kommunikationsbaustein hat den alten Kommunikationsbaustein ersetzt, und die Logik wurde zu einer Acknowledge-Logik umgeschrieben. Bis ein Acknowledge von der anderen Station kam, wurde die Kommunikation nicht als abgeschlossen betrachtet. Dies führte dazu, dass die Produktion nach Wiederherstellung eines fehlerfreien Zustands der Anlage fortgesetzt werden konnte und keine Kommunikation „verloren“ ging.

5.5 Projekt SmartFactory-Anlage: Ergebnis

Der finale Funktionsumfang des Projekts ist zum Zeitpunkt der Erstellung dieses Berichts noch nicht vollständig absehbar. Dennoch wurden bereits wesentliche Meilensteine gemäß den Vorgaben erreicht. Mit Ausnahme einiger kleinerer Bugs konnten alle kritischen, den Prozess unterbrechenden Fehler an der Abfüllstation behoben werden. Dadurch wurde das Hauptziel, die Robustheit der Anlage zu

steigern, erfolgreich erreicht. Die weiteren Arbeiten konzentrieren sich nun auf die Verbesserung der Benutzerfreundlichkeit und der Erweiterbarkeit der Anlage.

Aus den täglichen Meetings und dem Feedback der Projektmitglieder geht hervor, dass auch an den anderen Stationen der Anlage deutliche Fortschritte erzielt wurden. Falls der Projektfortschritt bis zum Ende des Praxissemesters nicht vollständig den gesetzten Zielen entspricht, bieten die sorgfältig erstellten Konzeptsdokumente eine solide Grundlage für eine erfolgreiche Weiterführung durch zukünftige Studierende.

Durch die vorgenommenen Änderungen wurde nicht nur die Effizienz der Abfüllstation gesteigert, sondern auch die Benutzerfreundlichkeit und Robustheit des Systems deutlich verbessert.

6 Persönliches Fazit zum Praktikum (umschreiben)

Rückblickend möchte ich auf meine persönlichen Eindrücke des Praxissemesters bei der Siemens AG eingehen. Für mich war die Arbeit in einem großen Industrieunternehmen eine völlig neue und spannende Erfahrung. Besonders die Arbeitsweise in meiner Fachabteilung, dem Expert House, hat mich beeindruckt. Hervorheben möchte ich die täglichen Meetings, die es jederzeit ermöglichten, einen ganzheitlichen Überblick über den Projektfortschritt zu erhalten.

Ich habe gelernt, dass verschiedene Projektmanagement-Modelle, wie z. B. das Wasserfallmodell oder Kanban, nicht zwingend getrennt voneinander eingesetzt werden müssen, sondern dass eine sinnvolle Kombination der Methoden je nach Situation große Vorteile bringen kann. Auch Konzepte wie „Lessons learned“ und „Expert Talks“ empfand ich als äußerst sinnvoll und hilfreich.

Etwas ungünstig war es aus meiner Sicht, dass die SPE-Phase mitten in der Abteilungsphase stattfand und nicht zu Beginn oder am Ende. Dadurch ging meines Erachtens ein Teil des Blicks auf den Projektfortschritt der SmartFactory-Anlage verloren. Neben den organisatorischen Aspekten empfand ich die Programmierung mit dem TIA-Portal als besonders interessant. Diese Art der Programmierung unterscheidet sich deutlich von klassischen Programmiersprachen und erfordert, beispielsweise durch den zyklischen Programmablauf und das Bausteinkonzept, besondere Aufmerksamkeit.

Mein persönliches Highlight war jedoch die Arbeit an einer Industrieanlage. Es ist ein besonderes Erfolgserlebnis, wenn ein gut durchdachtes Konzept in einen funktionierenden Code umgesetzt wird und dadurch ein reibungsloser Prozessablauf an einer Anlage mit unzähligen Sensoren und Aktoren ermöglicht wird. Wer neben der reinen Informatik auch an Mechanik und Elektrotechnik interessiert ist, dem kann ich ein Praxissemester im Expert House bei der Siemens AG sehr empfehlen.

Literaturverzeichnis

- [1] Liam Bee. *PLC and HMI development with Siemens TIA Portal: develop PLC and HMI programs using standard methods and structured approaches with TIA Portal V17*. Packt Publishing, Limited, 2022. Preface.
- [2] Siemens. Pflichtenheft-smartfactory, 2022. internes Dokument.