

Hochschule Merseburg (FH)
University of Applied Sciences



Fachbereich Ingenieur- und Naturwissenschaften
Fachgebiet Automatisierungstechnik

Bachelorarbeit

zur Erlangung des akademischen Grades
Bachelor of Engineering (B.Eng.)

Digitale Zwillinge in der Industrie 4.0: Gegenüberstellung der Softwarepakete SIMIT und Factory IO

Vorgelegt von: Rico Dorta Pérez

Erstgutachter: Prof. Dr.-Ing. Peter Helm

Betrieblicher Betreuer: M. Eng. Christian Reinhardt

Abgabe: 14. November 2023

Inhaltsverzeichnis

Selbstständigkeitserklärung.....	4
Abkürzungsverzeichnis.....	5
Abbildungsverzeichnis.....	6
Tabellenverzeichnis.....	6
Glossar.....	7
1 Einleitung.....	9
1.1 Die Partnerfirma INGenium EMSR GmbH.....	9
1.2 Motivation.....	10
1.3 Aufgabenstellung.....	11
2 Theoretischer Rahmen.....	12
2.1 Industrie 4.0.....	12
2.2 Automatisierung.....	16
2.2.1 IoT-Internet of Things.....	18
2.2.2 Ethernet.....	19
2.2.3 TCP/ IP.....	19
2.2.4 OPC UA.....	20
2.2.5 Cloud Computing.....	20
2.2.6 OSI-Modell.....	21
2.3 Feldbussysteme in der Automatisierungstechnik.....	23
2.3.1 Topologien.....	23
2.3.2 Profibus.....	24
2.3.3 Profinet.....	25
2.3.4 AS-Interface.....	25
2.4 Digitale Zwillinge.....	26
2.4.1 Anwendung.....	27
2.4.2 Die Schlüsseltechnologien.....	28
2.4.3 Eine erste Einschätzung.....	28
3 Softwarelösungen.....	29
3.1 SIMIT.....	29
3.1.1 SIMIT Simulation Plattform (SIMIT SP).....	30
3.1.2 SIMIT Unit.....	32
3.2 Factory IO.....	33
4 Die Förderstrecke.....	34
4.1 Einführung.....	34
4.2 Radio Frequency Identification-Chip (RFID-Chip).....	35
4.3 Aufbau der Anlage.....	38
4.3.1 Sortiereinheit 1.....	38

4.3.2	Palettierer.....	42
4.3.3	Sortiereinheit 2.....	43
4.4	Inbetriebnahme der Anlage.....	43
4.4.1	Betriebsart Handbetrieb.....	44
4.4.2	Betriebsart Automatik.....	44
4.5	Codesys V3.5: das Programm.....	45
4.5.1	Beschreibung.....	45
4.5.2	Betriebskontrolle.....	46
4.5.3	Sortiereinheit 1.....	47
4.5.4	Palettierer.....	48
4.5.5	Sortiereinheit 2.....	49
4.5.6	Optimierungsmöglichkeiten.....	50
5	Die Sortieranlage.....	51
5.1	Einführung.....	51
5.2	Aufbau der Anlage.....	52
5.3	Inbetriebnahme der Anlage.....	54
5.3.1	Betriebsart Handbetrieb.....	55
5.3.2	Betriebsart Automatik.....	55
6	Gegenüberstellung.....	56
7	Fazit und Ausblick.....	58
8	Literaturverzeichnis.....	Fehler! Textmarke nicht definiert.
	Anlagen.....	63

Selbstständigkeitserklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit (entsprechend der genannten Verantwortlichen) selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Ich versichere weiterhin, dass die auf elektronischem Wege eingereichten Unterlagen mit den schriftlichen Ausfertigungen übereinstimmen.

Ort, Datum

Unterschrift

Abkürzungsverzeichnis

Abb.....	Abbildung
AS-i	Aktor-Sensor-Interface
CD.....	collision detection
CIM	computer-integrated manufacturing
CS.....	carrier sense
CSMA.....	Carrier Sense Multiple Access
DARPA.....	Defense Advanced Research Projects Agency
dt.....	deutsch
EMSR.....	Elektrotechnik & Mess-, Steuerungs- & Regelungstechnik
ERP-System.....	Enterprise-Resource-Planning-System
FDDI.....	Fiber Distributed Data Interface
GBit/s	Gigabit pro Sekunde
HF	high frequency
HMI	Human Machine Interface
IaaS.....	Infrastructure as a Service
IHK.....	Industrie- und Handelskammer
IIoT.....	Industrial Internet of Things
IoT.....	Internet of Things
IP	Internet Protocol
IP67.....	Ingress Protection 67
ISO.....	International Organization for Standardization
IT	Information Technology
ITU.....	International Telecommunication Union
JSON	JavaScript object notation
LAN.....	local area network
LF.....	low frequency
M2M.....	machine-to-machine
MA	multiple access
MBit/s	Megabit pro Sekunde
MES	Manufacturing Execution System
NIST.....	National Institute for Standards and Technology
OPC UA	Open Platform Communications Unified Architecture
OSI.....	Open Systems Interconnection-Modell
PaaS	Platform as a Service
PAL	Prüfungsaufgaben- und Lehrmittelentwicklungsstelle
PI	Profinet International
PLC	programmable logic controller
Profibus DP	dezentraler Peripheriebus
Profibus FMS	Profibus Fieldbus Message Specification
Profibus PA	Profibus Process Automation
RFID-Chips	Radio Frequency Identification Chips
SaaS	Software as a Service
SCADA-Systeme.....	Supervisory Control and Data Acquisition
SHF.....	super high frequency
SoA	serviceorientierte Architektur
SPS.....	speicherprogrammierbare Steuerung
TCP.....	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
UHF.....	ultra high frequency
usw.....	und so weiter
XML.....	extensible markup language

Abbildungsverzeichnis

Abb. 2.1-1: Modicon 084 [17]	14
Abb. 2.1-2: Die vier industriellen Revolutionen [19]	15
Abb. 2.1.1-1: Die Automatisierungspyramide der industriellen Fertigung [5]	17
Abb. 2.3.5-1: Die sieben Layer des OSI-Modells. [25]	22
Abb. 2.2.7-1: Netzwerktopologien [26]	23
Abb. 2.3.4-1: Asi-Kabel und Durchdringungstechnik [33]	26
Abb. 2.4.-1: idealisierte Darstellung des Verhältnisses von physischem und virtuellem Raum [35]	27
Abb. 3.1-1: mögliche Einbindung von SIMIT [37]	30
Abb. 3.1.1-1: Bedienoberfläche	31
Abb. 4.1-1: Die Förderstrecke in Factory IO 1	35
Abb. 4.2.-1: RFID-Tag passiv [43]	36
Abb. 4.2.-2: RFID-Tag semi-passiv [44]	36
Abb. 4.2.-3: RFID-Tag aktiv [45]	37
Abb. 4.2-4: Vor und Nachteile der unterschiedliche RFID-Typen [21]	37
Abb. 4.3.1-2: Sortiereinheit 1	38
Abb. 4.3.1-2: Lichtschranke mit Dual- und Dezimalcode und Summe des Dezimalcodes	39
Abb. 4.3.1-2: RFID-Sensor	40
Abb. 4.3.2-1: Palettierer	42
Abb. 4.3.3-1: Sortiereinheit 2	43
Abb. 4.5.2-1: Startbaustein	46
Abb. 4.5.2-2: Zähler Not-Halt	47
Abb. 4.5.4-1: Kartonzähler	48
Abb. 4.5.5-1: Bedingung Sortierer 2	49
Abb. 5.1-1: Sortieranlage IHK-PAL-Prüfung [48]	51
Abb. 5.2-1: SIMIT Sortieranlage [14]	54

Tabellenverzeichnis

Tabelle 1: Typen der SIMIT Unit [40]	33
Tabelle 2: RFID-Tag-Frequenzen [21]	37
Tabelle 3: Einheitenkonvertierung der Kartons in Factory IO	41
Tabelle 4: Lizenzkosten Factory IO	57

Glossar

Assets

bezeichnen Hardware-, Softwaresysteme oder Informationen, welche für Unternehmen wertvoll sind. [1]

Broad network access

bedeutet, dass die angebotenen Dienste über Standard-Mechanismen über das Netzwerk zugänglich sind und nicht an spezifische Clients gebunden sind. [2]

deterministische Kommunikation

ist ein kontrollierter Vorgang, der genauestens strukturiert ist und nach einem festgelegten Schema durchgezogen wird. [3]

Hardware-in-the-Loop (HiL)

ist ein Verfahren, bei dem reale eingebettete Systeme über Simulationssoftware, genannt HiL-Simulatoren, getestet werden, um die reale Systemumgebung zu simulieren. [4]

IO-Link-Geräte

sind Kommunikationsfähige Sensoren und Aktoren. [5]

Lean Management

ist eine Geschäftsstrategie, die die effiziente Gestaltung der Wertschöpfungskette anstrebt. [6]

Measured Service

ermöglicht die Überwachung und Zuteilung von Ressourcen gemäß der Nutzung für Cloud-Anwender. [2]

On-demand self-service

ermöglicht die automatische Bereitstellung von Ressourcen wie Rechenleistung und Speicher ohne direkte Interaktion mit dem Service Provider. [2]

Physik-Engine

ist eine Computersoftware, die dazu verwendet wird, physikalische Gesetze und Phänomene zu berechnen und zu simulieren. [7]

Rapid elasticity

erlaubt eine schnelle und elastische Bereitstellung von Services, mitunter sogar automatisch. Dies schafft aus der Sicht der Anwender den Eindruck unbegrenzter Ressourcen. [2]

Redundanz (technisch)

Bauteile werden als redundant bezeichnet, wenn sie in einem System mehrmals vorkommen, um, im Falle eines Ausfalls, Sicherheitsfunktionen aufrecht zu erhalten. [8]

Ressource pooling

ermöglicht den Anwendern die gemeinsame Nutzung von Ressourcen in einem Multi-Tenant Modell (zu Deutsch Mehrmieter-Modell). [2]

Skaleneffekten

bezeichnen die Kostenvorteile, die einem Unternehmen durch reduzierte Kosten pro produziertem Produkt entstehen. [9]

Stand-alone-Lösungen

sind Lösungen, die darauf abzielen, eigenständiges Funktionieren (eines Gerätes, Programms usw.) zu ermöglichen. [10]

IP67

Schutzart, die angibt, wie ein Gerät vor dem unerwünschten Eindringen von Fremdkörpern sowie Flüssigkeiten geschützt ist. Ebenfalls ist der Schutz vor mechanischer Belastung genau definiert. Die Ziffern nach der Abkürzung IP geben Auskunft gegen welche Einflüsse ein Produkt geschützt ist (vgl. [11]).

Data Science,

zu Deutsch Datenwissenschaft, befasst sich mit der Analyse von großen Datenmengen, der Erkennung von Anomalien in diesen Daten und der Vorhersage zukünftiger Ereignisse. [12]

1 Einleitung

1.1 Die Partnerfirma INGenium EMSR GmbH

Die INGenium EMSR GmbH ist ein junges Start-Up-Dienstleistungsunternehmen, welches am 01.08.2023 gegründet wurde und sich auf maßgeschneiderte Lösungen im Bereich der Elektrotechnik sowie der Mess-, Steuerungs- und Regelungstechnik spezialisiert hat. Innerhalb dieses Kontexts bietet die Firma ihre Unterstützung unabhängig von speziellen Herstellern oder Lieferanten und immer in enger Abstimmung mit den Kunden an, sodass deren Wünsche und Erfahrungen in die Umsetzung integriert werden. Das vorrangige Ziel der INGenium EMSR GmbH besteht darin, bedarfsgerechte, wirtschaftliche und kostenoptimierte Planungen zu erstellen. Erfahrene Ingenieure überwachen und begleiten anschließend die Umsetzung, um sicherzustellen, dass sämtliche erforderlichen Revisionsunterlagen und Dokumentationen termingerecht bei der Abnahme der Elektroanlage für Bauvorhaben bereitstehen. Außerdem bietet das Unternehmen Ingenieurdienstleistungen speziell für die Chemie- und Pharma-Branche an. In diesen Branchen wird großer Wert auf hohe Automatisierung und Anlagenverfügbarkeit gelegt. Dabei spielen das elektro-, mess-, steuer- und regelungstechnische Engineering (EMSR) sowie die Automatisierung von Prozessen eine entscheidende Rolle in der Anlagenplanung. Die Entwicklung von energieeffizienten Energiesystemkomponenten und modernen Automatisierungssystemen ist von zentraler Bedeutung. Diese Planungen dienen der Optimierung der Anlagensteuerung und tragen zur Steigerung der Effizienz, Flexibilität und Wirtschaftlichkeit der Prozessanlagen bei. Die Dienstleistungen der INGenium EMSR GmbH im Bereich Mess-, Steuer- und Regelungstechnik (MSR) umfassen unter anderem:

- die Planung von Energieversorgungssystemen
- die Entwicklung von Prozessleitsystemen
- die Auslegung von Antriebstechnik.

Die Erstellung von Modellen für Automatisierungstrecken erfolgt zunächst mithilfe spezieller Software. Je nach den Anforderungen des Kunden werden entweder wissensorientierte oder prozessorientierte Ansätze angewendet. Die wissensorientierte Methode nutzt analytische Prozesse, um einen Entwurf für die Implementierung verschiedener Automatisierungsfunktionen zu erstellen. Hierbei entstehen Automatisierungssysteme, die modellbasierte Regelung, Steuerung und Fehlerdiagnose umfassen. Im Gegensatz dazu konzentriert sich die prozessorientierte Methode auf die Entwicklung von Prozessen und mechatronischen Systemen. Dies umfasst die computergestützte Modellbildung, Simulation und digitale Steuerung von Werkzeugmaschinen sowie Robotern. Die Automatisierungslösung wird dabei an die vorhandene

Infrastruktur und die etablierten Prozesse angepasst. Basierend auf diesen Modellen erfolgt anschließend die Planung der tatsächlichen Automatisierungsstrecke. [13] [14]

1.2 Motivation

Die vierte industrielle Revolution, oft als "Industrie 4.0" bezeichnet, hat die Art und Weise, wie Unternehmen produzieren, überwachen und warten, revolutioniert. Eine entscheidende Komponente dieser Revolution sind digitale Zwillinge, virtuelle Replikationen physischer Anlagen oder Prozesse, die es ermöglichen, reale Situationen in einer digitalen Umgebung nachzubilden. In dieser spannenden Ära der Industrie 4.0 tragen sie dazu bei, die Effizienz zu steigern, Kosten zu senken und Innovationen voranzutreiben.

Meine persönlichen Erfahrungen haben mich bewegt, diese Bachelorarbeit zu verfassen. Während meines Studiums an der Hochschule Merseburg konnte ich erste Schritte mit Factory IO unternehmen, einer Software, die es ermöglicht, virtuelle Fabriken zu erstellen und zu programmieren. Diese Erfahrungen weckten mein Interesse an der Anwendung digitaler Zwillinge in der Praxis.

Bei meinem letzten Arbeitgeber hatte ich die Gelegenheit, an einem Projekt teilzunehmen, bei dem die Idee entstand, für einen Kunden ein digitales Ebenbild seiner Anlage mithilfe der SIMIT Simulationsplattform zu entwickeln. Um die Funktionalitäten von SIMIT vollständig zu verstehen und effektiv anzuwenden, nahm ich an einem dreistufigen Online-Seminar teil. Auch wenn der Projektansatz an der finanziellen Umsetzung scheiterte, so gaben mir die Erfahrungen einen tieferen Einblick in die Welt der virtuellen Abbilder und zeigten mir das immense Potenzial dieser Technologie für die Industrie 4.0.

Die Motivation für diese Bachelorarbeit liegt darin, die beiden Softwarepakete, SIMIT und Factory IO, gegenüberzustellen und ihre jeweiligen Vor- und Nachteile in der Anwendung zu untersuchen. Durch diese Gegenüberstellung können Erkenntnisse gewonnen werden, die Unternehmen wiederum bei der Auswahl, der am besten geeigneten Software für ihre spezifischen Anforderungen unterstützen können.

1.3 Aufgabenstellung

Das Ziel dieser Bachelorarbeit besteht darin, einen detaillierten Vergleich zwischen den Simulationssoftwarepaketen SIMIT und Factory IO durchzuführen. Hierbei wird jeweils ein Anlagenmodell erstellt, das die Grundlage für die Schaffung eines digitalen Zwillings bildet. Dieser Ansatz dient dazu, einen Einblick in die vielseitige Welt der digitalen Zwillinge zu gewinnen und ihre praktische Anwendung zu untersuchen.

Aufgrund von Budgetbeschränkungen und nicht vorhandener Lizenzen wird das Anlagenmodell "Sortieranlage" in der SIMIT Simulationsplattform durch ein Modell aus einem Projektheft des Verlags der Westermann Gruppe ersetzt [15], das ausschließlich als Referenzobjekt für das Modell in Factory IO dient. Daher wird eine detaillierte Untersuchung des SPS-Programms in dieser Arbeit nicht durchgeführt. Jedoch lässt sich auch dieses Modell vollumfänglich testen, sodass ein Vergleich beider Anlagen möglich ist.

Die hauptsächliche praktische Tätigkeit dieser Arbeit beinhaltet die Erstellung und Programmierung einer Förderstrecke, die einen Kartonpalettierer und einen Sortierer in der Simulationssoftware Factory IO umfasst. In diesem Prozess werden Kartons in drei verschiedenen Größen zufällig ausgegeben und anschließend auf verschiedene Förderbänder entsprechend ihres Gewichts verteilt. Die Programmierung erfolgt mithilfe von CODESYS, während die Kommunikation und Datenübertragung zwischen beiden Programmen durch die Verwendung von OPC UA realisiert wird.

Diese Bachelorarbeit zielt darauf ab, die Stärken und Schwächen der beiden Simulationssoftwarepakete im Hinblick auf die Erstellung digitaler Zwillinge für industrielle Anwendungen herauszuarbeiten und wertvolle Erkenntnisse zur Auswahl der geeigneten Softwarelösung zu bieten.

2 Theoretischer Rahmen

2.1 Industrie 4.0

Im Zusammenhang mit dem Begriff Industrie 4.0 wird immer wieder von der vierten industriellen Revolution gesprochen. Dem zu Folge ist es nur logisch auch die vergangenen drei Revolutionen näher zu beleuchten, um einen Überblick über eine der wichtigsten und zugleich bedenklichsten Epochen der Menschheitsgeschichte zu bekommen. Die industrielle Revolution, die in der zweiten Hälfte des 18. Jahrhunderts in Großbritannien begann und sich über die nächsten zwei Jahrhunderte weltweit ausbreitete, war ein bedeutender Wendepunkt in der Geschichte der Menschheit. Sie veränderte die Art und Weise, wie Menschen arbeiten, produzieren und leben. Die erste Phase begann, getrieben von der Entwicklung der Dampfmaschine, um 1750. Diese Revolution ermöglichte die Industrialisierung und trug wesentlich dazu bei, dass es seitdem in industriell entwickelten Ländern keine strukturell bedingten Hungerkatastrophen mehr gab, was wiederum dazu führte, dass die Bevölkerungszahlen signifikant anstiegen. Außerdem konnte die Bevölkerung dank Verbesserungen im Transportsystem, wie der Dampfschiffahrt und der Eisenbahn, zuverlässiger mit Kleidung und Nahrung versorgt werden. Andererseits führte die gesteigerte Produktivität bei der Herstellung von Grundnahrungsmitteln, insbesondere in der Landwirtschaft, zu erheblichen Verbesserungen.

Natürlich haben derartige Revolutionen stets auch enorme Auswirkungen auf die Gesellschaft. Traditionelle Handwerksbetriebe und die Landwirtschaft erlebten drastische Beschäftigungsabbauten, während sich zwei neue soziale Schichten formten: die Arbeiter in den Fabriken und die Eigentümer dieser Fabriken. Letztere waren in den meisten Fällen die Profiteure der industriellen Wertschöpfung, wohingegen besonders in den Anfängen der Industrialisierung die Fabrikarbeiter ausgebeutet wurden. Kinderarbeit war weit verbreitet, selbst Kleinkinder mussten in den Fabriken arbeiten und die Lebenserwartung der Arbeiterschaft war gering. Trotz der widrigen Arbeitsbedingungen zog es immer mehr Menschen in die Städte, was eine strukturelle Armut unter den Fabrikarbeitern, auch als "Pauperismus" bekannt, zur Folge hatte.

Beim Übergang zur zweiten Industriellen Revolution (1870), geprägt durch arbeitsteilige Massenproduktion und die Einführung elektrischer Energie, kam es zu einer sozialen Umgestaltung. Die Gesellschaft erkannte die Notwendigkeit, die Fabrikarbeiter nicht mehr auszubeuten, sondern ihren Wohlstand zu fördern, um soziale Spannungen abzubauen. Dies wurde durch die großindustrielle Massenproduktion ermöglicht, die aufgrund von *Skaleneffekten* kostengünstige Produkte ermöglichte. In dieser Zeit gewannen Gewerkschaften stark an Bedeutung und der Übergang von der ersten zur zweiten Industriellen Revolution ebnete den Weg für die Sozialdemokratie. Gleichzeitig wurden elektrische Antriebe und Verbrennungsmotoren

entwickelt. Insbesondere elektrifizierte Antriebssysteme ermöglichten die Dezentralisierung, bei der Maschinen nicht mehr zentral, sondern dezentral betrieben wurden. Zugleich gewann Erdöl als wichtiger Rohstoff für die chemische Industrie und als Treibstoff für mobile Systeme an Bedeutung, insbesondere für Automobile. Die großindustrielle Massenproduktion breitete sich in Branchen wie der Chemie- und Elektroindustrie sowie dem Maschinenbau und der Automobilindustrie aus und die Bevölkerung wuchs weiter an. In dieser Zeit wurden die Grundlagen für unsere moderne konsumorientierte Wohlstandsgesellschaft gelegt. [16]

In der zweiten Hälfte des 20. Jahrhunderts erlebte die Fertigungstechnologie eine wegweisende Entwicklung, welche die dritte industrielle Revolution einleitete. Diese Veränderung wurde durch den Fortschritt in der Informationstechnologie initiiert und wird oft als die "digitale Revolution" bezeichnet. Computer, Software und Mikroprozessoren begannen, einen immer größeren Einfluss auf Industrieanlagen auszuüben und übernahmen zunehmend die Kontrolle über verschiedene Produktionsprozesse. Ein Meilenstein in diesem Wandel war die Einführung der Modicon 084 (siehe Abb. 2.1-1: Modicon 084) im Jahr 1969, die als erste speicherprogrammierbare Steuerung in eine Produktionsanlage integriert wurde. Mit dieser Innovation konnten Maschinen nicht mehr auf starre Abläufe beschränkt sein, sondern sie konnten schnell und flexibel gemäß den Anforderungen eines bestimmten Auftrags umprogrammiert werden. Dies war möglich, dank der fortschreitenden Miniaturisierung elektronischer Komponenten und der Steigerung ihrer Leistungsfähigkeit. Die bis dahin zentralisierte Steuerung der Produktionsabläufe wurde durch dezentrale, selbstregulierende Einheiten ersetzt. Diese mussten in Echtzeit überwacht und koordiniert werden, um eine reibungslose Produktion sicherzustellen. Gleichzeitig begannen Unternehmen, Fertigungsschritte, die nicht zu ihrem Kerngeschäft gehörten, vermehrt an externe Zulieferer auszulagern. Dies geschah im Rahmen von Outsourcing-Strategien, bei denen enge Zusammenarbeit in Lieferketten und Wertschöpfungsnetzwerken mit den Zulieferern eine wichtige Rolle spielte. [17]

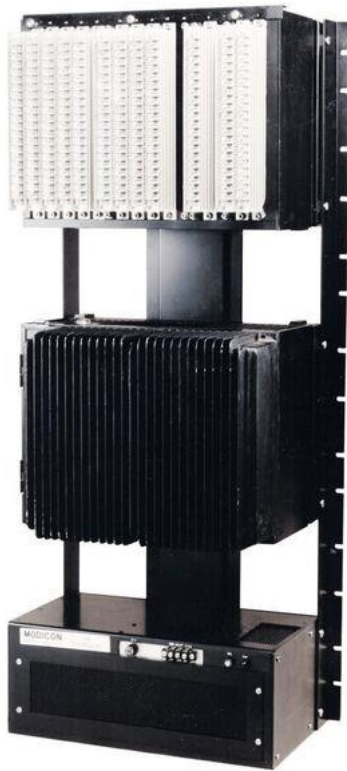


Abb. 2.1-1: Modicon 084 [18]

Die aktuelle und erstmals 2011 auf der Hannover-Messe aufgegriffen, ist die vierte industrielle Revolution, kurz Industrie 4.0. Dieser neue Abschnitt ist geprägt von vernetzten und kommunizierenden Systemen, die auf modernster Internettechnologie basieren. Dieser Fortschritt folgt auf frühere Revolutionen, die mit mechanischen Produktionsanlagen unter Nutzung von Wasser- und Dampfkraft (erste Revolution), der Einführung arbeitsteiliger Massenproduktion durch elektrische Energie (zweite Revolution) und dem Einsatz von Elektronik und Informationstechnologie zur Automatisierung (dritte Revolution) einhergingen. Durch die Kombination dieser Faktoren mit Produktions- und Automatisierungstechnik wird angestrebt, eine neue Ebene der Organisation und Steuerung der gesamten Wertschöpfungskette über den gesamten Lebenszyklus von Produkten zu erreichen. Das Ziel ist eine erhebliche Steigerung der Flexibilität und Verbesserung der Wertschöpfung sowie die Individualisierung von Produkten.

In dieser vierten industriellen Revolution, die im Wesentlichen als eine durch die Informationstechnologie ausgelöste Revolution betrachtet wird, sollen produzierende Unternehmen bis zum Jahr 2025 durch folgende vier Merkmale gekennzeichnet sein:

- Ein hohes Maß an Produktindividualisierung, das über die herkömmlichen Standardkonfigurationsoptionen hinausgeht.
- Hochgradig flexible und gleichzeitig effiziente Produktion, selbst bei sehr kleinen Losgrößen.

- Umfassende Einbindung von Kunden und Geschäftspartnern in Geschäfts- und Wertschöpfungsprozesse.
- Die Verknüpfung von Produktion und hochwertigen Dienstleistungen, beispielsweise die bereitgestellten nutzerorientierten Informationsdienste, die zu sogenannten hybriden Produkten führen, wie beispielsweise den Verkauf von "Mobilität" anstelle von reinen Fahrzeugen. [19]

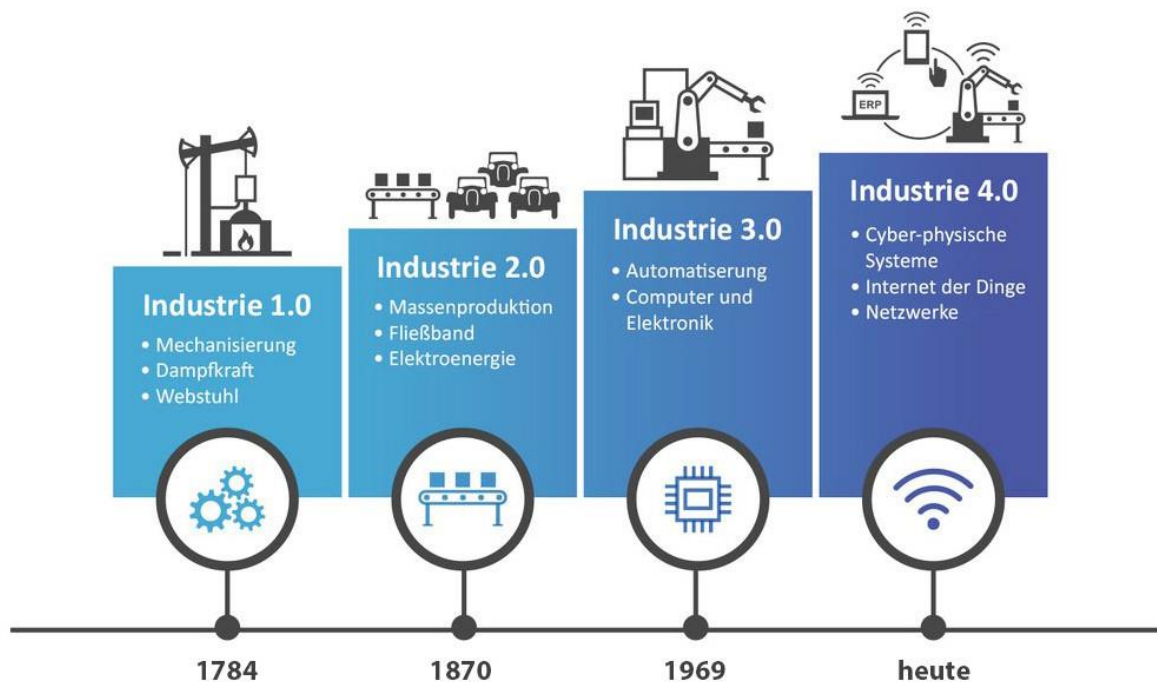


Abb. 2.1-2: Die vier industriellen Revolutionen [20]

Basierend auf diesen Punkten hat der Autor des Buches „Einführung und Umsetzung von Industrie 4.0“, Armin Roth, den Begriff Industrie 4.0 wie folgt definiert:

„Industrie 4.0 umfasst die Vernetzung aller menschlichen und maschinellen Akteure über die komplette Wertschöpfungskette sowie die Digitalisierung und Echtzeitauswertung aller hierfür relevanten Informationen, mit dem Ziel die Prozesse der Wertschöpfung transparenter und effizienter zu gestalten, um mit intelligenten Produkten und Dienstleistungen den Kundennutzen zu optimieren.“ ([19] S. 6) Die folgende Grafik (Abb. 2.1-2: Die vier industriellen Revolutionen) soll die Entwicklung der Industrie abschließend noch einmal veranschaulichen.

Die Ziele von Industrie 4.0 finden sich weltweit unter verschiedenen Begriffen wieder, darunter „Smart Factory,“ (dt. intelligente Fabrik) „Internet of Things (IoT)“ (dt. Internet der Dinge) und „(Business) Digitalization“ (dt. (Geschäfts) Digitalisierung). In der vorliegenden Quelle wird der Begriff "Internet of Things" jedoch in einem technologischen Kontext verwendet, was bedeutet, dass er die Vernetzung physischer Komponenten mithilfe von Internet-Technologien

beschreibt. Das IoT fungiert somit als ein technologischer Baustein innerhalb des Gesamtkonzepts von Industrie 4.0 (vgl. [19] S.6). Insgesamt betrachtet stellt Industrie 4.0 keine revolutionäre Neuerung dar. Viele der Konzepte, die unter dem Begriff Industrie 4.0 zusammengefasst werden, sind vielmehr eine Kombination oder konsequente Weiterentwicklung von bereits existierenden Konzepten aus der Automatisierung, dem computer-integrated manufacturing (CIM) (dt. rechnergestützte Produktion) der 1970er Jahre oder beispielsweise dem *Lean Management*. Allerdings wird erst durch den aktuellen Stand der konvergierenden Technologien die schrittweise Umsetzung dieser grundlegenden Konzepte in ihrer Gesamtheit möglich. Dadurch werden digitalisierte Produkte und Lösungen für die Kunden erfahrbar und mit dem Fortschritt der Technologien zunehmend intensiver erlebbar. Vor diesem Hintergrund kann Industrie 4.0 eher als eine Evolution, denn als Revolution betrachtet werden. Dennoch wird sie für Unternehmen, die diese Entwicklungen und die damit einhergehenden Veränderungen auf dem Markt unterschätzen, zu einer Revolution. Daher ist Industrie 4.0 sicherlich mehr als nur ein kurzfristiger Hype. Vielmehr ist sie mittelfristig, insbesondere aufgrund der steigenden Anforderungen der Kunden an zukünftige Lösungen, nicht mehr aufzuhalten. [19]

2.2 Automatisierung

Automation oder Automatisierung leitet sich vom griechischen Wort "auto" ab und beschreibt die eigenständige Ausführung eines Prozesses, bei dem Material, Informationen und Energie verändert oder umgewandelt werden. Bis heute gibt es keinen „fest definierten Fachinhalt“ (vgl. S. 18 [21]) für diesen Begriff, was womöglich an der Tatsache liegt, dass er mehr eine Art beschreibt, wie Prozesse ablaufen, als dass er definiert welche festen Bestandteile diese Prozesse beinhalten können. Vielmehr bedient sich die Automatisierung bestimmter Kernfelder (im Folgenden Techniken genannt), die dazu nötig sind, einen Prozess teil- oder vollautomatisch ablaufen zu lassen. Die wichtigsten sogenannten Basistechniken lauten:

- Sensor- Aktortechnik
- Regelungstechnik
- Steuerungstechnik
- Leittechnik
- Robotertechnik [21]

Die Automatisierung, besonders in der industriellen Fertigung, wird fortwährend komplexer, was zur Folge hat, dass Datenerhebung und -verarbeitung ebenso aufwendiger werden. Um diese Komplexität einsehbarer zu gestalten, wurde bereits in den 1980er Jahren die sogenannte Automatisierungspyramide eingeführt, welche die verschiedenen Ebenen der automatisierten Fertigung abbildet. Anfangs bestand die Pyramide aus drei Ebenen, wurde aber im Laufe der 1990er Jahre durch die drei Ebenen, der in den siebziger und achtziger Jahren

entstandenen CIM-Pyramide, ergänzt (siehe Abb. 2.1.1-1: Die Automatisierungspyramide der industriellen Fertigung). [19]

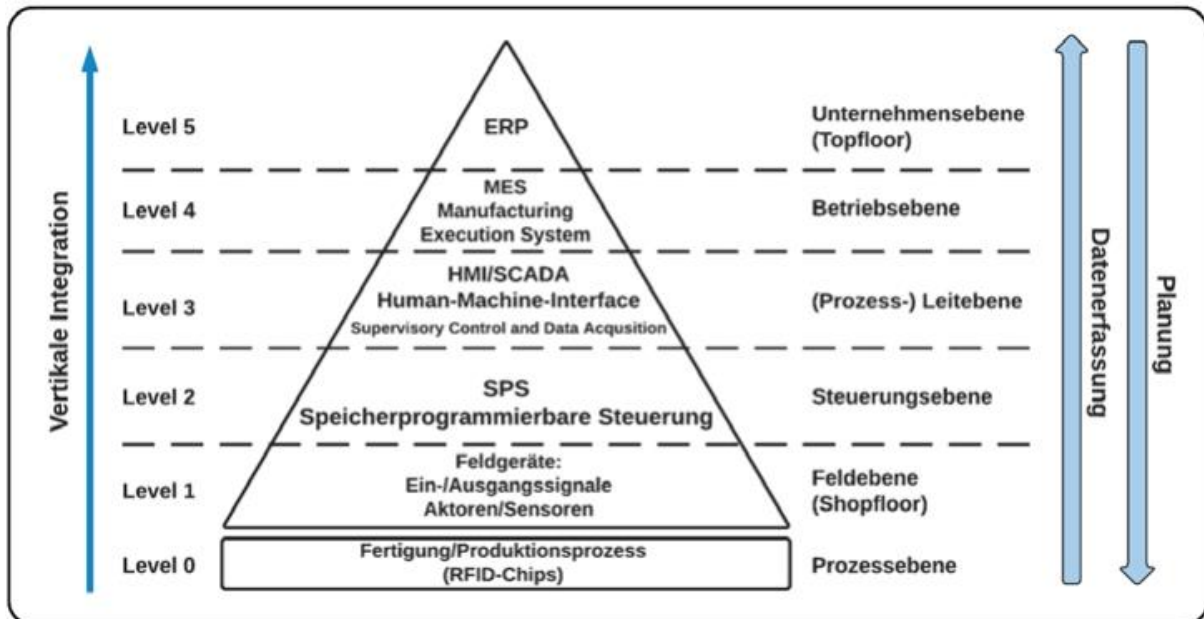


Abb. 2.1.1-1: Die Automatisierungspyramide der industriellen Fertigung [5]

Die unterste Ebene der Automatisierungspyramide umfasst die Fertigung und den eigentlichen Produktionsprozess. Hier liefern intelligente Produkte und Radio Frequency Identification-Chips (RFID-Chips) (dt. Radiofrequenz-Identifikationschips) Informationen zu Produkteigenschaften und Produktionsabläufen. Die Feldebene, auch als Shopfloor bezeichnet, umfasst den Produktionsbereich und die Produktionsstätte, wo die Wertschöpfung stattfindet. Hier sind Sensoren wie Temperaturfühler und Lichtschranken sowie Aktoren in Form von Schaltschützen oder elektrischen Reglern zu finden. Die Feldebene erfasst und verarbeitet produktionsrelevante Daten in Form von Ein- und Ausgangssignalen.

Die Steuerungsebene verwendet eine speicherprogrammierbare Steuerung (SPS), um Eingangssignale (Sensordaten) auszuwerten und entsprechende Ergebnisdaten (Ausgangssignale) an die Feldebene zurückzusenden. Aktoren, beispielsweise in Form von Druckluft- und Hydrauliksystemen, verarbeiten diese elektrischen Signale und setzen sie in mechanische Bewegungen um. Die Steuerungsebene spielt eine entscheidende Rolle bei der Umsetzung einer dezentral realisierten Maschinen- und Anlagensteuerung.

Die Leitebene dient der Visualisierung produktionsrelevanter Vorgänge. Hier werden Steuerungsinformationen und Warnmeldungen über Prozessleit-, HMI- (dt. Mensch-Maschine-Schnittstelle) und SCADA-Systeme (dt. Überwachungssteuerung und Datenerfassung) benutzerfreundlich dargestellt. Die Leitebene fungiert als eine Art Mensch-Maschine-Schnittstelle und ermöglicht die Bedienung und Überwachung des Systems.

Das Manufacturing Execution System (MES) (dt. Produktionsleitsystem) ist für die Steuerung, Lenkung und Kontrolle der Produktion verantwortlich und fungiert als Bindeglied zwischen der Maschinensteuerung und der Unternehmensebene (Topfloor). Die Hauptaufgaben des MES liegen in der Produktionsfeinplanung und der Datenerfassung, wobei relevante Daten an das Enterprise Resource Planning-System (ERP-System) (dt. Unternehmensressourcenplanung) für zukünftige Planungen weitergeleitet werden.

Das ERP-System auf der Unternehmensebene, auch als Topfloor bezeichnet, übernimmt die Produktionsgrobplanung und die Abwicklung von Bestellungen in der industriellen Fertigung. [5]

2.2.1 IoT-Internet of Things

Das Internet of Things (IoT), auch als "Allesnetz" bekannt, wurde erstmals 1999 von Kevin Ashton eingeführt und ermöglicht die Vernetzung physischer und virtueller Gegenstände über das Internet. Es umfasst eine breite Palette von Geräten, von Haushaltsgegenständen bis hin zu Industriemaschinen. Experten prognostizieren, dass bis 2025 mindestens 22 Milliarden vernetzte IoT-Geräte existieren werden. Die Ursprünge des IoT reichen jedoch zurück bis ins Jahr 1968, als Richard Morley Programmable Logic Controller (PLC) (dt. programmierbare Steuerung) für die Steuerung von Industriemaschinen entwickelte. Die Idee von Maschine-zu-Maschine-Vernetzung (M2M) wurde erstmals 1983 praktisch umgesetzt. Die umfassende Vernetzung aller Geräte erhielt 1989 einen Schub mit der Einführung des World Wide Webs und des Internet-Protokolls TCP/IP.

Den Begriff "Internet of Things" prägte Kevin Ashton im Jahr 1999. Die ursprüngliche Idee war die nahtlose Integration von mit Sensoren ausgestatteten Objekten in die Umwelt. Die Kommunikation über das Internet fehlte zu dieser Zeit, aber Sensoren in der Feldebene der Automatisierung erfüllten bereits wichtige Anforderungen des IoT. Ziel war es, Informationen von Prozessen zu erfassen, zu vernetzen und global verfügbar zu machen, wodurch prädiktive Wartung und verbesserte Prozesse ermöglicht wurden.

Die Einführung der Cloud-Technologie im Jahr 2003 und von OPC UA im Jahr 2007 machte das IoT zu einem globalen Vernetzungswerkzeug. Bis 2020 gab es zwischen 18 und 22 Milliarden vernetzte Geräte und bis 2025 werden voraussichtlich weitere 10 Milliarden hinzukommen. Wichtige Aspekte des IoT sind die Standardisierung von Komponenten und Diensten, sichere Netzwerkanbindung, Kosteneffizienz und die Entwicklung von digitalen Services. IoT ist eng mit Industrie 4.0 verbunden und beide Konzepte streben nach weit verbreiteter Informationsverteilung zur Lösung von Problemen. Insgesamt war das IoT von Anfang an darauf

ausgerichtet, Informationen weit zu verbreiten und sie überall dort nützlich zu machen, wo sie benötigt werden, ein Ansatz, der auch in Industrie 4.0 eine zentrale Rolle spielt. [22]

Zusammenfassend lässt sich sagen, dass IoT sich auf die umfassende Vernetzung von Maschinen und intelligenten Sensoren über das Internet bezieht. Dieses Konzept setzt auf Schlüsselkomponenten wie Ethernet, TCP/IP, das Internet selbst, OPC UA und Cloud Computing, die im Folgenden näher erläutert werden.

2.2.2 Ethernet

Im Laufe der Jahre hat sich Ethernet als dominierende Netzwerktechnologie im LAN-Bereich etabliert. Durch kontinuierliche Weiterentwicklung des ursprünglichen Ethernet-Standards mit 10 MBit/s konnte sich Ethernet gegenüber konkurrierenden Technologien wie Token Ring oder FDDI (dt. Lichtwellenleiter-Metro-Ring) am Anfang des neuen Jahrtausends durchsetzen. Aktuell hat Ethernet einen Marktanteil von mehr als 95% in allen installierten lokalen drahtgebundenen Netzwerken und wird voraussichtlich noch lange Zeit seine führende Position behalten.

Inzwischen hat sich Gigabit-Ethernet mit einer beeindruckenden Datenrate von 1 GBit/s etabliert. Die neuesten Entwicklungen ermöglichen sogar Übertragungsraten von 40 oder 100 GBit/s, was zu erheblichen Leistungssteigerungen in der klassischen Netzwerktechnik geführt hat. Wenn es heute um lokale Netzwerke geht, handelt es sich in den meisten Fällen um Ethernet in einer seiner vielfältigen Ausprägungen. [23]

2.2.3 TCP/ IP

TCP/IP ist ein Internetprotokoll, das die Datenübertragung zwischen Computern und Geräten im Internet ermöglicht. Es steht für "Transmission Control Protocol/Internet Protocol" und wurde in den 1970er Jahren von der DARPA (Defense Advanced Research Projects Agency) in den USA entwickelt. Ursprünglich war es nur eines von vielen Protokollen, wurde aber später zum Standard für das ARPAnet (Advanced Research Projects Agency Network), dem Vorläufer des heutigen Internets. TCP/IP regelt, wie Daten zwischen Geräten übertragen werden, wobei die Daten in Pakete aufgeteilt und am Empfangsort wieder zusammengeführt werden. Dies ermöglicht eine zuverlässigere Übertragung und verhindert Datenverlust. Die Pakete können verschiedene Routen nehmen, um überlastete Verbindungen zu vermeiden. Dieses Protokoll basiert auf einem 4-Schichten-Modell, durch das die Daten in einer bestimmten Reihenfolge verarbeitet werden, bevor sie auf der Empfängerseite in umgekehrter Reihenfolge wieder zusammengesetzt werden. Dieses Verfahren trägt dazu bei, die Effizienz und Zuverlässigkeit der Internetkommunikation sicherzustellen und markiert TCP/IP somit als weltweiten Standard. [24]

2.2.4 OPC UA

Eine zentrale Herausforderung von Industrie 4.0 und dem Industrial Internet of Things (IIoT) (dt. industrielles Internet der Dinge) besteht darin, einen sicheren und standardisierten Datenaustausch zwischen Geräten, Maschinen und Diensten zu ermöglichen. Mit Open Platform Communications Unified Architecture (OPC UA) wurde ein wichtiger Standard für diesen Zweck entwickelt. OPC UA ist mehr als nur ein Kommunikationsprotokoll, es handelt sich um ein Framework (dt. Gerüst) für industrielle Interoperabilität.

Mit OPC UA können Hersteller von Geräten und Maschinen objektorientierte Informationen über ihre Systeme beschreiben und gleichzeitig Zugriffsrechte unter Berücksichtigung von IT-Sicherheitsaspekten definieren. Dies ermöglicht es, die Kontrolle über Daten zu behalten und diese gezielt zu verteilen und zu monetarisieren.

OPC UA bietet zwei Mechanismen für den Datenaustausch: ein Client-Server-Modell für sicheren und bestätigten Informationsaustausch und ein Publisher-Subscriber-Modell für unbestätigten Informationsaustausch. Beide Mechanismen sind unabhängig vom zugrunde liegenden Kommunikationsprotokoll implementiert. Die eigentliche Bedeutung liegt in der Informationsmodellierung, bei der Hersteller bereits heute implizit Datenmodelle für ihre Geräte liefern. Die neue Welt von Geräten mit serviceorientierter Architektur (SoA) erleichtert das Verständnis dieser Modelle, da sie Dienste und deren Bedeutung anbieten. Dieses Konzept zieht bis in die "Things" selbst ein und spielt eine entscheidende Rolle in der Umsetzung von Industrie 4.0. [25]

2.2.5 Cloud Computing

Cloud Computing (dt. Datenwolke) ist eine IT-Infrastruktur, die über das Internet verfügbar gemacht wird, ohne dass sie lokal auf einem Computer installiert werden muss. Es umfasst Applikationen, Plattformen und Infrastruktur und spielt eine wichtige Rolle in der prädiktiven Wartung von Anlagen und Maschinen. Die NIST (National Institute for Standards and Technology) definierte die Cloud im Jahr 2011, obwohl erste Ansätze bereits in den 1990er Jahren existierten. Internetfirmen wie Amazon, Google und Yahoo trieben die Entwicklung ab 2004 voran, um Rechenleistung und Speicher virtuell zur Verfügung zu stellen. Durch Cloud Computing werden große Kosteneinsparungen im Bereich der Software und Hardware sowie in den skalierbaren Anwendungen erreicht. [22]

Um eine gewisse Kundenzufriedenheit zu garantieren, gibt es drei Servicemodelle:

- Software as a Service (SaaS) mit einer Vielzahl an Softwaresammlungen und Anwendungsprogrammen.
- Platform as a Service (PaaS) mit Programmierungs- und Laufzeitumgebungen und
- Infrastructure as a Service (IaaS) welches die Nutzung von Computerhardwareressourcen, Rechner und Speicher ermöglicht.

Clouds können öffentlich, privat oder hybrid sein. Die NIST definiert wichtige Arten für Cloud-Computings, darunter *On-demand self-service* (dt. Selbstbedienung auf Abruf), *Broad network access* (dt. breiter Netzwerkzugriff), *Ressource pooling* (dt. Ressourcenbündelung), *Rapid elasticity* (dt. schnelle Elastizität) und *Measured Service*. (dt. gemessener Service) (vgl. [22] S.42) In der Flugzeugindustrie und Automobilbranche wird Cloud Computing in Verbindung mit OPC UA genutzt, um Komponenten aus verschiedenen Standorten zu synchronisieren und zusammenzuführen. Der größte Nachteil und die höchsten Risiken von Cloud Computing liegen in der Sicherung gegen Zugriffe und Hacking bei der Kommunikation zwischen lokalen Kunden und entfernten Servern, was hohe Investitionen in die Softwareentwicklung erfordert. [22]

2.2.6 OSI-Modell

Das OSI-Modell (Open Systems Interconnection-Modell) bildet die Grundlage für die IoT-Kommunikation über Feldbusse und Bussysteme. Es regelt die Schnittstellen für horizontale (Maschine-zu-Maschine) und vertikale Kommunikation (Maschine-zu-SCADA-System) innerhalb der Automatisierungspyramide. Das OSI-Modell ist von großer Bedeutung für IoT und Industrie 4.0, da es die Kommunikation in vertikalen und horizontalen Schnittstellen detailliert definiert.

Die Entwicklung des OSI-Modells begann bereits in den späten 1970er Jahren und wurde später von der ITU und der ISO als Standard veröffentlicht. Das Modell besteht aus sieben Schichten, die Hardware- und Softwarekomponenten in aufsteigender Komplexität organisieren. Im Folgenden werden diese beschrieben und anschließend werden ihnen die zuvor erläuterten Begriffe zugeordnet.

Die unterste Schicht ist die Bitübertragungsschicht (Physical Layer), die sich um Hardwarekomponenten und die physische Übertragung von Bits kümmert. Die Sicherungsschicht für Daten und Verbindung (Data Link Layer) folgt, um fehlerfreie Datenübertragung sicherzustellen und den Zugriff auf die Bitübertragungsschicht zu kontrollieren. In diesen beiden Schichten ist das Ethernet angesiedelt.

Die Vermittlungsschicht (Network Layer) übernimmt das Routing von Datenpaketen und stellt Netzwerkadressen bereit, wobei Router eine wichtige Rolle spielen. Diese Vorgänge werden durch das „Internet Protocol“ (IP) bewerkstelligt. Anschließend ermöglicht das „Transmission Control Protocol“ (TCP) auf der Transportschicht (Transport Layer) die Segmentierung von Datenströmen und die Vermeidung von Überlastungen. Auf ihr könnte beispielsweise das Protokoll OPC UA eingesetzt werden, um Daten zwischen industriellen Geräten auszutauschen.

Es folgt die Sitzungsschicht (Session Layer), welche die Prozesskommunikation zwischen Systemen unterstützt und die Wiederaufnahme unterbrochener Verbindungen ermöglicht. Auf sie folgt die Darstellungsschicht (Presentation Layer). Diese wandelt Datenformate um und unterstützt Datenkompression sowie Verschlüsselung. Die oberste Ebene wird von der Anwendungsschicht (Application Layer) gebildet und bietet Funktionen für Anwendungen, regelt die Verbindungen zu den darunter liegenden Schichten und ermöglicht die Eingabe und Ausgabe von Daten. Hier könnte beispielsweise auf das Cloud Computing in Form von cloudbasierten Anwendungen und Diensten zurückgegriffen werden. Diese sieben Schichten bilden das OSI-Modell und dienen als grundlegender Rahmen für die Entwicklung von Netzwerkprotokollen und -technologien. [22] Folgende Abbildung soll zuvor beschriebenes zum besseren Verständnis visualisieren.

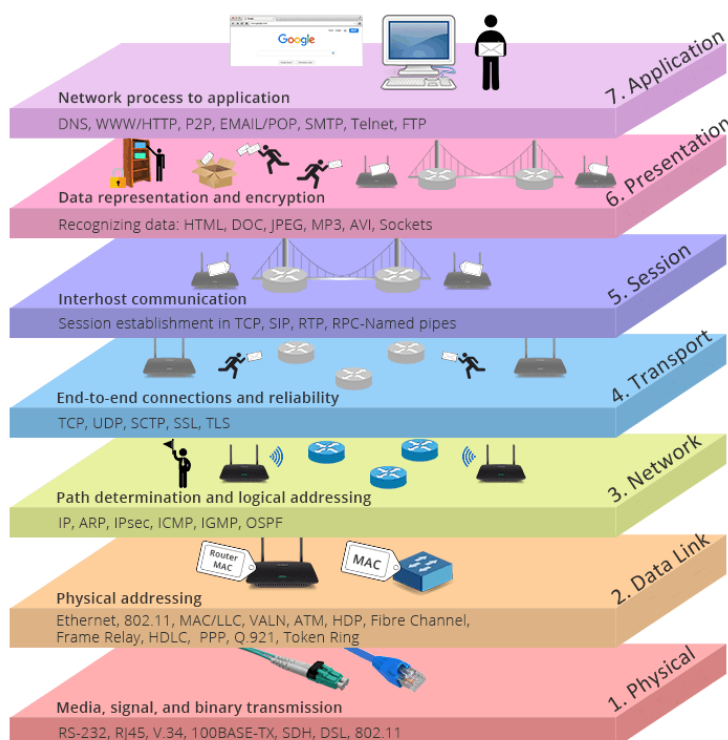


Abb. 2.3.5-1: Die sieben Layer des OSI-Modells. [26]

2.3 Feldbussysteme in der Automatisierungstechnik

2.3.1 Topologien

Bei Bussystemen können verschiedene Topologien verwendet werden, um die physische Verbindung der Geräte und Komponenten festzulegen. Abbildung Abb. 2.2.7-1: Netzwerktopologien bildet die am häufigsten verwendeten ab:

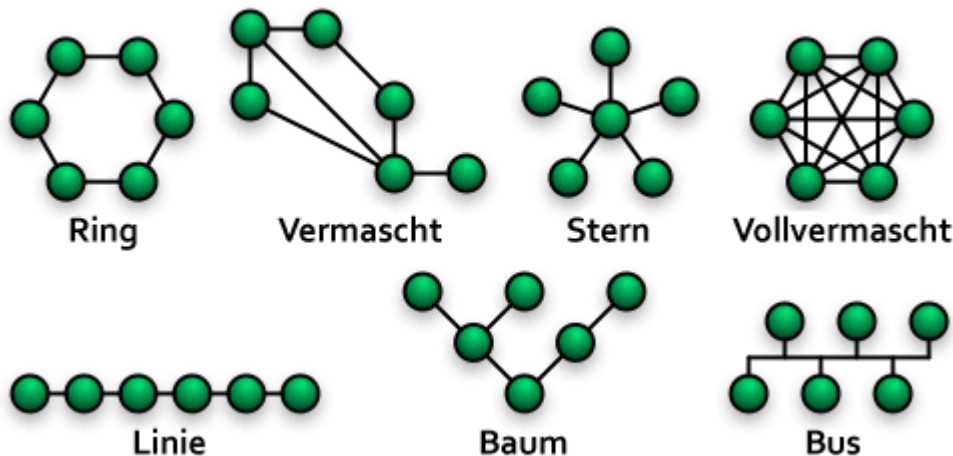


Abb. 2.2.7-1: Netzwerktopologien [27]

- **Ring-Topologie:** In einer Ring-Topologie sind die Geräte in einem geschlossenen Ring miteinander verbunden. Daten werden entlang des Rings von einem Gerät zum Nächsten übertragen, bis sie ihr Ziel erreichen.
- **Stern-Topologie:** In einer Stern-Topologie sind alle Geräte mit einem zentralen Knotenpunkt (Hub oder Switch) verbunden. Alle Datenkommunikation erfolgt über diesen zentralen Knotenpunkt, der die Daten an die entsprechenden Geräte weiterleitet.
- **Mesh-Topologie:** In einer Mesh-Topologie (Vermascht und Vollvermascht) sind alle Geräte miteinander verbunden, sodass mehrere Kommunikationspfade zwischen den Geräten existieren. Dies bietet *Redundanz* und erhöhte Zuverlässigkeit, da bei Ausfall eines Pfades alternative Pfade genutzt werden können.
- **Linientopologie (Bus-Topologie):** In einer Linientopologie sind alle Geräte in einer Reihe miteinander verbunden, wobei ein gemeinsames Kommunikationskabel verwendet wird. Daten werden von einem Endgerät zum anderen übertragen und jedes Gerät auf der Linie kann die Daten empfangen und entscheiden, ob sie relevant sind.
- **Baum-Topologie:** Die Baum-Topologie ähnelt der Stern-Topologie, hat jedoch hierarchische Ebenen. Ein zentrales Gerät verbindet sich mit mehreren untergeordneten Knoten, die wiederum weitere untergeordnete Knoten haben können. Dies schafft eine hierarchische Struktur für die Datenübertragung. [28]

2.3.1.1 Buszugriffsverfahren

Buszugriffsverfahren sind in Busstrukturen notwendig, um den Datenverkehr zu organisieren. Ohne solche Verfahren würden die Kommunikationen der Teilnehmer kollidieren, was einen reibungslosen Betrieb des Netzwerks unmöglich machen würde. Eine wichtige Unterscheidung bei Buszugriffsverfahren betrifft die Determiniertheit der Kommunikation. In kontrollierten oder deterministischen Verfahren wird der Sender vor der Datenübertragung eindeutig festgelegt. Diese Festlegung kann entweder zentral (Master/Slave) oder dezentral (Tokenbus oder -ring) erfolgen. Dadurch ist die maximale Zykluszeit bei begrenztem Sendeumfang berechenbar und spricht von Echtzeitfähigkeit. In nicht-deterministischen oder zufälligen Verfahren erfolgt der Zugriff bei Bedarf. In einem weit verbreiteten Verfahren wird zunächst überprüft, ob das Übertragungsmedium belegt ist (Carrier Sense - CS). Wenn die Leitung ausreichend frei ist, kann die Datenübertragung erfolgen. Bei Kollisionen muss die Übertragung wiederholt werden (Multiple Access - MA). Dieses Verfahren wird als CSMA abgekürzt. Es gibt verschiedene Möglichkeiten, mit Kollisionen umzugehen oder sie zu vermeiden. Collision Detection (CD), wie es zum Beispiel im Ethernet implementiert ist, ist eine verbreitete Methode. [29]

2.3.2 Profibus

Profibus steht für "Process Field Bus" und ist ein weit verbreitetes Bussystem in der industriellen Automatisierung und Prozessautomatisierung. Profibus wurde entwickelt, um die Kommunikation zwischen verschiedenen Geräten und Steuerungssystemen in industriellen Anwendungen zu erleichtern. Es handelt sich um ein offenes Kommunikationsprotokoll, das von der Profibus & Profinet International (PI) Organisation standardisiert und weiterentwickelt wird. Profibus ermöglicht die Übertragung von Daten zwischen Sensoren, Aktoren, Steuerungen und anderen Geräten in Echtzeit. Dieses Bussystem wird in verschiedenen Industriezweigen eingesetzt, darunter die Fertigungsindustrie, die Prozessindustrie, die Automobilindustrie und viele andere. Profibus kann sowohl in Punkt-zu-Punkt-Verbindungen als auch in komplexen Netzwerktopologien wie Linien-, Stern- und Baumstrukturen implementiert werden. Es gibt verschiedene Varianten von Profibus, darunter **Profibus DP** (für schnelle Kommunikation in der Fabrikautomation), **Profibus PA** (für die Prozessautomatisierung in der Prozessindustrie) und **Profibus FMS**. Letztere Ausführung hat in vielen Bereichen an Bedeutung verloren, weshalb sie hier nicht weiter beleuchtet wird. [30]

2.3.2.1 Profibus DP

Profibus DP (Dezentraler Peripheriebus) ist eine Variante des Profibus-Protokolls, das speziell für die schnelle Echtzeitkommunikation zwischen zentralen Steuerungen und dezentralen Peripheriegeräten in industriellen Anwendungen entwickelt wurde. Es bietet hohe Datenübertragungsraten, *deterministische Kommunikation* und erweiterte Diagnosefunktionen und wird häufig in der Fabrikautomation und anderen industriellen Umgebungen eingesetzt. [31]

2.3.2.2 Profibus PA

Profibus PA (Process Automation) ist eine Variante des Profibus-Protokolls, die speziell für die Prozessautomatisierung entwickelt wurde. Sie bietet sichere und zuverlässige Kommunikation in anspruchsvollen Umgebungen wie der chemischen und petrochemischen Industrie sowie in anderen prozessorientierten Anwendungen. Profibus PA unterstützt intrinsische Sicherheit, präzise Zeitsynchronisation und die Integration von Feldgeräten. [31]

2.3.3 Profinet

Profinet (Process Field Network) ist ein Industrial Ethernet-Kommunikationsprotokoll, das in der industriellen Automatisierung weit verbreitet ist. Es ermöglicht die Echtzeitkommunikation, Flexibilität und Integration von Geräten und Systemen in industriellen Anwendungen. Mit Profinet können Unternehmen ihre Produktionsprozesse optimieren und verschiedene Geräte in einem Netzwerk miteinander verbinden. [32]

2.3.4 AS-Interface

AS-i steht für Aktor-Sensor-Interface, ein einfaches Bussystem für die Kommunikation zwischen Aktoren und Sensoren in industriellen Anwendungen. Es verwendet Bit-Signale und besteht aus einem Busmaster und Bus-Slaves. Der Busmaster übernimmt die Kommunikationsfunktion, während ein Steuerungsgerät die Steuerungslogik enthält. AS-i ermöglicht verschiedene Verbindungsarten, einschließlich *Stand-alone-Lösungen*, Integration in SPS-Systeme und Anbindung an übergeordnete Feldbussysteme oder Netzwerke. Es bietet eine kostengünstige Lösung für die Kommunikation und Steuerung von Geräten. [33]

Eine Besonderheit stellen die ASi-Kabel mit ihrer ungewöhnlichen Form dar. Während Kabel in der Regel rund sind, hat das ASi-Kabel ein flaches Profil (siehe Abb. 2.3.4-1: Asi-Kabel und Durchdringungstechnik (1)). Dieses ungeschirmte 2-adrige Kabel überträgt nicht nur Daten und Energie, sondern ermöglicht auch die mühelose und kostengünstige Installation von ASi im Feld. [34]

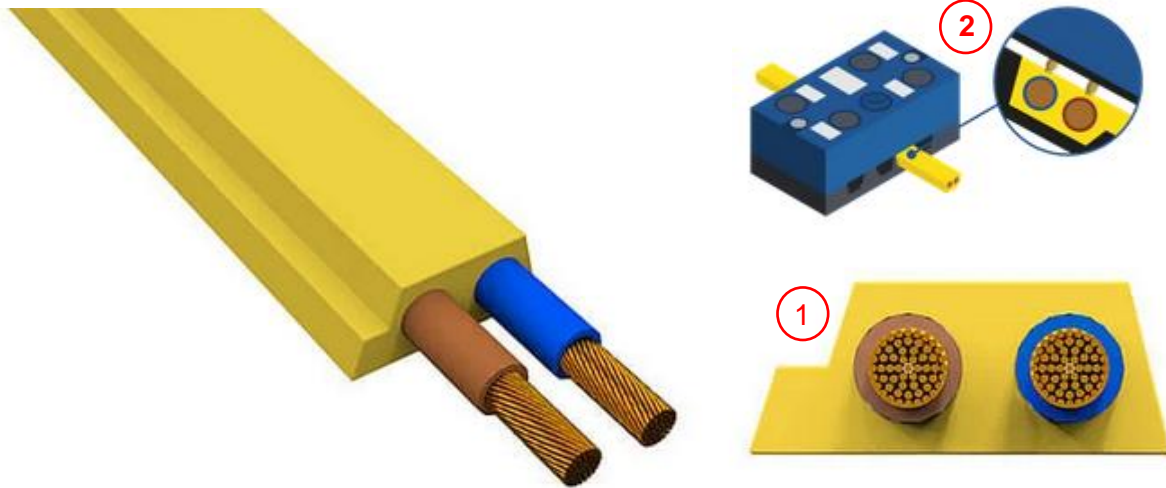


Abb. 2.3.4-1: Asi-Kabel und Durchdringungstechnik [34]

Durch die Durchdringungstechnik (vgl. Abb. 2.3.4-1: Asi-Kabel und Durchdringungstechnik (2)) können die Teilnehmer problemlos an das profilierte und verpolungssichere ASi-Kabel montiert werden. Dies gewährleistet eine perfekte Kontaktierung und erfüllt die Schutzart IP67. Zudem ist es problemlos möglich, die Geräte mit zusätzlicher 24-Volt-Versorgung zu betreiben. Dadurch ist AS-Interface ideal für energieintensive Anwendungen wie *IO-Link-Geräte* oder Motoren in der Fördertechnik. [34]

2.4 Digitale Zwillinge

Die Verfügbarkeit von Daten und Informationen ist entscheidend für die Entwicklung und Bereitstellung digitalisierter Produktsysteme. Unternehmen stehen jedoch vor Herausforderungen durch ständige Veränderungen in der Informationstechnik. Digitale Zwillinge sind eine Lösung, die die Kommunikationsfähigkeit von Produktsystemen nutzt und Daten aus dem gesamten Produktlebenszyklus aggregiert. Sie repräsentieren physische Produkte digital und ermöglichen die Optimierung von Entwicklung, Produktion, Nutzung und Recycling sowie die Schaffung neuer Geschäftsmodelle. Digitale Zwillinge bieten konkrete Vorteile in verschiedenen Lebenszyklusphasen, von der Gestaltung bis zur Wiederverwendung von Komponenten.

Die Einführung von digitalen Zwillingen erfordert ein strukturiertes Vorgehen. Zunächst ist es wichtig, das Konzept der digitalen Zwillinge zu verstehen und das Potenzial für das Unternehmen zu erkennen. Dann erfolgt die Entwicklung eines Zielbilds, bei dem der Mehrwert für das Unternehmen festgelegt wird. Die Umsetzung erfolgt in der Entwicklungsphase, wobei Designelemente hinzugefügt werden und das bestehende Unternehmensumfeld angepasst wird. Digitale Zwillinge können in verschiedenen Phasen des Produktlebenszyklus eingesetzt werden. Ebenso wichtig und grundlegend bei der Entwicklung und Bereitstellung digitaler Zwillinge ist die Berücksichtigung relevanter Richtlinien und Gesetze (vgl. [35] S.18ff).

Eine allgemeingültige, allesumfassende Definition für den digitalen Zwilling gibt es noch nicht, es lässt sich jedoch aus den vielfältigen theoretischen Ansätzen eine Kernaussage über sie formulieren:

Ein digitaler Zwilling ist eine virtuelle Darstellung eines realen Objekts oder Elements. Dieses Objekt kann physischer Natur sein, wie beispielsweise eine Maschine oder ein Gebäude, oder virtueller Natur, wie ein Plan oder ein Prozess (vgl. [36] S. 6).

Sowohl das Internet der Dinge (IoT) als auch der digitale Zwilling werden oft in Zusammenhang betrachtet, da sie auf den grundlegenden Prinzipien von Industrie 4.0 basieren, nämlich Vernetzung, Informationstransparenz und dezentrale Entscheidungsfindung. Dennoch fokussieren sie unabhängig voneinander verschiedene Anwendungsbereiche. Das IoT konzentriert sich auf die Vernetzung von Geräten und die Kommunikation über das Internet, wobei es darum geht, die grundlegenden Fähigkeiten dieser vernetzten *Assets* zu beschreiben.

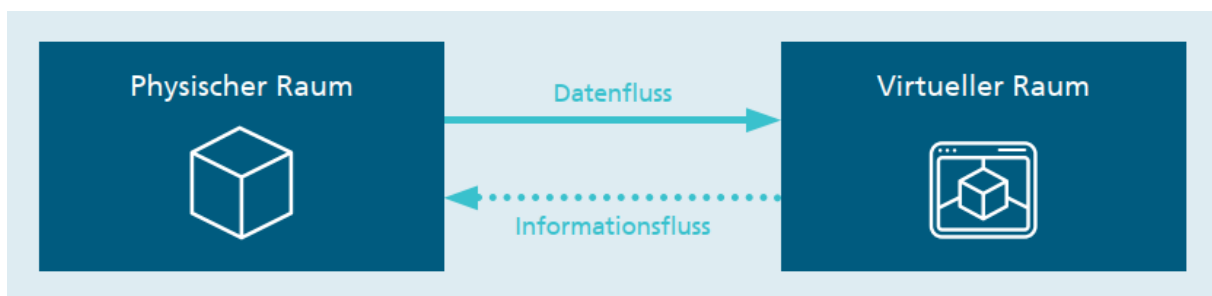


Abb. 2.4.-1: idealisierte Darstellung des Verhältnisses von physischem und virtuellem Raum [36]

Der digitale Zwilling hingegen widmet sich der Schaffung von Informationstransparenz und der Sammlung, Verwaltung, Modellierung und Organisation von Daten. Auf konzeptioneller Ebene stellt der digitale Zwilling eine direkte Verbindung zwischen der realen Welt und ihrer digitalen Repräsentation her, indem er Objekte so in die virtuelle Welt überträgt, dass Nutzer auf sie zugreifen und sie steuern können (siehe Abb. 2.4-1). Dies erfordert die Umsetzung von Abbildungsregeln, die den standardisierten Prinzipien der Informations- und Datenmodellierung entsprechen, um einen nahtlosen Daten- und Informationsfluss zwischen beiden Welten sicherzustellen. [36]

2.4.1 Anwendung

Digitale Zwillinge bieten vielfältige Anwendungen, insbesondere im Anlagenbau. Sie ermöglichen detaillierte Modellierung von Maschinen bis hin zu einzelnen Antrieben, was präzise Vorhersagen für das Verhalten in realen Umgebungen ermöglicht. Dies erleichtert die effiziente Planung und teilweise virtuelle Inbetriebnahme von Produktionsstraßen, wodurch wertvolle

Ressourcen eingespart werden. Während des laufenden Anlagenbetriebs liefern Sensordaten wichtiges Feedback über den Zustand von Maschinen und Prozessen und ermöglichen Einblicke durch einfache Visualisierungen. Ähnliche Erkenntnisse werden durch Gebäudemessungen gewonnen, wodurch das tatsächliche Nutzerverhalten besser verstanden und Sanierungsmaßnahmen gezielt geplant werden können. Digitale Zwillinge sind auch für Geschäftsprozesse wertvoll, insbesondere wenn Modelle aus verschiedenen Bereichen, wie Produktionsanlagen und ERP-Systemen, verknüpft werden. Dies ermöglicht die Entdeckung neuer Zusammenhänge und verbessert die Genauigkeit von Vorhersagen, beispielsweise zur Berechnung der Herstellungskosten eines Produkts für eine optimierte Angebotserstellung. [37]

2.4.2 Die Schlüsseltechnologien

Digitale Zwillinge sind auf Synergien mit anderen Technologien angewiesen. Die Rechenleistung spielt eine entscheidende Rolle, wobei sowohl Cloud-Ressourcen als auch leistungsschwächere Hardware an den Rändern des Datennetzwerks (Edge Computing) genutzt werden können. Die Kommunikationstechnologie ist ebenfalls von großer Bedeutung, da sie die Verbindung zwischen dem digitalen Modell und der realen Welt ermöglicht. Hierbei kommen spezialisierte Kommunikationsprotokolle auf Hardware- und Softwareebene zum Einsatz, darunter Transportprotokolle, Industriestandards wie OPC UA, sowie webbasierte Protokolle wie JSON (JavaScript Object Notation) und XML (Extensible Markup Language). Mit der steigenden Komplexität von digitalen Zwillingen werden auch die generierten Ergebnisse komplexer. Die Auswertung erfordert zunehmend Methoden aus dem Bereich *Data Science* und maschinelles Lernen, um Zusammenhänge besser zu verstehen. [37]

2.4.3 Eine erste Einschätzung

Auf Basis des studierten Quellenmaterials erfolgt an dieser Stelle ein erstes Statement zu digitalen Zwillingen. Denn neben den zahlreichen Vorteilen, die die Nutzung solcher virtuellen Abbilder mit sich bringt, können gewisse Risiken und besondere Herausforderungen nicht vernachlässigt werden.

Ein wesentliches Risiko betrifft den Datenschutz und die Datensicherheit. Die Erfassung und Verarbeitung von Daten für den digitalen Zwilling können Datenschutzbedenken aufwerfen, da sensible oder personenbezogene Informationen gefährdet sein können. Ebenfalls erfordern die Modellierung und Integration verschiedener Datenquellen Fachwissen und Ressourcen, was die Implementierung komplex gestalten kann. Dies kann wiederum die Kosten in die Höhe steigen lassen, insbesondere wenn umfangreiche Sensorik oder leistungsstarke Hardware erforderlich sind. Die Qualität der verwendeten Daten ist ebenfalls von entscheidender Bedeutung, da fehlerhafte oder unvollständige Daten die Genauigkeit und Zuverlässigkeit des

digitalen Zwillingen beeinträchtigen können. Die Abhängigkeit von Technologie, mögliche Unterbrechungen in der Datenübertragung und Hardwareausfälle können den Betrieb stören. Die Interoperabilität stellt eine weitere Herausforderung dar, da die Integration verschiedener Systeme und Datenquellen erfordert, dass unterschiedliche Datenformate und Protokolle miteinander kommunizieren können.

Auch Sicherheitsrisiken sind nicht zu vernachlässigen, da unzureichend geschützte digitale Zwillinge bspw. Ziel von Cyberangriffen werden können, was die Integrität der Daten und die Sicherheit des physischen Systems gefährdet. Schlussendlich können auch Rechtsfragen im Zusammenhang mit digitalen Zwillingen auftreten, insbesondere in Bezug auf Haftung und Verantwortlichkeit im Falle von Fehlfunktionen oder Schäden. Daher ist es von großer Bedeutung, diese Risiken sorgfältig zu berücksichtigen und geeignete Maßnahmen zur Risikominimierung und Sicherheit zu ergreifen.

3 Softwarelösungen

Der steigende Druck bezüglich Projektlieferzeiten, die Notwendigkeit einer höheren Anlagenrentabilität und der Mangel an hochqualifiziertem Personal sind Herausforderungen, die nahezu alle Industriezweige betreffen. Die Echtzeitsimulation von Automatisierungsprojekten hat sich als äußerst effiziente Lösung für diese und weitere Anforderungen bewährt. [38] Im Folgenden werden die Softwarepakete SIMIT und Factory IO näher beleuchtet, um am Ende dieser Arbeit eine Gegenüberstellung durchführen zu können.

3.1 SIMIT

Die von der Siemens AG entwickelte Softwarelösung SIMIT (Simulation of Industrial Management and Training, zu Deutsch Simulation von Industriemanagement und Schulung) spielt eine entscheidende Rolle in der Umsetzung der Digitalisierungsstrategie des Unternehmens. Sie dient zur Darstellung von digitalen Verhaltensmodellen und gewinnt zunehmend an Bedeutung in diesem Bereich. SIMIT ist eine kostenintensive, dennoch äußerst leistungsstarke Softwareplattform, deren Anwendungsbereiche weit über die Modellierung von digitalen Verhaltensmodellen hinausgehen.

Besonders bemerkenswert ist die Fähigkeit von SIMIT, wenn mehrere Softwarelösungen von Siemens in einer Simulationsumgebung eingesetzt werden, die Rolle des zentralen Taktgebers zu übernehmen. Dies ermöglicht selbst bei komplexen Anlagen eine zeitsynchrone Simulation zwischen den Steuerungen und dem geometrischen digitalen Zwilling. [39]

Die Bibliothekselemente von SIMIT sind zwar auf Komponenten von Siemens beschränkt, jedoch besteht die Möglichkeit, eigene Komponenten zu modellieren und mit diesen zu arbeiten. [39]

Mit SIMIT können umfassende Tests von Automatisierungsprojekten durchgeführt werden, ebenso wie die virtuelle Inbetriebnahme von Systemen, Maschinen und Prozessen. Zudem kann die Simulationsplattform zur Schulung des Bedienpersonals in realitätsnahen Trainingsumgebungen genutzt werden. Abbildung 3.1-1: „mögliche Einbindung von SIMIT“ zeigt beispielhaft welche Szenarien mit SIMIT realisiert werden können.

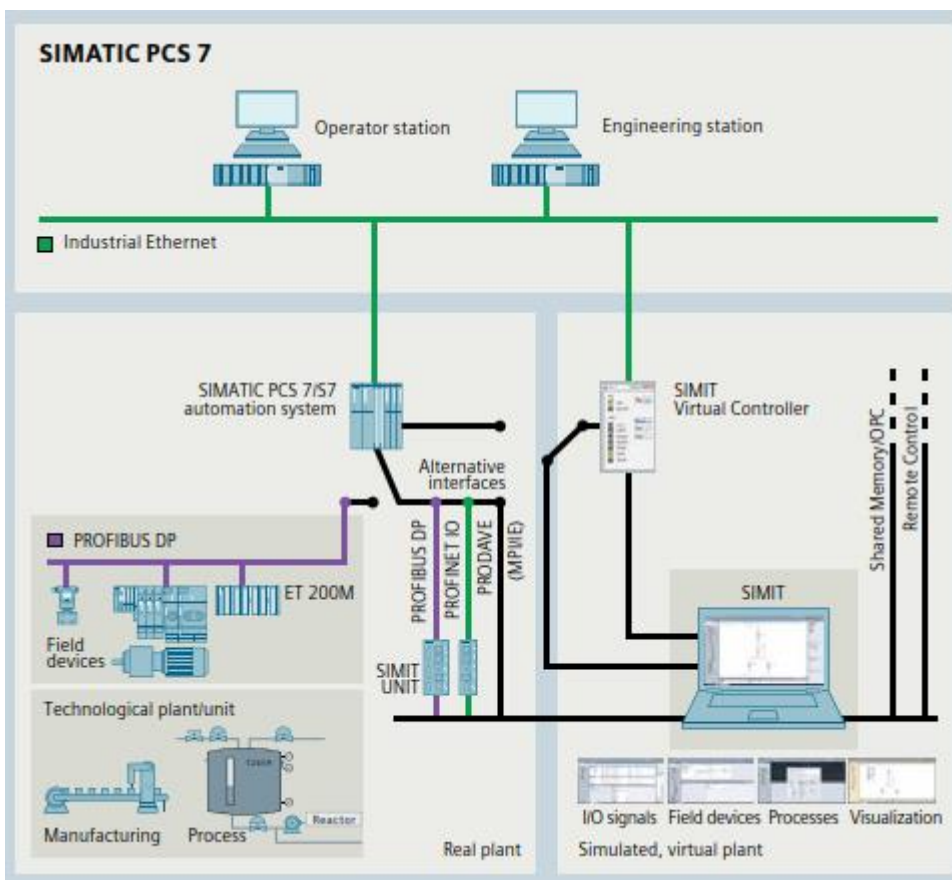


Abb. 3.1-1: mögliche Einbindung von SIMIT [38]

3.1.1 SIMIT Simulation Plattform (SIMIT SP)

SIMIT ermöglicht umfassende Tests von Automatisierungsprojekten, die virtuelle Inbetriebnahme von Systemen, Maschinen und Prozessen sowie Schulungen des Bedienpersonals in realitätsnahen Umgebungen.

Die Plattform zeichnet sich laut der Siemens AG durch eine hohe Benutzerfreundlichkeit, vereinfachte Simulationsmodellierung, eine hohe Leistungsfähigkeit, Offenheit und Flexibilität aus. Die virtuelle Inbetriebnahme mit SIMIT trägt dazu bei, die Effizienz im Engineering zu

steigern und die Planungssicherheit zu erhöhen. Dies wird durch die Möglichkeit erreicht, Tests unter Verwendung der originalen Automatisierungsprogramme durchzuführen, ohne auf reale Hardware angewiesen zu sein.

Durch SIMIT kann eine hohe Qualität im Engineering, nahtlose Integration von Engineeringdaten, eine beschleunigte Inbetriebnahme und gesteigerte Anlagenverfügbarkeit realisiert werden. Zudem ermöglicht SIMIT sichere Schulungen des Anlagenpersonals und einen reibungslosen Transfer von Fachwissen in Trainingseinheiten. [38]

SIMIT unterstützt die Wiederverwendung bereits vorhandener Daten und Modelle aus anderen Simulationswerkzeugen. Die Benutzeroberfläche ist benutzerfreundlich, Anpassungsmöglichkeiten sind flexibel, und die Plattform ist offen für Systemintegration.

Die Nutzung von SIMIT als virtuelle Trainingsumgebung eröffnet diverse Anwendungsmöglichkeiten für die Schulung des Anlagenpersonals, ohne den laufenden Anlagenbetrieb zu stören. Die folgende Abbildung (Abb. 3.1.1-1: Bedienoberfläche) visualisiert die Benutzeroberfläche von SIMIT. [38]

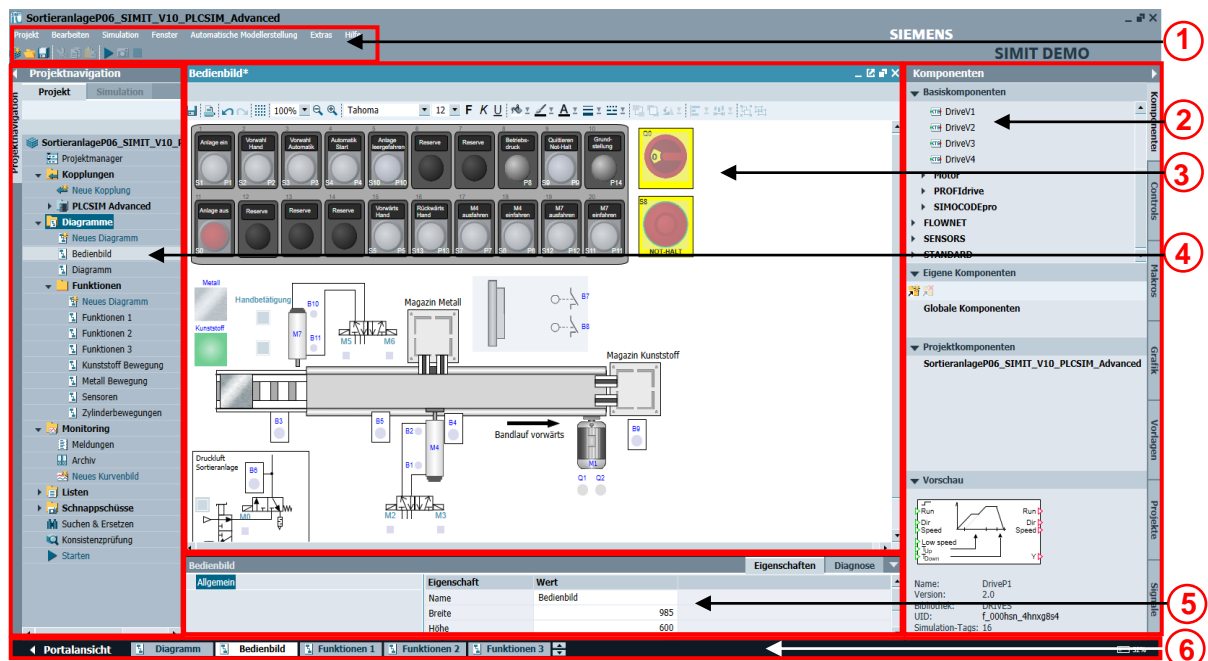


Abb. 3.1.1-1: Bedienoberfläche

Die SIMIT-Benutzeroberfläche umfasst mehrere Schlüsselemente zur effizienten Steuerung und Bearbeitung von Projekten.

Die Menüleiste und Symbolleiste (1) bieten Zugang zu den verschiedenen Funktionen und Einstellungen von SIMIT, wobei die Symbolleiste speziell häufig genutzte Funktionen über Symbole und Schaltflächen bereitstellt. Task-Cards (zu Deutsch Aufgabenkarten) (2) dienen der Auflistung von Objekten, die im aktuellen Editor verwendet werden können und nach thematischen Gesichtspunkten organisiert sind. Der Arbeitsbereich (3) stellt den

Hauptarbeitsbereich dar, in dem die Editoren zur Projekt- und Aufgabenbearbeitung geöffnet werden. Die Projektnavigation (4) präsentiert das aktuelle Projekt in einer Baumansicht, was die Navigation und den Zugriff auf Projektelemente erleichtert. Das Eigenschaftsfenster (5) zeigt detaillierte Informationen zu ausgewählten Objekten und ermöglicht die Anpassung ihrer Eigenschaften. Die Editorleiste und Statuszeile (6) ermöglichen den Wechsel zwischen geöffneten Editoren und den Zugang zur Portalansicht sowie die Anzeige des aktuellen Anwendungsstatus.

Zudem bietet SIMIT die Möglichkeit, den Arbeitsbereich zu teilen, um zwei Editoren parallel anzuzeigen und somit die Bearbeitung und den Vergleich von Projektdaten zu erleichtern (vgl. [40] S.28f).

3.1.2 SIMIT Unit

Die SIMIT UNIT bietet die Möglichkeit, die Automatisierung bereits in der Engineering-Phase zu testen, um einen effizienteren Workflow im Engineering zu realisieren. Mithilfe der SIMIT UNIT ist es möglich, das vollständige Verhalten einer Anlage in Echtzeit zu simulieren, unabhängig von der physischen Peripherie der Anlage. Dieser Ansatz trägt zur Steigerung der Qualität der Steuerungsprogramme bei und führt durch die frühzeitige Erkennung von Fehlern zu Zeit- und Kosteneinsparungen. Infolgedessen werden sowohl die Sicherheit von Mensch und Maschine als auch ein beschleunigter Produktionsstart sichergestellt.

In einer *Hardware-in-the-Loop*-Konfiguration wird das Automatisierungsprogramm in das tatsächliche Automatisierungssystem integriert (siehe Abb. 3.1-1: mögliche Einbindung von SIMIT). Die SIMIT UNIT ermöglicht die Simulation der I/O-Signale der dezentralen Peripherie sowie der Signale von Feldgeräten wie Aktoren und Sensoren. Dieser Ansatz unterstützt die präzise Prüfung und Verfeinerung von Steuerungsprogrammen und trägt zur Gewährleistung eines sicheren und effizienten Betriebs bei Produktionsanlagen bei.

Durch die Kopplung der "SIMIT Unit" wird die Kommunikation von SIMIT mit einem oder mehreren der folgenden Bussysteme ermöglicht:

- PROFINET IO-Controller
- PROFIBUS DP-Master

Je nach den Anforderungen des Automatisierungsprojekts kann das benötigte Gerät ausgewählt werden. Hier kommt die SIMIT UNIT PN für die Kommunikation über PROFINET und die SIMIT UNIT PB für die Kommunikation über PROFIBUS in Frage. Dies ermöglicht eine flexible Integration und Kommunikation im Automatisierungssystem. [41]



Bussystem	SIMIT UNIT Variante	Anschlussmöglichkeiten	Gerät
PROFIBUS	SIMIT UNIT PB	<ul style="list-style-type: none"> - 3 Ethernet Ports - 3 elektrische Ethernet RJ45-Buchsen - 1 USB Master Anschluss - 1 Klemmenblock 4-polig für Versorgungsspannung + ein Anschluss für digitalen Ein/Ausgang - 1 Klemmenblock 4-polig für Anschluss zweier digitaler Ein/Ausgänge - 2 SUB D-Buchsen 9-polig für Anschluss zweier Profibus-Stränge 	
PROFINET	SIMIT UNIT PN	<ul style="list-style-type: none"> - 3 Ethernet Ports - 3 elektrische Ethernet RJ45-Buchsen - 1 USB Master Anschluss - 1 Klemmenblock 4-polig für Versorgungsspannung + ein Anschluss für digitalen Ein/Ausgang 	

Tabelle 1: Typen der SIMIT Unit [41]

3.2 Factory IO

FACTORY I/O ermöglicht die konstruktive Modellierung und Simulation dreidimensionaler Anlagen mittels vordefinierter Anlagenteile. Die Simulation wird durch die Integration von hochwertiger 3D-Grafik, einer leistungsfähigen *Physik-Engine* und Echtzeitaudioeffekten geprägt, wodurch ein beeindruckend realitätsnahes Umfeld geschaffen wird.

Darüber hinaus unterstützt FACTORY I/O die nahtlose Integration mit verschiedenen SPS-Steuerungen, darunter Allen-Bradley, Automgen, MHJ und Siemens. Das System basiert auf einem Baukastenprinzip, das eine umfangreiche Sammlung von über 80 Bauteilen bereitstellt, darunter Transportbänder, Rollenbahnen, Schwingarmsortierer, Zylinder, Drehplatten, Sensoren (kapazitiv und induktiv), Lichtschranken, Hebewerke, "Pick and Place" Roboter, Hochregale, Regalbediengeräte und viele andere. Diese Bauteile verfügen größtenteils über sogenannte Tags, die während des Aufbaus manuell beeinflusst werden können. Die Montage und Konfiguration der Bauteile erfolgt durch einfaches Ziehen und Ablegen (Drag-and-Drop) in eine

zunächst leere Industriehalle. Diese Bauteile können, falls erforderlich, mit Sensoren ausgestattet werden, um bestimmte Funktionalitäten zu simulieren.

Verschiedene Kameraansichten, wie Orbit, Flug und First-Person, ermöglichen es dem Nutzer, die Anlage aus verschiedenen Perspektiven zu betrachten und zu analysieren. Die Ausbilder-Konsole bietet die Möglichkeit, die Anlage durch ein Passwort zu schützen, um unautorisierte Änderungen zu verhindern. Zusätzlich können bewusst Fehler in Sensoren und Aktoren eingebaut werden, um spezifische Fehlerzustände zu simulieren.

Weitere Funktionen von FACTORY I/O umfassen die Verwendung vorgefertigter und editierbarer Maschinen, eine umfangreiche Auswahl an Maschinenteilen, einen Zeitlupenmodus zur detaillierten Untersuchung von Simulationen, die gezielte Fehlererkennung und -behebung in Sensoren und Aktoren sowie die Möglichkeit, erstellte Maschinen in einer einzigen Datei zu speichern und zu teilen. [42] [43]

4 Die Förderstrecke

4.1 Einführung

Dieses Projekt beschäftigt sich mit der Erstellung und Programmierung einer industriellen Anlage in Factory IO, die mittels CODESYS und OPC UA-Kommunikation gesteuert wird. Diese Anlage simuliert eine digitale Förderstrecke (siehe Abb. 4.1-1: Die Förderstrecke in Factory IO), wie sie in der physischen Industriewelt existieren könnte. Das Hauptziel dieser Anlage ist es, verschiedene Kartongrößen mithilfe von Sensorik zu erfassen und an die entsprechenden Einheiten zur weiteren Verarbeitung zu verteilen.

Die Anlage besteht aus mehreren Schlüsselkomponenten, die in diesem Kapitel näher erläutert werden. Im ersten Teil der Anlage, der als Sortiereinheit 1 (1) bezeichnet wird, wird der Prozess über das Bedienpult gestartet. Außerdem werden Kartons erzeugt und die erste Verteilung findet statt. Hierbei gibt es zwei mögliche Wege für die Kartons, ein Weg führt zum Palettierer und der andere Weg führt zur Sortiereinheit 2. Die integrierte Waage gibt bei Belastung ein analoges Ausgangssignal aus, welches an den Rollensortierer zur Weiterverteilung der Kartons übergeben wird. Vor der Verteilung in Sortiereinheit 2 passieren alle Kartons einen ersten RFID-Sensor. Dieser Sensor beschreibt die RFID-Etiketten der Kartons "M" und "L" mit jeweils einem festen ganzzahligen Wert.

Die zweite Einheit, der Palettierer (2), ist für die Verarbeitung der kleinsten Kartongröße verantwortlich. Diese Kartons werden mittels eines Schiebers auf eine Tür geschoben. Sobald eine Ebene fertiggestellt ist, öffnet sich die Tür und die Ebene landet auf einer Palette, die zuvor mithilfe eines integrierten Aufzugs unter besagter Tür platziert wurde. Nachdem drei

Ebenen auf die Palette gelegt wurden, ist diese fertig gepackt, der Aufzug fährt nach unten und die Palette wird wegtransportiert.

Der dritte Abschnitt, die Sortiereinheit 2 (3) verarbeitet die Kartongrößen "M" und "L", die zuvor von dem RFID-Sensor in Sortiereinheit 1 beschrieben wurden. Ein zweiter RFID-Sensor liest diese Werte aus und leitet die finale Verteilung ein.

Dieses Projekt ermöglicht die Anwendung und Vertiefung von Fähigkeiten, die während des Studiums erworben wurden. Die Kombination aus Factory IO, CODESYS-Programmierung und OPC UA-Kommunikation ermöglicht die Erstellung einer realitätsnahen Simulationsanlage, die einen Einblick in die Prozessautomatisierung und Steuerungstechnik in der Industrie bietet. Die Erfahrungen und Kenntnisse, die durch die Entwicklung dieses Projekts gewonnen wurden, können von erheblichem Nutzen für zukünftige Projekte und die berufliche Weiterentwicklung sein.

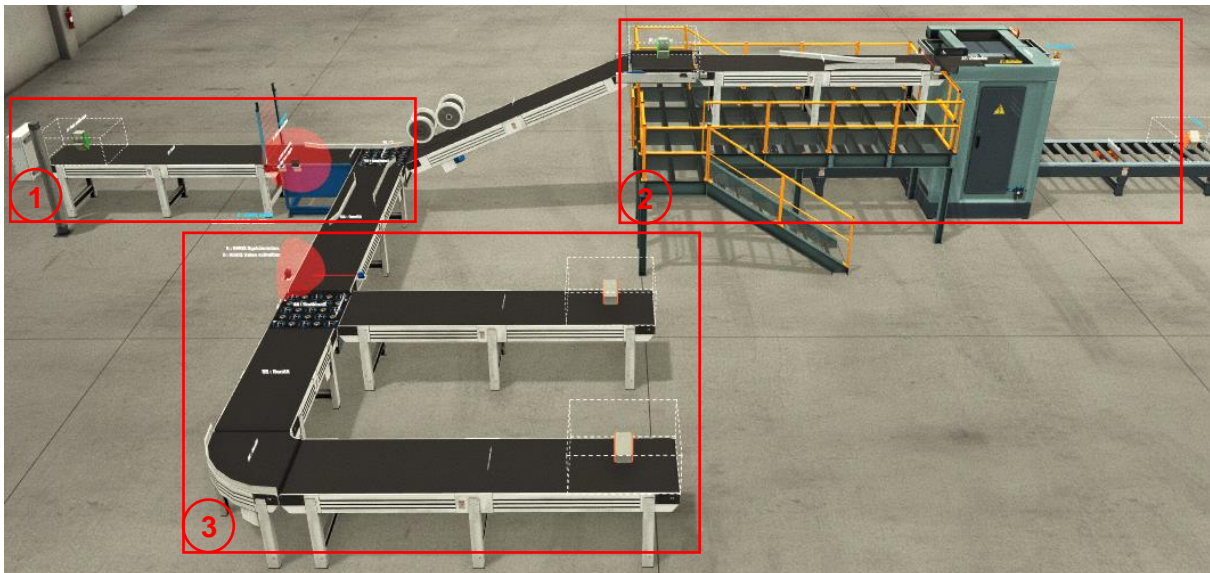


Abb. 4.1-1: Die Förderstrecke in Factory IO

4.2 Radio Frequency Identification-Chip (RFID-Chip)

RFID, oder Radio-Frequency Identification, bezeichnet eine Technologie zur automatisierten Identifikation und Lokalisierung von Objekten mithilfe von Funkwellen. Es existieren drei Hauptkategorien von RFID-Systemen, von denen jede charakteristische Vor- und Nachteile aufweist.

Die erste Kategorie, die passive RFID-Technologie, umfasst Lesegeräte und passive RFID-Tags (zu Deutsch RFID-Etiketten) ohne eigene Stromversorgung. Diese Tags nutzen das Abfragesignal des Lesegeräts zur Energiegewinnung und kommunizieren entweder induktiv oder

durch Rückstreuungskopplung. Die Vorteile dieser Technologie sind die kosteneffiziente Herstellung, jedoch liegt die Reichweite bei hohen Frequenzen nur bei etwa 0,6 bis 3 Meter und die Speicherkapazität ist begrenzt (vgl. [22] S. 9).

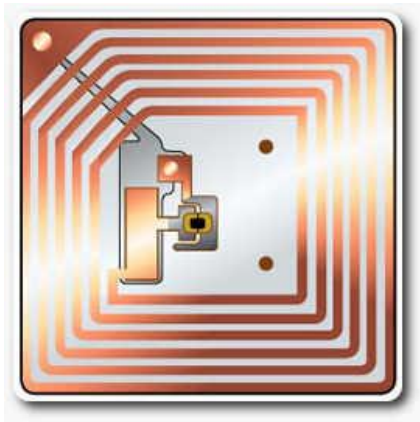


Abb. 4.2.-1: RFID-Tag passiv [44]

Semi-passive RFID-Systeme, die zur zweiten Kategorie gehören, zeichnen sich durch Tags aus, die zwar über eine Batterie, aber keinen aktiven Sender verfügen. Sie nutzen nach wie vor induktive und Strahlungskopplung zur Kommunikation und sind weniger anfällig für Störungen. Die Reichweite liegt bei maximal 25 Metern. Allerdings ist die Batterielebensdauer begrenzt und liegt in der Regel zwischen zwei und sechs Jahren. Zudem sind diese Tags auf Lesegeräte angewiesen (vgl. [22] S. 10).



Abb. 4.2.-2: RFID-Tag semi-passiv [45]

Die dritte Kategorie, die aktiven RFID-Systeme, weist eine komplexere Technologie auf. Diese Tags sind mit einer Batterie und einem aktiven Sender ausgestattet, was eine erweiterte Reichweite von bis zu 90 Metern ermöglicht. Zudem bieten sie die Integration von Sensoren und eine höhere Speicherkapazität. Nachteile dieser Technologie sind die höheren Kosten und die begrenzte Batterielebensdauer, die in der Regel zwischen drei und fünf Jahren liegt. Des Weiteren können aktive RFID-Systeme aufgrund ihrer Störbeeinflussung untereinander nicht

in der Lagerlogistik eingesetzt werden. Sie finden jedoch Anwendung in kostenintensiven Szenarien wie der Containerlogistik oder bei Lastkraftwagen (vgl. [22] S. 10f).



Abb. 4.2.-3: RFID-Tag aktiv13 [46]

Folgende Übersicht (Abb. 4.2-1: „Vor und Nachteile der unterschiedliche RFID-Typen“) fasst die Vor und Nachteile der einzelnen Systeme zusammen.

	Vorteile	Nachteile
Passive RFID	Kostengünstig wegen weniger Komponenten IC passiv Antenne	Maximale 60 cm-3 m Reichweite Begrenzter Speicher Starkes Lesegerät erforderlich Keine Sensoren möglich, da keine Batterie
Semi passive RFID	Mittlere Reichweite bis 15m Störungsarm Sensoren vernetzbar aufgrund von Batterie	Begrenzte Lebensdauer 2-6 Jahre Tag weniger robust wegen IC und Antennen
Aktive RFID	Reichweite bis 90 m Lesegerät mit geringem Stromverbrauch Aktive Tags haben stromführende Antenne Multisensorfähig, leistungsfähige Prozessoren	Teuer Begrenzte Lebensdauer 3-6 Jahre Gewicht und Größe (Batterie) Aktiver Sensor erzeugt Rauschen Für Lagerscans nicht geeignet

Abb. 4.2-4: Vor und Nachteile der unterschiedliche RFID-Typen [22]

Alle RFID-Tags arbeiten bei bestimmten Frequenzen, welche von der International Telecommunication Union (ITU, zu Deutsch internationale Fernmeldeunion) definiert und koordiniert werden. Die Tabelle zeigt die festgelegten Frequenzen.

Langwelle (LF)	Mittelwelle	Kurzwellen (HF)	UHF	SHF
125 kHz	375 kHz	13,65 MHz	Europa	
134 kHz	500 kHz		865 - 869 MHz	2,45 GHz
250 kHz	625 kHz		USA und Asien	5,8 GHz
	750 kHz		950 MHz	
	875 kHz			

Tabelle 2: RFID-Tag-Frequenzen [22]

4.3 Aufbau der Anlage

Zur Gewährleistung der Übersichtlichkeit wurde die Förderstrecke in drei Teilanlagen unterteilt, die im weiteren Verlauf genauer erläutert werden.

4.3.1 Sortiereinheit 1

In diesem Abschnitt der Anlage werden Kartons unterschiedlicher Größen, nämlich M und L, sowie solche vom Typ "Palletizing Box" mithilfe des sogenannten „Parts Emitter“ (dt. Teileerzeuger) erzeugt und die erste Verteilung findet statt. Dieser Prozess setzt sich aus der Funktionalität von fünf elementaren Baueinheiten (siehe Abb. 4.3.1-2: Sortiereinheit 1) zusammen, welche im Folgenden näher erläutert werden.

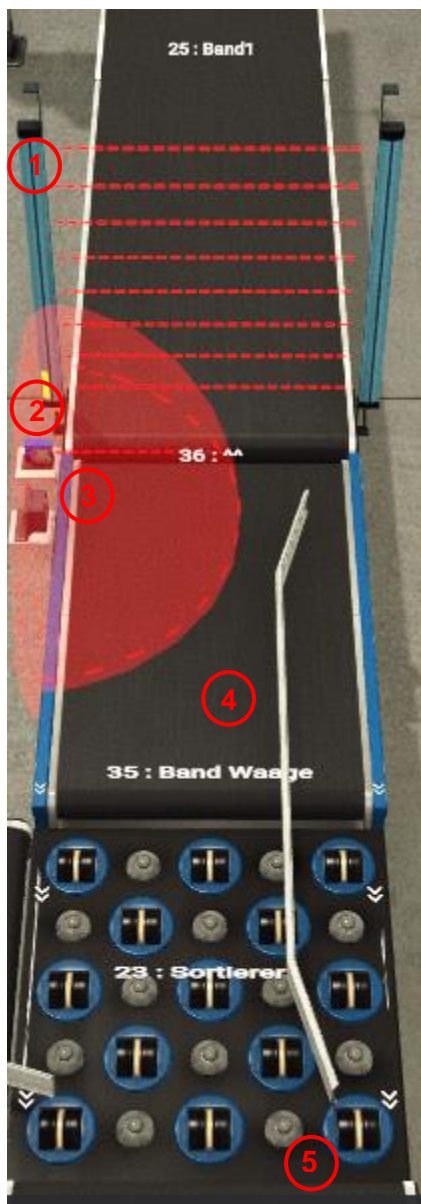


Abb. 4.3.1-2: Sortiereinheit 1

Primär durchläuft jeder Karton eine Lichtschranke (1), welche insgesamt acht Laserstrahlen emittiert, wobei jeder Strahl ein Binärbit repräsentiert. Sobald ein Karton die Strahlen unterbricht, erfolgt die Umwandlung der Binärbits von ihrer dualen in die dezimale Darstellung, woraufhin sie zusammengefasst und als Ganzzahl in der Benutzeroberfläche von Factory IO dargestellt werden. Mit acht Binärbits ergibt sich ein darstellbarer Wertebereich von 0 bis 255. In folgender Abbildung (Abb. 4.3.1-1: Lichtschranke mit Dual- und Dezimalcode und Summe des Dezimalcodes) wird die Lichtschranke im Zusammenhang mit einem Karton als Unterbrechungselement dargestellt, begleitet von den dazugehörigen dualen und dezimalen Kodierungen in diesem exemplarischen Fall.

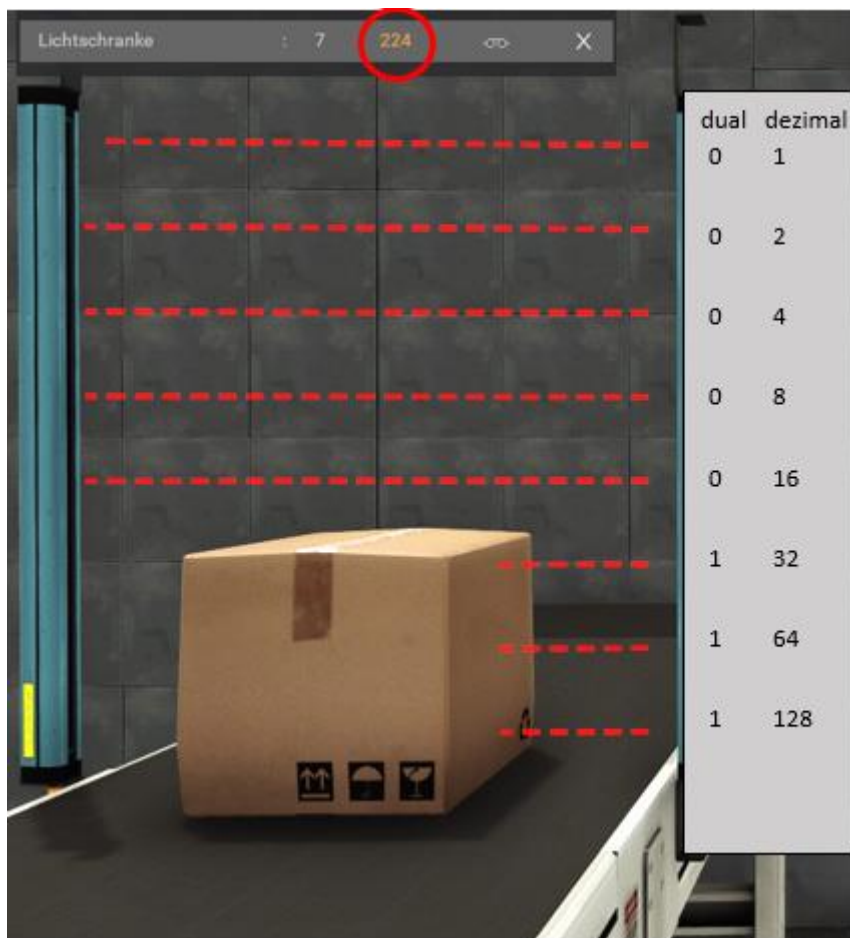


Abb. 4.3.1-2: Lichtschranke mit Dual- und Dezimalcode und Summe des Dezimalcodes

Wie der Abbildung zu entnehmen ist, unterbricht der Karton -in diesem Beispiel der Größe „L“- drei der Laserstrahlen, was zur Generierung des dualen Codes 1110 0000 führt. Nach der Umwandlung dieses Codes in eine dezimale Form und der anschließenden Addition ergibt sich der Wert 224. Dieser Prozess wird automatisch und im Hintergrund von Factory IO durchgeführt und in einem Anzeigefenster (siehe Abbildung 4.3.1-1, oberer roter Kreis) visualisiert.

Die Lichtschranke bildet die Grundlage der Funktionsweise des nächsten Sensors, dem RFID1 (3). Dieser Sensor soll die RFID-Etiketten der Kartons "M" und "L" für die Weiterverarbeitung beschreiben, um sie später zuordnen zu können. Der Sensor empfängt einen ganzzahligen Wert, der von der Lichtschranke erfasst wurde. Diese Information dient dazu, den Karton zu identifizieren und die geeignete Aktion auszuführen.

Die eigentliche Anweisung zur Beschreibung der RFID-Etiketten erfolgt jedoch nicht direkt von der Lichtschranke, sondern durch einen photoelektrischen Sensor (2), der sich unmittelbar in der Nähe befindet. Wenn dieser photoelektrische Sensor auslöst, wird der RFID1-Sensor aktiviert und führt den Schreibbefehl aus. Dieser Vorgang wird in Abbildung 4.3.1-2: "RFID-Sensor1" veranschaulicht.

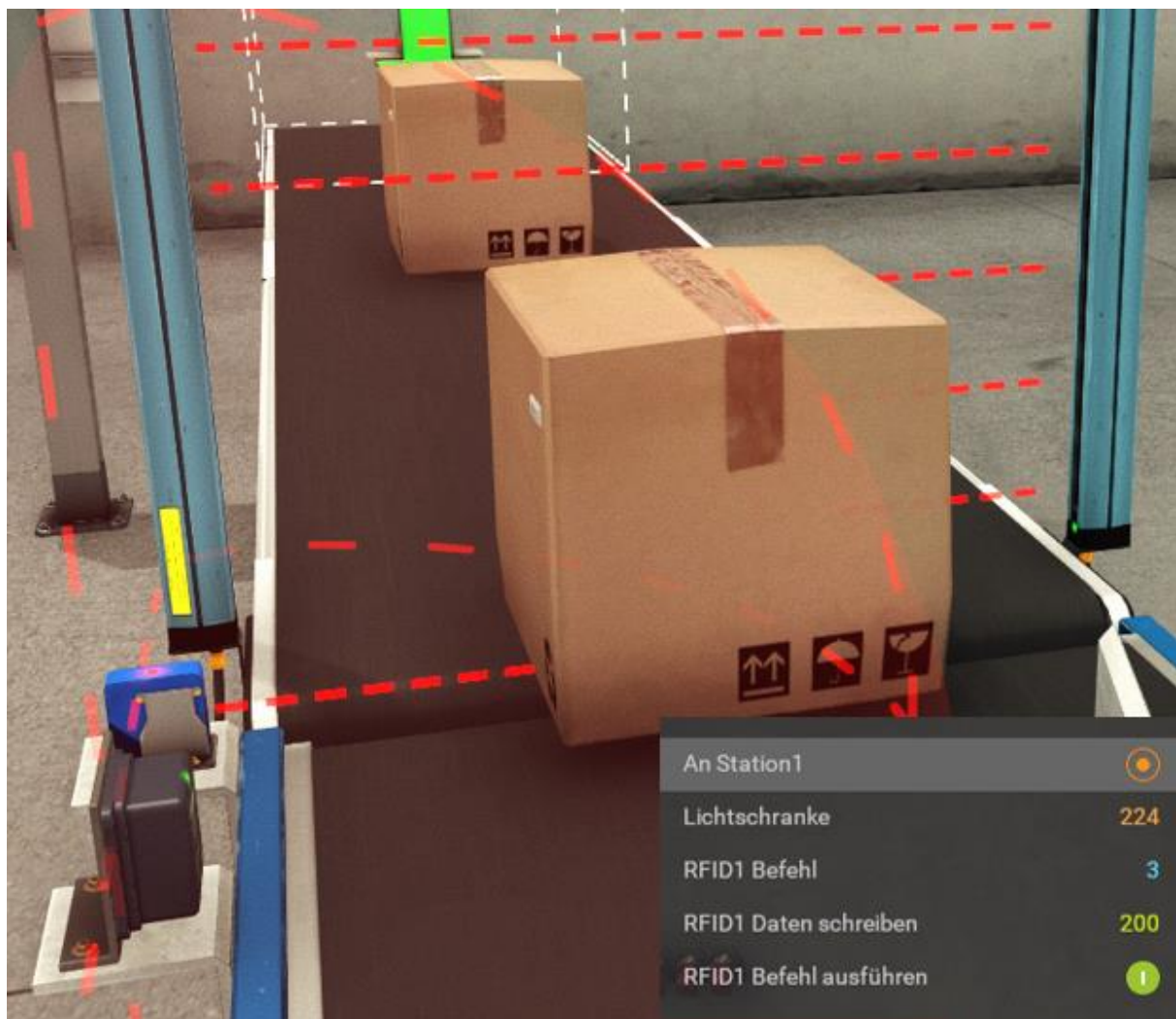


Abb. 4.3.1-2: RFID-Sensor 1

Wie in der Grafik dargestellt, wird der Karton der Größe "L" unmittelbar nach dem Durchqueren der Lichtschranke vom photoelektrischen Sensor (An Station1) erkannt. Dieser Impuls aktiviert den RFID-Sensor und löst den Schreibbefehl aus. Die Lichtschranke liefert dem RFID-Sensor

die Information über die Kartongröße, wodurch dieser das RFID-Etikett mit dem Wert 200 beschreibt (siehe Abbildung 4.3.1-2, rechts unten). Kartons der Größe "M" hingegen werden mit dem Wert 100 beschrieben.

Es ist wichtig anzumerken, dass der RFID-Sensor vor der Verwendung konfiguriert werden muss, um Daten schreiben zu können. Diese Konfiguration erfolgt über die Adresse "RFID1 Befehl" (siehe Abbildung 4.3.1-2, rechts unten), wobei die Zahl 3 für die Schreibfunktion steht.

Eine ebenso zentrale Bedeutung innerhalb des Verteilungsprozesses nimmt die Integration einer Waage (4) ein. Bei Beanspruchung generiert besagte Waage ein analoges Spannungssignal, welches in einem Wertebereich von 0 bis 10 Volt variiert. Dieser Wert wird anschließend auf der Softwareoberfläche dargestellt. Unter Berücksichtigung der von Factory IO bereitgestellten Angaben bezüglich der Gewichtsobergrenze, die in diesem spezifischen Projekt bei 20 Kilogramm liegt (vgl. [25]), gestaltet sich die Ermittlung des Verhältnisses zwischen der erzeugten Spannung und der Masse äußerst unkompliziert:

$$\mathbf{WB_U = [0,10] V ; WB_m = [0,20] kg ; m_o = 20kg}$$

$$10V = 20kg \quad |: 10$$

$$\underline{\underline{1V = 2kg}}$$

So ergeben sich für die Kartons folgende Gewichte:

	U in V	m in kg
Palletizing Box	1,50	3,00
Karton M	5,00	10,00
Karton L	7,50	15,00

Tabelle 3: Einheitenkonvertierung der Kartons in Factory IO

Basierend auf den Spannungswerten entscheidet anschließend ein Rollensortierer (5), welche Kartons zum Palettierer und welche zur Sortiereinheit 2 weitergeleitet werden. Konkret erfolgt die Zuweisung, indem Kartons der Kategorie „Palletizing Box“ zum Palettierer und die der Kategorie „M“ und „L“ an die Sortiereinheit 2 geleitet werden.

4.3.2 Palettierer

Der Palettierer (siehe Abb. 4.3.1-1: Palettierer) ist eine vorgefertigte Station in der Factory IO-Oberfläche. Seine Hauptaufgabe besteht darin, die kleinsten Kartons (1), die sogenannten „Palletizing Boxes“ (dt. Palettierkartons) zu verarbeiten. Diese Kartons werden in einer festgelegten Abfolge mithilfe eines Schiebers (2) auf eine Tür (3) geschoben und durch eine Klemme (4) zentriert. Sobald eine Ebene, bestehend aus 6 Kartons, komplett ist, öffnet sich die Tür, diese Ebene wird auf eine Palette abgesetzt und der Umlenker (5) öffnet sich. Er ist dafür verantwortlich, dass die Kartonebenen verschachtelt aufeinander liegen, um die Stabilität auf der Palette zu gewährleisten. Die Palette wird zuvor durch einen integrierten Aufzug (6) unterhalb der Tür platziert. Nachdem drei Ebenen auf die Palette gelegt wurden, ist sie vollständig verpackt. Der Aufzug fährt dann in die untere Position und die Palette wird über ein Rollenförderer-System zur weiteren Verarbeitung abtransportiert. Die obere Ebene des Palettierers ist über eine Treppe erreichbar und kann in der laufenden Simulation mit der „First-Person-Kamera“ begangen werden.

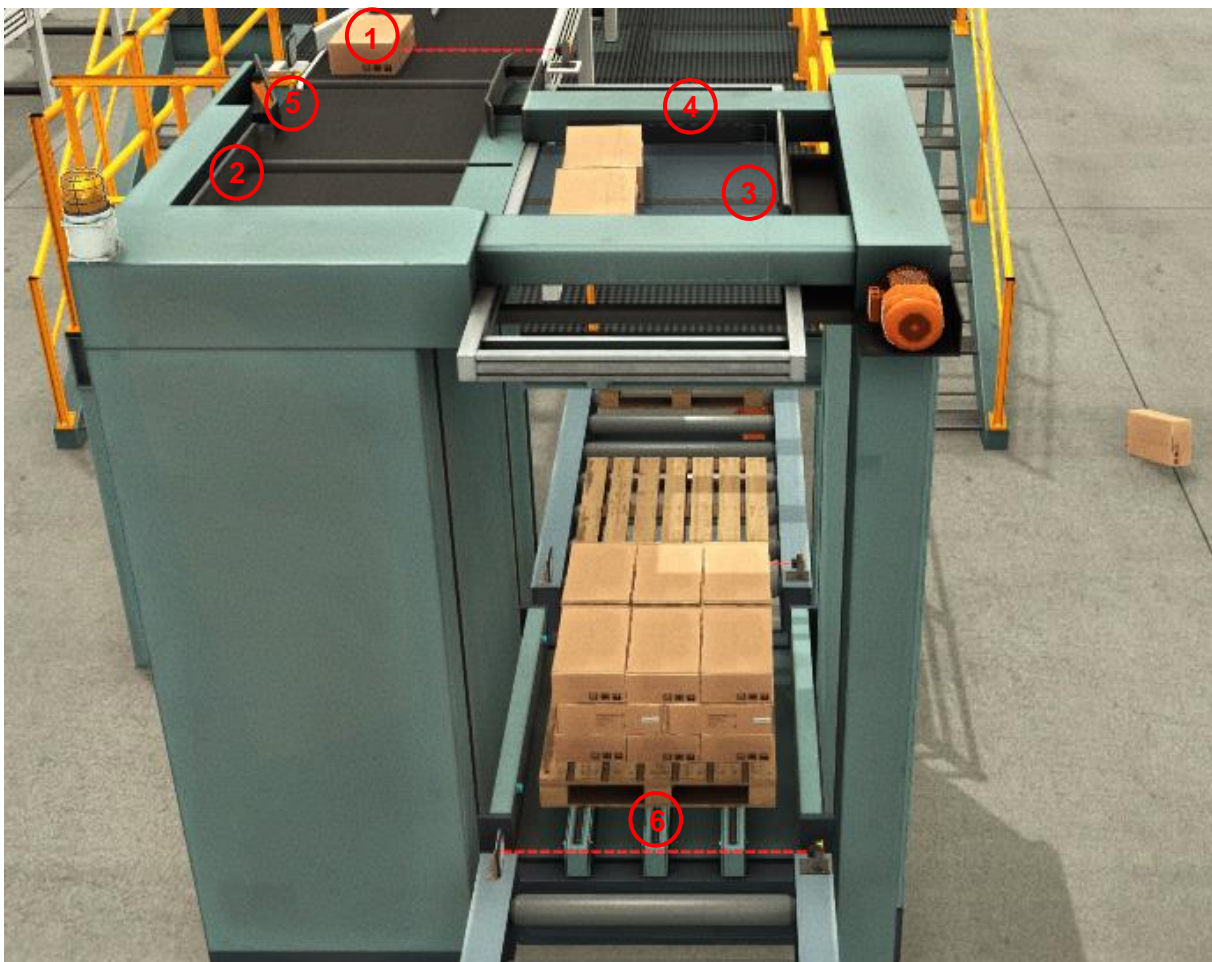


Abb. 4.3.2-1: Palettierer

4.3.3 Sortiereinheit 2

Die Sortiereinheit 2 verarbeitet die Kartongrößen "M" und "L", die zuvor von dem RFID-Sensor in Sortiereinheit 1 mit den festen Werten 100 bzw. 200 beschrieben wurden. Ein zweiter RFID-Sensor (1) liest diese Werte aus und übergibt sie an einen weiteren Rollensortierer (2). Jedoch ist auch hier zu beachten, dass, wie bereits in Abschnitt 4.3.1, "Sortiereinheit 1" erörtert, die Anweisung von einem unmittelbar benachbarten photoelektrischen Sensor (3) stammt. Sobald dieser photoelektrische Sensor auslöst, aktiviert er den RFID2-Sensor, welcher daraufhin den Lesebefehl ausführt. Dieser Vorgang wird in Abbildung 4.3.3-1: "Sortiereinheit 2" veranschaulicht. Der Rollensortierer verteilt schließlich die beiden Kartongrößen an eines der beiden Bänder, wo die weitere Verarbeitung stattfinden kann. Präzise gesagt, Kartongröße „L“ wird auf das erste Band (4) und die Kartongröße „M“ auf das zweite Band (5) befördert.

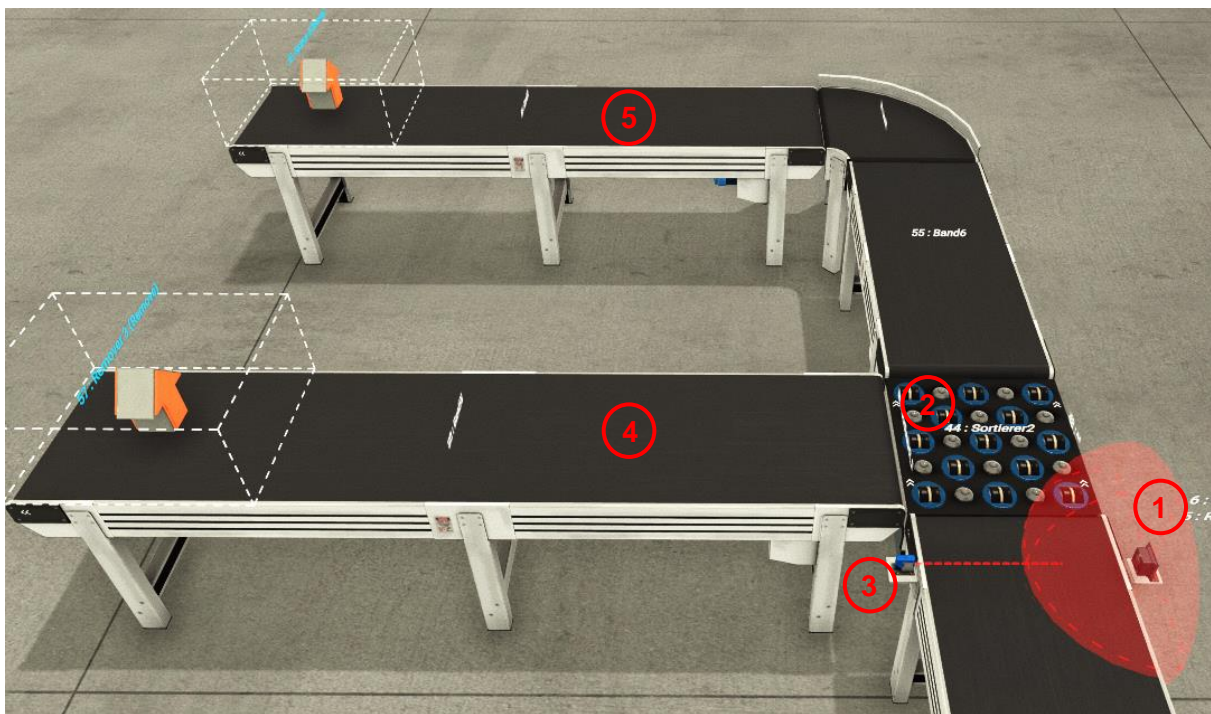


Abb. 4.3.3-1: Sortiereinheit 219

4.4 Inbetriebnahme der Anlage

In Vorbereitung auf die Inbetriebnahme der Anlage sind mehrere aufeinanderfolgende Schritte auszuführen. Der erste Schritt erfordert die Herstellung einer Verbindung zur virtuellen SPS mithilfe der Programmiersoftware Codesys. Nach erfolgreicher Übersetzung des Programmiercodes und Aktivierung der globalen Variablen wird die Simulation über Codesys gestartet. Hierdurch wird das zuvor geschriebene Programm übertragen und ermöglicht die Steuerung der Anlage. Die folgende Aufgabe besteht darin, den Treiber in Factory IO zu konfigurieren. Hierbei wird unter den Treibereinstellungen der OPC Client DA/UA ausgewählt. Anschließend ist der OPC-Server zu wählen, der in diesem Fall der PC ist, von dem aus das Codesys-

Programm ausgeführt wird. Es ist nun erforderlich, die globale Variablenliste zu selektieren, um sicherzustellen, dass die gewünschten Ein- und Ausgänge im Treiber verfügbar sind. Schließlich müssen die Sensoren und Aktoren von Factory IO mit den Ein- und Ausgängen des Treiberbausteins verbunden werden. Dieser Prozess wird in Anlage 1 „Ausschnitt Treiberbaustein Factory IO“ dargestellt.

4.4.1 Betriebsart Handbetrieb

Gemäß den Vorgaben des Projekts ist ein Handbetrieb im eigentlichen Sinne nicht vorgesehen und wurde nicht implementiert. Factory IO bietet jedoch die Möglichkeit, die Werte von Sensoren und Aktoren zu erzwingen. Dies ermöglicht die manuelle Steuerung bestimmter Szenarien, indem die vom Controller gelesenen Werte überschrieben werden.

Im Verlauf der Anlagenerstellung innerhalb der Factory IO-Oberfläche wurde bereits ein Drehschalter am Bedienpult mit den Schaltstellungen "Automatik" und "Manuell" installiert. Zusätzlich wurden beide Betriebsmodi in das SPS-Programm und somit im Treiberbaustein integriert. Aufgrund dieser bestehenden Struktur wäre eine spätere Implementierung eines Handbetriebs unkompliziert durchzuführen.

4.4.2 Betriebsart Automatik

Die Initiierung des Automatikbetriebs der Anlage erfordert eine Reihe von Schritten. Zunächst wird die Simulation durch Betätigen des "RUN"-Knopfes gestartet. Anschließend kann der Drehschalter am Bedienpult (siehe Anlage 2 Bedienpult in den Anlagen) auf die Position "Automatik" gebracht werden, gefolgt von der Aktivierung des "Start"-Knopfes. Dadurch wird die Anlage in den Vollautomatikmodus versetzt. Um sicherzustellen, dass das Programm in einem neutralen Zustand ohne vorherige Speicherstände beginnt, wird empfohlen, vor jedem Start den "Reset"-Knopf einmal zu betätigen.

Die Förderstrecke kann durch Betätigen des "Stop"-Knopfes angehalten werden. Um die Anlage nach dieser Unterbrechung erneut zu starten, genügt es, den "Start"-Knopf zu betätigen. Jedoch gestaltet sich die Wiederaufnahme des Betriebs anders, wenn der "Not Halt"-Schalter betätigt wird. In diesem Fall ist ein umfassender Neustart erforderlich, der mit der Betätigung des "Reset"-Knopfes beginnt und anschließend mit den Schritten vom Anfang dieses Abschnittes fortgesetzt wird.

4.5 Codesys V3.5: das Programm

Codesys (offiziell CODESYS, zuvor CoDeSys) ist eine integrierte Entwicklungsumgebung für Speicherprogrammierbare Steuerungen (SPS) gemäß dem IEC 61131-3-Standard zur Anwendungsprogrammierung in der Industrieautomatisierung. Codesys wird von der Codesys Group mit Sitz in Kempten entwickelt und vermarktet. Das Softwareunternehmen wurde 1994 gegründet und in den Jahren 2018 und 2020 in Codesys Group bzw. Codesys GmbH umbenannt.

Der Name Codesys ist ein Akronym für "Controller Development System." Die Version 1.0 wurde erstmals im Jahr 1994 veröffentlicht. Dieses Programmiersystem kann frei lizenziert und ohne Kopierschutz legal auf verschiedenen Arbeitsplätzen installiert werden. [47]

Codesys bietet Unterstützung für alle fünf von der IEC 61131-3 (Internationale Elektrotechnische Kommission) spezifizierten Programmiersprachen:

- **IL (Instruction List)** oder zu Deutsch Anweisungsliste (AWL), ist eine textbasierte Sprache.
- **ST (Structured Text)** oder zu Deutsch Strukturierter Text (ST), ist eine textbasierte Sprache für strukturierte Programmierung.
- **LD (Ladder Diagram)** oder zu Deutsch Kontaktplan (KOP), bezeichnet die grafische Darstellung von Relais-Verdrahtungen.
- **FBD (Function Block Diagram)** oder zu Deutsch Funktionsplan-Diagramm (FUP), ist eine grafische Sprache mit vorgefertigten Bausteinen. Diese wurde in dem vorliegenden Projekt benutzt.
- **SFC (Sequential Function Chart)** oder zu Deutsch Ablaufsprache (AS), bezeichnet eine grafische Sprache für logische Abläufe und Prozesse. [48]

4.5.1 Beschreibung

Das erstellte Programm ist wie in Abschnitt 4.3 in die drei Teilanlagen Sortiereinheit 1, Palettierer und Sortiereinheit 2 gegliedert. Außerdem werden in dem Abschnitt 4.5.2 die Funktionen „START“, „Stop“, „Reset“ und „Not Halt“ näher beleuchtet. Im Rahmen der Implementierung wurde zudem nahezu ausschließlich auf die Verwendung globaler Variablen zurückgegriffen, da diese für den Austausch von Daten zwischen Factory IO und Codesys nötig sind. Sie bilden die Schnittstelle für die Ein- und Ausgangssignale auf dem Treiberbaustein in Factory IO und sind anhand der Bezeichnung „FIO.xxx“ von lokalen Variablen zu unterscheiden (siehe Anlage 1 Ausschnitt Treiberbaustein Factory IO).

4.5.2 Betriebskontrolle

Im Rahmen dieses Projekts fungiert der Startbaustein als der erste elementare Funktionsbaustein. Seine Hauptaufgabe besteht darin, die lokale Programmvariable "START" mithilfe der globalen Variablen "FIO.iStart", "FIO.iAuto", "FIO.iStop", "FIO.iNotHalt" und "FIO.iManuell" zu initiieren. Die Einführung der "START"-Variable dient dazu, den Programmieraufwand zu reduzieren und gleichzeitig die Übersichtlichkeit des Programms zu wahren. Dieser Ansatz ist besonders von Bedeutung, da die Startbedingung für sämtliche Anlagenteile identisch ist.

Die in der beigefügten Abbildung 4.5.2-1: „Startbaustein“ dargestellten Bedingungen verdeutlichen, welche Kriterien erfüllt sein müssen, um den „SET“-Eingang des RS-Flipflop (RS_0) zu setzen, wodurch "START" den logischen Wert „1“ annimmt.

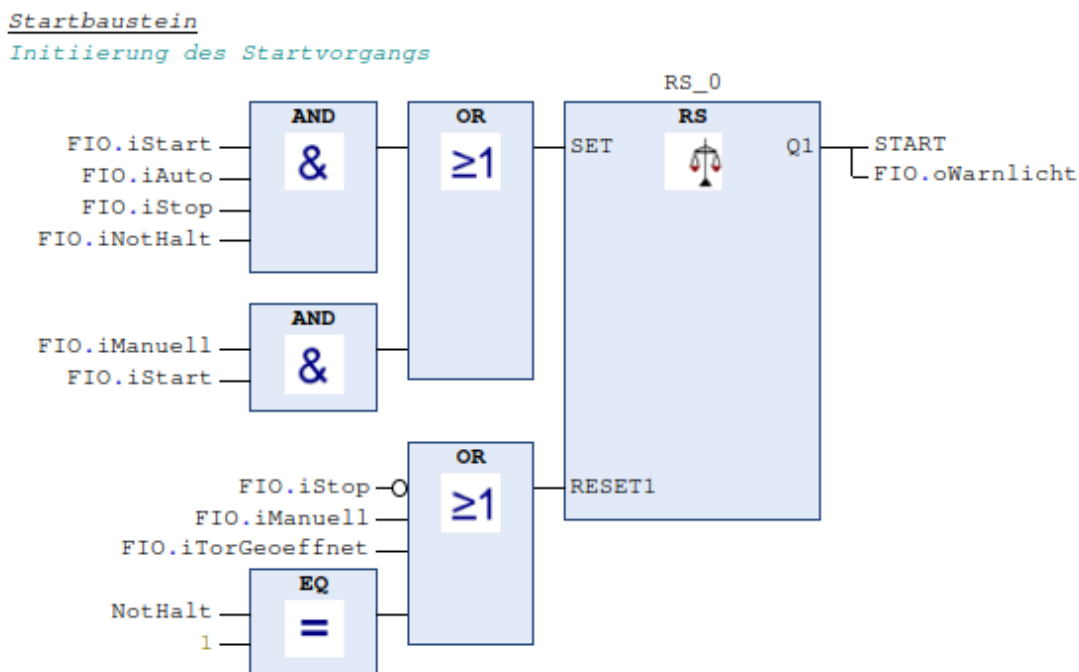


Abb. 4.5.2-1: Startbaustein

Ein RS-Flipflop ist durch einen dominanten "Reset"-Eingang gekennzeichnet. Wenn dieser aktiviert ist ($R = „1“$), wird das RS-Flipflop unabhängig von seinem Set-Eingangszustand unverzüglich zurückgesetzt. Dies bedeutet, dass der Ausgangszustand (Q) und somit die Variable „START“ auf „0“ gesetzt wird. Diese Festlegung hat eine herausragende Bedeutung in Bezug auf Sicherheit, insbesondere im Falle eines Notfalls, da das Abschalten der Anlage oberste Priorität hat.

In diesem vorliegenden Projekt wird die hier beschriebene Funktionalität an verschiedenen Stellen gezielt eingesetzt. Einerseits betrifft dies die Umsetzung der Stoppssequenz. Wenn der Wert der Variablen "FIO.iStop" auf den Zustand "0" gesetzt wird, erfolgt ein Rücksetzen des

Flipflops. Jedoch kann dessen Ausgang nach erneuter Aktivierung der Variable "FIO.iStart" ebenfalls wieder in den Zustand "1" versetzt werden.

Eine unterschiedliche Herangehensweise gilt für die Sicherheitsfunktion, die von der Variable "FIO.iNotHalt" erfüllt werden muss. Ein Not-Halt hat den Zweck, die Bewegung oder Aktivität einer Anlage bei drohender Gefahr unverzüglich zu stoppen. Dies impliziert, dass nach Betätigen des Not-Halt-Tasters keine anderen Funktionen der Anlage verfügbar sein dürfen. Aus diesem Grund ist es nach dem Auslösen der Variable "FIO.iNotHalt", die einem logischen Zustand von "0" entspricht, nicht ohne weiteres möglich, die Startsequenz erneut auszuführen, selbst wenn der Not-Halt-Taster in der Anlage in seine Ausgangsposition zurückgeführt wird.

Um dieses Verhalten zu realisieren, wurde ein Vorwärtszähler (siehe Abbildung 4.5.2-2: Zähler Not-Halt) in das Programm integriert. Dieser Zähler erhöht die Zählvariable "NotHalt" um eins, sobald "FIO.iNotHalt" auf den Wert "0" gesetzt wird. Diese Zählvariable wurde in Verbindung mit einem Gleichheitsvergleicher in den Startbaustein integriert (siehe Abb. 4.5.2-1: Startbaustein). Dadurch muss der Zähler erst mittels der globalen Variable "FIO.iReset" zurückgesetzt werden, bevor der Startvorgang erneut ausgeführt werden kann.

Zähler Not-Halt

wird Not-Halt betätigt, zählt Zähler 1 hoch und sperrt somit den START Baustein.
Erst nach Betätigen des Reset-Knopfes, wird der Zähler zurückgesetzt und START Baustein wird freigegeben.

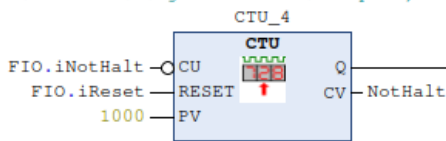


Abb. 4.5.2-2: Zähler Not-Halt

4.5.3 Sortiereinheit 1

In diesem Programmabschnitt wird die Grundlage für die Kartonverteilung in der Simulation gelegt, wie im Abschnitt 4.3.1 beschrieben. Durch die Integration des Startbausteins werden sämtliche relevanten Aktoren aktiviert (siehe Anlage 2, Netzwerk 26 in Anlagen).

Mittels eines Verschiebeoperators werden die ganzzahligen Werte der lokalen Variablen "KartonM" (100) und "KartonL" (200) an die globale Variable "FIO.oRFID1Schreiben" übertragen, sobald die Variable "FIO.iLichtschranke" einen bestimmten Wert annimmt, wie in der Anlage 2, Netzwerk 28 und 29 detailliert dargestellt. Im Netzwerk 30, "Bedingung Wert auf Karton Tag schreiben" (siehe Anlage), erhält "FIO.oRFID1Schreiben" in der Simulation die Anweisung, die zuvor empfangenen Werte auf einen RFID-Tag zu schreiben, wie im Abschnitt 4.3.1 beschrieben.

Daraufhin erfolgt ein Vergleich mit einem im Programm festgelegten Wert und der globalen Variable "FIO.iGewicht". Diese globale Variable überträgt ihrerseits Werte aus der Simulation

in das Programm. Anhand dieser Werte wird entschieden, ob der logische Zustand der globalen Variable "FIO.oSortierer1_R" auf "1" oder "0" gesetzt wird. Der entsprechende Baustein ist den beigefügten Anlagen zu entnehmen (Anlage 2, Netzwerk 27).

4.5.4 Palettierer

Im vorliegenden Programm stellt der Palettierprozess das zentrale Element dar. Dieser Prozess weist eine umfangreiche Programmierung auf und besteht aus insgesamt 18 Netzwerken, was ihn zu einem der aufwendigsten Teilsysteme innerhalb der Gesamtanlage macht. Charakteristisch sind die zahlreich verwendeten Flipflops, die mit Hilfe von Einschaltverzögerungen ein zuverlässiges Ein- und Ausschalten begünstigen. Wie bereits in vorherigen Abschnitten erläutert, erfolgt auch hier die Integration des Startbausteins, um sämtliche relevanten Aktoren in Betrieb zu setzen.

Kartonzähler

zählt die Kartons, die in den Palettierer hinein fahren. Resetbedingung des Zählers ist die Zählvariable des Zählers der Tür.

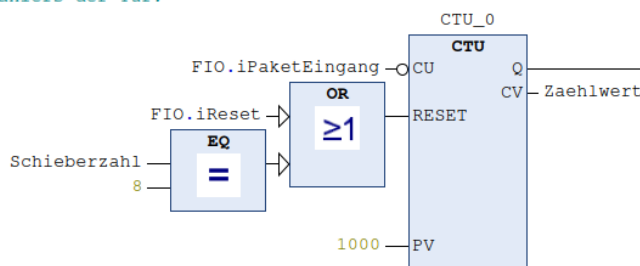


Abb. 4.5.4-1: Kartonzähler

Der Schlüsselbaustein des Palettierprozesses ist der Aufwärtszähler "Kartonzähler," wie in Abbildung 4.5.4-1: „Kartonzähler“ dargestellt. Dieser Baustein trägt die Verantwortung dafür, dass der Schieber in der Simulation eine vordefinierte Anzahl von Kartons auf die Tür schiebt, wie im Abschnitt 4.3.2 beschrieben. Darüber hinaus gewährleistet der Kartonzähler, dass die Aktoren, die durch das Signal der Variable "START" aktiviert werden, kurzzeitig gestoppt werden, um einen Stau von Kartons in der Simulation zu verhindern (vgl. Anlage 4, Netzwerk 8). Zudem gewährleistet der Kartonzähler die zeitlich abgestimmte Öffnung und Schließung der Klemme (FIO.oKlemme) und der Tür (FIO.oTuer), welche für die Simulation von großer Bedeutung sind. Schließlich sorgt er auch dafür, dass der Umlenker (FIO.oUmlenker) die Positionen der Kartons gemäß des geplanten Ablaufs ändert (siehe Anlagen 4 bis 8 in den beigefügten Dokumenten).

Der "Kartonzähler" wird durch einen weiteren Aufwärtszähler, den "Schieberzähler," zurückgesetzt. Dieser "Schieberzähler" ist im Netzwerk 9 der Anlage 5 zu finden. Das gesamte Funktionsplan-Diagramm sowie die globale Variablenliste ist den Anlagen beigefügt.

4.5.5 Sortiereinheit 2

Im letzten Abschnitt des vorliegenden Programms erfolgt ein Vergleich zweier ganzzahliger Werte, ähnlich wie im Abschnitt 4.5.3. Konkret handelt es sich um den Vergleich der Werte, die in der Sortiereinheit 1 vergeben wurden und in der Simulation erfasst werden. Diese Werte werden über die Verwendung der globalen Variable "FIO.iRFID2Lesen" in das Codesys-Programm übertragen. Der Impuls, der zum Auslesen dieser Werte führt, wird über den Einsatz eines Bausteins im Netzwerk 32 gesteuert, wie in Anlage 11 detailliert beschrieben.

Bedingung Sortierer2

*Liest der Sensor den Wert 200, fährt der entsprechende Karton (L) nach links.
Liest der Sensor den Wert 100, fährt der entsprechende Karton (M) geradeaus weiter.*

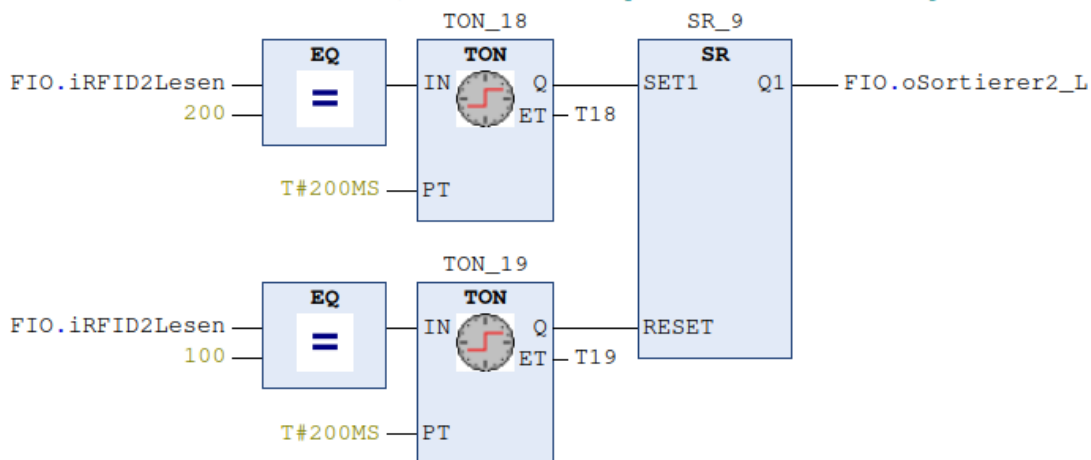


Abb. 4.5.5-1: Bedingung Sortierer 2

Die Abbildung 4.5.5-1: "Bedingung Sortierer 2," zeigt, wie die Steuerung der globalen Ausgangsvariable "FIO.oSortierer2_L" durch den zuvor erwähnten Vergleich realisiert wird. Wenn die Variable "FIO.iRFID2Lesen" den Wert 200 erhält, wird das SR-Flipflop (SR_9) gesetzt und "FIO.oSortierer2_L" wird auf den logischen Zustand "1" gesetzt. Dies führt dazu, dass der Sortierer 2 in der Simulation einen Karton der Größe "L" nach links umleitet, wie im Abschnitt 4.3.3 beschrieben.

Eine signifikante Eigenschaft dieses Mechanismus besteht darin, dass die Variable "FIO.iRFID2Lesen" den aktuellen Wert beibehält, bis ein anderer ganzzahliger Wert aus der Simulation übertragen wird. Dies gewährleistet eine reibungslose Umsetzung des Mechanismus, da das Netzwerk einem einfachen Ein- und Ausschaltvorgang entspricht.

4.5.6 Optimierungsmöglichkeiten

Im Verlauf der Entwicklung des Programmcodes und während der anschließenden Testphase wurden Funktionen implementiert, die zwar nicht unmittelbar mit der aktuellen Aufgabenstellung in Verbindung stehen, dennoch erzeugt wurden, sodass ihre zukünftige Integration in die Simulation problemlos durchgeführt werden kann. Als exemplarisches Beispiel sei der Betriebszustand "Handbetrieb" genannt, der innerhalb des Programms mittels einer globalen Variable so ausgearbeitet wurde, dass lediglich die notwendigen Bedingungen wie die Steuerung von Aktoren bei Bedarf in das Programm aufgenommen werden müssen. In der Simulation kann der Betriebszustand mithilfe eines Drehschalters zwischen "Automatik" und "Manuell" umgeschaltet werden. Eine detaillierte Darstellung der Implementierung der "Handbetrieb"-Funktion ist in Anlage 12 zu finden.

Des Weiteren besteht die Möglichkeit, die in der Simulation kontinuierlich leuchtenden Lichter der Druckschalter auf dem Bedienpult über ein separates Baustein mit einem Taktgeber zu versehen, sodass sie in bestimmten Phasen der Simulation blinken. Die zugehörigen globalen Variablen sind in der globalen Variablenliste in Anlage 13 aufgeführt. Zuletzt ist die globale Variable "FIO.oZähler" zu erwähnen. Durch ihre Verwendung kann beispielsweise eine digitale Anzeige, die in der Oberfläche von Factory IO verfügbar ist, als Karton- oder Palettenzähler programmiert werden.

5 Die Sortieranlage

5.1 Einführung

Dieses Kapitel thematisiert die Implementierung, Beschreibung und das Testen eines Simulationsmodells, in einem Projektbuch aufgearbeitet und von der Westermann Gruppe bereitgestellt. [15] Dieses Lehrmaterial wurde entwickelt, um angehende Auszubildende praxisnah auf die erweiterte SPS-Programmierung vorzubereiten. Es basiert auf einem Sortieranlagenmodell, das eng an die Anforderungen der IHK-PAL-Prüfung für Elektroniker/in für Automatisierungstechnik angelehnt ist (siehe Abb. 5.1-1: Sortieranlage IHK-PAL-Prüfung). Die Umsetzung dieses Lehrmaterials erfordert die Nutzung der Software Siemens SIMATIC Manager, die derzeit kostenlos zur Verfügung steht. Zusätzlich ist eine Voll- oder Demoversion des Siemens TIA Portal V15.1 erforderlich, um die notwendige Funktionalität sicherzustellen. Das Sortieranlagenmodell kann im Demo-Modus von SIMATIC dearchiviert und geöffnet werden. Die Anlagensimulation läuft im Demo-Modus für 45 Minuten und kann nach Ablauf dieser Zeitspanne erneut gestartet werden. Des Weiteren sind das SPS-Programm sowie viele andere wesentliche Dateien über die Internetseite des Herausgebers des Projektbuches, Stefan Manemann, zum Download verfügbar (vgl. [49]).

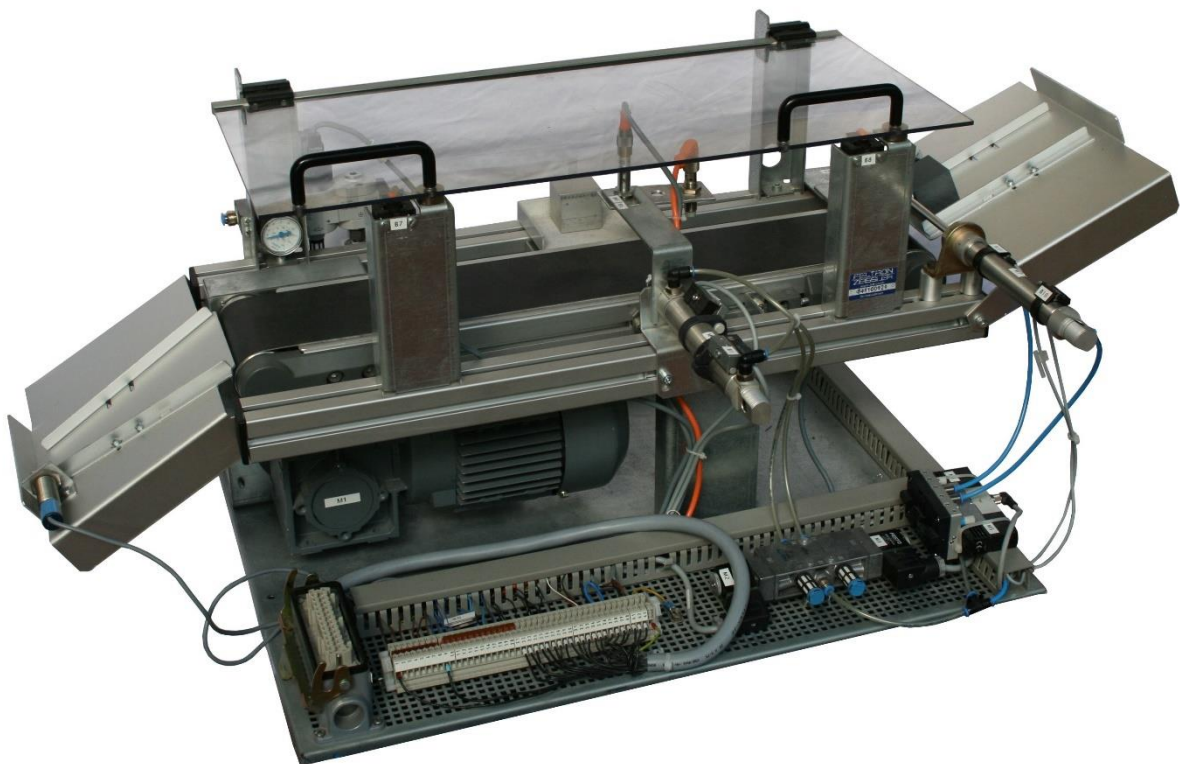


Abb. 5.1-1: Sortieranlage IHK-PAL-Prüfung [49]

Durch die Verbindung von SIMIT mit der CPU 1512C-1 PN durch PLCSIM Advanced wird ermöglicht, Programme zu testen, als würde eine reale Anlage verwendet werden, ohne die Notwendigkeit einer physischen SPS oder eines tatsächlichen Bandmodells. Die CPUs der S7-1500er-Serie können reibungslos an SIMIT SP mit PLCSIM Advanced gekoppelt werden können. Die Inbetriebnahme von PLCSIM Advanced ist der erste Schritt, um sicherzustellen, dass SIMIT SP auf die virtuelle CPU zugreifen kann. Vor dem Starten der SIMIT-Simulation sollte S7-PLCSIM Advanced bereits aktiviert sein, da andernfalls die Anlagensimulation in SIMIT SP die Schnittstelle zur SPS nicht finden kann.

Anschließend wird das Sortieranlagenmodell in der SIMIT Projektansicht dearchiviert. Die grafische Darstellung der Sortieranlage wird durch Doppelklicken auf "Bedienbild" geöffnet. Die Simulation des Anlagenmodells wird durch einen Klick auf "Starten" initiiert. Im nächsten Schritt sollte das SPS-Programm geöffnet und die Simulierbarkeit in den Projekteigenschaften aktiviert werden. Sobald die virtuelle CPU in PLCSIM Advanced in Betrieb ist, kann das SPS-Programm in die virtuelle CPU geladen und die Anlage mit dem Programm getestet werden. Die Simulation der Sortieranlage weist eine hohe Ähnlichkeit zum realen Betrieb auf.

Es besteht die Möglichkeit, das Förderband mithilfe von zwei rechteckigen, farbigen Buttons zu bestücken, wobei Silber für Metall und Grün für Kunststoff steht. Die Werkstücke werden entsprechend ihrer Materialeigenschaften von den verschiedenen Sensoren erkannt. Der induktive Sensor „B5“ erkennt beispielsweise nur metallische Werkstücke, während der Sensor „B4“ alle Werkstücke erkennt (siehe Abb. 5.2-1: „SIMIT Sortieranlage“). Das Bedienfeld mit seinen Leuchttastern und -meldern ist ebenfalls in der Simulation abgebildet (vgl. [15] S. 108ff).

5.2 Aufbau der Anlage

Grundlegend besteht die Sortieranlage, wie Abbildung 5.2-1: „SIMIT Sortieranlage“ zeigt, aus drei Haupteinheiten, welche über ein Förderband miteinander verbunden sind.

Die erste Einheit (1) umfasst eine Rutsche, auf der die Werkstücke zur Sortierung positioniert werden. Hierbei passiert jedes Werkstück den optischen Sensor "B3", welcher die Laufrichtung "Q1" des Motors "M1" des Förderbandes aktiviert (siehe Einheit drei). Des Weiteren beinhaltet diese Einheit den Zylinder "M7" mit den Endlagen "B10" und "B11", der zur pneumatischen Vereinzelung rückwärtsfahrender Werkstücke verwendet wird. Die Betätigung des Zylinders "M7" erfolgt über ein pneumatisch gesteuertes 5/2-Wegeventil mit den Schaltstellungen "M5" und "M6".

Die zweite Einheit (2) stellt die "Sortiereinheit 1" dar. Sie enthält den induktiven Sensor "B5", den Zylinder "M4" mit den Endlagen "B1" und "B2", sowie den optischen Sensor "B4" und das "Magazin Metall". Der Zylinder "M4" wird über ein pneumatisch gesteuertes 5/2-Wegeventil

mit den Schaltstellungen "M2" und "M3" betätigt. Wenn ein Werkstück aus Metall den induktiven Sensor "B5" aktiviert, bewegt sich das Förderband bis zum Sensor "B4" und stoppt dann. Anschließend fährt der Zylinder "M4" aus und befördert das Werkstück in das Magazin. Nach Erreichen der Endlage "B2" kehrt der Zylinder wieder in seine Ausgangsposition zurück, und eine erneute Bestückung des Förderbandes kann stattfinden.

Die dritte Einheit (3) besteht zum einen aus dem Motor "M1" mit den Laufrichtungen "Q1" für den Vorwärtslauf und "Q2" für den Rückwärtslauf. Zum anderen beinhaltet sie das "Magazin Kunststoff" und den optischen Sensor "B9". Die Voraussetzung dafür, dass ein Werkstück das Kunststoffmagazin erreicht, ist, dass es zuvor nicht den induktiven Sensor "B5" (siehe Einheit zwei) aktiviert hat. Ist diese Bedingung erfüllt, erreicht das Kunststoffwerkstück das Magazin, und der Sensor "B9" löst aus, wodurch der Motor "M1" gestoppt wird. Durch erneutes Auslösen des Sensors "B3" in Einheit eins wird der Motor wieder gestartet.

In der Abbildung 5.2-1: „SIMIT Sortieranlage“ wird zusätzlich die Kontrolle der Druckluftzufuhr (4) für die Sortieranlage dargestellt. Dies erfolgt mithilfe von zwei 3/2-Wegeventilen und dem induktiven Sensor "B6". Des Weiteren ist ein grauer Schalter mit den Kontakten "B7" und "B8" (5) zu sehen. Dieser Schalter repräsentiert beispielhaft die Schutztür der realen Anlage, wie sie in Abbildung 5.1-1 mit dem Titel "Sortieranlage IHK-PAL-Prüfung" zu erkennen ist.

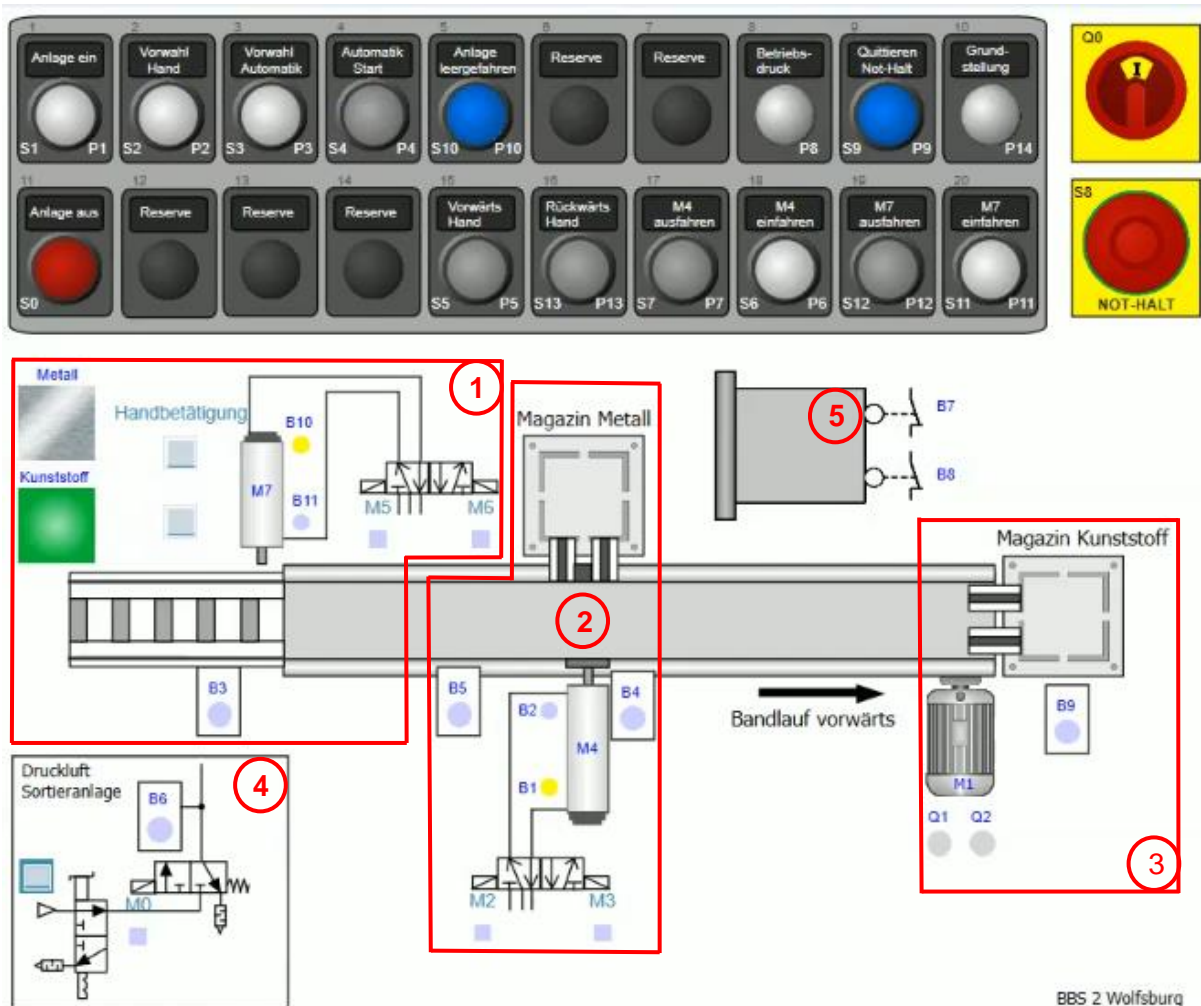


Abb. 5.2-1: SIMIT Sortieranlage [15]

5.3 Inbetriebnahme der Anlage

Einleitend sei gesagt, dass alle im Folgenden genannten Kennzeichen der Abbildung 5.2-1: „SIMIT Sortieranlage“ im Abschnitt 5.2 entnommen werden können.

Die Initiierung der Anlage erfordert zunächst die Aktivierung des Hauptschalters (Q0). Im Anschluss erfolgt die Einschaltung der Steuerspannung über den Taster "Anlage ein" (S1), wobei dieser Zustand durch die Beleuchtung der Leuchte (P1) kenntlich gemacht wird. Sobald die Steuerspannung freigegeben ist, besteht die Möglichkeit, das Not-Halt-Relais durch Betätigen des Tasters "Quittieren" (S9) zu bestätigen. Nach der Quittierung erlischt die Anzeige an der Quittierungsleuchte (P9) und die Druckluft wird aktiviert. Im Falle der Aktivierung des Not-Halt-Tasters (S8) an der Anlage oder der Öffnung des Schutzgitters (B7/B8) erfordert dies eine erneute Quittierung des Not-Halt-Relais, gefolgt von einem ordnungsgemäßen Herunterfahren der Anlage. Das Unterlassen der Quittierung eines ausgelösten Not-Halts wird durch die leuchtende Anzeige "Quittierung" (P9) angezeigt.

In diesem Stadium befindet sich die Anlage in einem Betriebsmodus, in dem zwischen den Betriebsarten "Handbetrieb" und "Automatikbetrieb" gewählt werden kann. Es sei jedoch darauf hingewiesen, dass weder eine Betriebsart auswählbar ist, wenn die Druckluft deaktiviert ist (B6= „0“), noch im Falle eines ausgelösten Motorschutzschalters (F1= „0“). [15]

5.3.1 Betriebsart Handbetrieb

Die Auswahl der Betriebsart "Handbetrieb" erfolgt mittels Aktivierung des Tasters "Vorwahl Hand" (S2). Die Auswahl des Handbetriebs wird durch die Beleuchtung des Tasters "Vorwahl Hand" (P2) signalisiert. Innerhalb des Handbetriebsmodus besteht die Möglichkeit, den Antrieb M1 schrittweise vorwärtszubewegen, vorausgesetzt, der Zylinder M4 befindet sich in der eingefahrenen Position. Die Vorwärtsbewegung des Antriebs wird durch das Leuchten des Melders "Vorwärts Hand" (P5) visualisiert.

Ebenfalls ermöglicht der Handbetrieb das Aus- und Einfahren des Zylinders M4 durch Betätigung der Taster "M4 ausfahren" (S7) und "M4 einfahren" (S6). Die Endposition des Zylinders M4 wird durch "M4 ausfahren" (P7) und "M4 einfahren" (P6) signalisiert. Um den Übergang zur Betriebsart "Automatikbetrieb" zu initiieren, ist es erforderlich, dass die Anlage im Handbetrieb geleert wird. Das bedeutet, dass sich keine Werkstücke auf dem Förderband befinden dürfen. Zusätzlich muss der Zylinder M4 eingefahren sein, das Förderband abgeschaltet sein und das Kunststoffmagazin bereitstehen.

Nachdem das Förderband leer ist, muss dies durch die Betätigung des Tasters "Anlage leergefahren" (S10) quittiert werden, wodurch der Melder "Anlage leergefahren" (P10) aufleuchtet. Falls der Zustand "Anlage leergefahren" nicht bestätigt wird, wird der Melder "Anlage leergefahren" (P10) mit einer Frequenz von 1 Hz blinken. Ohne die Bestätigung des Zustands "Anlage leergefahren" ist die Auswahl der Betriebsart "Automatikbetrieb" nicht möglich. Sofern sich das Förderband in der Grundstellung befindet, wird der Melder "Grundstellung" (P14) dauerhaft leuchten, andernfalls wird er blinken. [15]

5.3.2 Betriebsart Automatik

Im Falle der Aktivierung des Zustands "Anlage leergefahren" (Anzeige P10 leuchtet), steht die Option zur Verfügung, die Betriebsart "Automatikbetrieb" durch Betätigung des Tasters "Vorwahl Automatik" (S3) zu aktivieren. Nach Auswahl des Automatikbetriebs leuchtet die Anzeige "Vorwahl Automatik" (P3). Die Initialisierung des Automatikbetriebs erfolgt durch Aktivierung des Tasters "Automatik Start" (S4), wodurch die Leuchte "Automatik Start" (P4) zu leuchten beginnt. Zu diesem Zeitpunkt ist die Anlage bereit, Werkstücke aus Metall oder Kunststoff zur automatischen Sortierung auf das Förderband entgegenzunehmen. Wenn ein Werkstück von Sensor B3 auf der Rutsche erfasst wird, wird das Förderband in Vorwärtsrichtung gestartet.

Während des Sortiervorgangs blinken die Anzeigen "Anlage leergefahren" (P10) und "Grundstellung" (P14) mit einer Frequenz von 1 Hz. Kunststoffwerkstücke werden in das Kunststoffmagazin einsortiert. Wenn ein Kunststoffwerkstück im Kunststoffmagazin erkannt wird, signalisiert Sensor B9 dies und der Antrieb des Förderbands wird deaktiviert. Im Falle, dass ein Metallwerkstück auf das Förderband gelegt wird, identifiziert der induktive Sensor B5 das metallische Werkstück. Sobald das metallische Werkstück das Metallmagazin erreicht, wird Sensor B4 ausgelöst, was zur Abschaltung des Förderbands führt. Anschließend schiebt der Zylinder M4 das Werkstück in das Metallmagazin. Nachdem das Werkstück in das Metallmagazin verschoben wurde, kann das nächste Werkstück sortiert werden. [15]

6 Gegenüberstellung

Eine bemerkenswerte Stärke von SIMIT liegt in seiner umfassenden Prozesssimulation. Die Software vermag komplexe Abläufe in der Prozessindustrie äußerst präzise zu reproduzieren, was für Unternehmen, die realistische Tests und Validierungen von Automatisierungssystemen durchführen, von großer Bedeutung ist. Zudem zeichnet sich SIMIT durch seine Integrationstiefe aus, da es eng mit Siemens Steuerungssystemen und anderen Automatisierungslösungen verknüpft ist, was eine nahtlose Integration in bestehende Automatisierungssysteme ermöglicht und die Test- und Validierungsprozesse erheblich vereinfacht. Des Weiteren erlaubt SIMIT die Anpassung an die spezifischen Anforderungen eines Projekts durch die Erstellung komplexer Modelle und Simulationen, was seine Flexibilität und Anpassbarkeit an verschiedene Branchen und Anwendungen unterstreicht.

Auf der anderen Seite stellt die Kostenstruktur potenziell einen Nachteil von SIMIT dar, insbesondere wenn umfassende Funktionalitäten und Lizenzierungen erforderlich sind, was sie für kleinere Unternehmen oder Schulungszwecke weniger wirtschaftlich macht. Zusätzlich dazu ist SIMIT aufgrund seiner Tiefe und Komplexität mit einer steilen Lernkurve verbunden, wodurch neue Benutzer möglicherweise zusätzliche Schulung benötigen, um die Software effektiv zu nutzen. Eine detaillierte Liste aller Lizenzkosten gibt die Siemens AG heraus, wenn ein Unternehmenskonto existiert, was im Laufe der Erstellung dieser Arbeit nicht realisiert werden konnte.

Factory IO zeichnet sich insbesondere durch seine Benutzerfreundlichkeit aus, da es eine intuitive Benutzeroberfläche bietet, die leicht erlernbar ist. Dies macht es ideal für Schulungszwecke und die Ausbildung von Bedienern. Eine weitere Stärke von Factory IO liegt in seiner visuellen Darstellung, da die Software eine realistische 3D-Visualisierung von Produktionsanlagen und -maschinen ermöglicht, was das Verständnis von Produktionsprozessen und -abläufen erheblich fördert. Zudem ist Factory IO oft kostengünstiger in der Anschaffung und

Lizenzierung, wodurch es insbesondere für Schulungszwecke und kleinere Unternehmen attraktiv wird (vgl. Tabelle 4: „Lizenzkosten Factory IO“).

Lizenzedition	Kosten in €		
	monatlich	jährlich	einmalig
Factory I/O Ultimate Edition	28	278	765
Factory I/O Siemens Edition	17	158	435
Factory I/O Allen-Bradley Edition	17	158	435
Factory I/O Modbus & OPC Edition	17	158	435
Factory I/O MHJ Edition	-	40	109
Factory I/O Automgen Edition	-	40	109
Factory I/O Starter Edition	-	40	109

Tabelle 4: Lizenzkosten Factory IO

Trotz seiner Stärken weist Factory IO ebenfalls einige Schwächen auf. Einer der offensichtlichsten Nachteile besteht in der begrenzten Tiefe der Simulation. Factory IO konzentriert sich primär auf die Visualisierung und Interaktion und bietet weniger Möglichkeiten für tiefgehende Prozesssimulation im Vergleich zu SIMIT. Darüber hinaus weist Factory IO eine begrenzte Integrationstiefe auf und bietet nicht das gleiche Maß an Integration mit Steuerungssystemen wie SIMIT, was die Anwendung in komplexen Automatisierungsprojekten einschränken kann.

7 Fazit und Ausblick

Im Rahmen dieses Projektes konnte exemplarisch dargelegt werden, in welcher Weise Unternehmen ihre Vorgehensweise bei der Entwicklung eines digitalen Zwillings gestalten könnten. Dieser Prozess beginnt mit der Konzeption, gefolgt von der Planungsphase, bevor die eigentliche Umsetzung erfolgt.

Es ist festzuhalten, dass die Wahl zwischen den Softwarepaketen SIMIT und Factory IO in erheblichem Maße von den individuellen Anforderungen und Zielsetzungen abhängt. Insbesondere der Verwendungszweck spielt hierbei eine zentrale Rolle, wobei Factory IO sich besonders für Schulungszwecke eignet, während SIMIT besser für Tests von Automatisierungssystemen geeignet ist. Die Komplexität des Projekts, das Budget und die vorhandene Technologie sind ebenfalls entscheidende Faktoren, die die Wahl beeinflussen. SIMIT ist in der Regel kostenintensiver, bietet jedoch bessere Möglichkeiten für anspruchsvolle Prozesssimulationen. Wenn ein Unternehmen bereits Siemens-Technologie einsetzt, könnte SIMIT aufgrund seiner Integrationstiefe bevorzugt werden. Die Erfahrung und technische Expertise der Mitarbeiter sind weitere wichtige Überlegungen, da SIMIT sich an technisch versierte Anwender richtet, während Factory IO aufgrund seiner Benutzerfreundlichkeit auch von weniger technikaffinen Nutzern verwendet werden kann. Der zeitliche Rahmen und spezifische Projektanforderungen spielen ebenfalls eine bedeutende Rolle. Bei knappen Zeitvorgaben könnte SIMIT möglicherweise nicht die optimale Wahl sein, da erforderliche Schulungen und Einarbeitung zusätzliche Zeit in Anspruch nehmen können. Daher ist es von großer Bedeutung, diese Faktoren sorgfältig zu analysieren, um die bestmögliche Softwarelösung auszuwählen.

Unternehmen in der Industrie 4.0 müssen zukünftig weiterhin digitale Technologien und Trends vorantreiben, um dem steigenden Bedarf an Konsumgütern und dem Ruf nach Fortschritt und Digitalisierung gerecht zu werden. Andernfalls riskieren sie, den Anschluss zu verlieren. Digitale Zwillinge in Kombination mit maschinellem Lernen und künstlicher Intelligenz werden hierbei als wertvolle Werkzeuge dienen und zukünftig eine immer größere Verantwortung übernehmen können.

8 Literaturverzeichnis

- [1] Atlassian. Pty Ltd, „Was ist IT-Asset-Management (ITAM)? Ein Leitfaden | Atlassian,“ 2023. [Online]. Available: <https://www.atlassian.com/de/itsm/it-asset-management#before-the-incident>. [Zugriff am 4 November 2023].
- [2] Bundesamt für Sicherheit in der Informationstechnik (BSI), „BSI - Grundlagen,“ 2023. [Online]. Available: https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Empfehlungen-nach-Angriffszielen/Cloud-Computing/Grundlagen/grundlagen_node.html. [Zugriff am 11 November 2023].
- [3] KUNBUS GmbH, „Ein Überblick zu Buszugriffsverfahren und zur Systematisierung,“ [Online]. Available: <https://www.kunbus.de/einueberblickzubuszugriffsverfahrenundzursystematisierung>. [Zugriff am 4 November 2023].
- [4] Fraunhofer-Institut für Techno- und Wirtschaftsmathematik ITWM, „sys_flyer_Hardware in the loop-HIL_DE.pdf,“ Kaiserslautern, 2016.
- [5] SICK AG, „IO-Link – Grundlagen und Technologie | SICK AG,“ 2023. [Online]. Available: <https://www.sick.com/de/de/io-link-grundlagen-und-technologie/w/io-link-basics-and-technology/>. [Zugriff am 5 November 2023].
- [6] IPH – Institut für Integrierte Produktion Hannover gGmbH, „Lean Management Methoden - Beschreibung und Umsetzung,“ 2023. [Online]. Available: <https://www.iph-hannover.de/de/information/lean-production/methoden-lean-management/#>. [Zugriff am 4 November 2023].
- [7] DATACOM Buchverlag GmbH, „Physik-Engine :: physics engine :: Itwissen.info,“ 14 Januar 2011. [Online]. Available: <https://www.itwissen.info/Physik-Engine-physics-engine.html>. [Zugriff am 6 November 2023].
- [8] Heinze GmbH | NL Berlin | BauNetz, „Redundanz | Gebäudetechnik | Glossar | Baunetz_Wissen,“ [Online]. Available: <https://www.baunetzwissen.de/glossar/r/redundanz-49413>. [Zugriff am 6 November 2023].
- [9] TrainingsManufaktur Dreiklang GmbH & Co. KG, „*Skaleneffekte | TrainingsManufaktur,“ 2023. [Online]. Available: <https://trainingsmanufaktur.de/lexikon/skaleneffekte/>. [Zugriff am 4 November 2023].
- [10] World of VR GmbH, „Stand-Alone: Was bedeutet das bei Geräten und Software? - World of VR,“ 2022. [Online]. Available: <https://worldofvr.de/stand-alone/>. [Zugriff am 6 November 2023].
- [11] Conrad Electronic SE, „IP Schutzklassen und Schutzarten,“ 2021. [Online]. Available: <https://www.conrad.de/de/ratgeber/industrie-40/ip-schutzklassen-und-schutzarten.html>. [Zugriff am 5 November 2023].

- [12] L. Wuttke, „Was ist Data Science? Definition, Anwendung und Beispiele,“ datasolut GmbH, 2023. [Online]. Available: <https://datasolut.com/was-ist-data-science/>. [Zugriff am 8 November 2023].
- [13] INGenium-EMSR GmbH (Hsg.), „Dienstleistungen,“ Juli 2023. [Online]. Available: <https://ingenium-emsr.de/allesimblick.html>. [Zugriff am 25 August 2023].
- [14] C. Reinhardt, Interviewee, [Interview]. 11 Januar 2023.
- [15] S. Manemann, Industrie 4.0, Projekt 6 "Eine Sortieranlage analysieren , simulieren und erweitern", 1. Auflage Hrsg., Braunschweig: Bildungshaus Schulbuchverlage Schroeder Diesterweg Winklers GmbH, 2020.
- [16] T. Bauernhansl, M. ten Hompel und B. Vogel-Heuser, Industrie 4.0 in Produktion, Automatisierung und Logistik, 1. Auflage Hrsg., Wiesbaden: Springer Vieweg, 2014.
- [17] M. Steven, Industrie 4.0, 1. Auflage Hrsg., Stuttgart: Kohlhammer, 2019.
- [18] K. Pfeiffer, „Meilenstein der Automation: Grundmuster des Industrial Internet of Things,“ 01 Juni 2018. [Online]. Available: <https://www.elektrotechnik.vogel.de/meilenstein-der-automation-grundmuster-des-industrial-internet-of-things-a-711219/>. [Zugriff am 14 September 2023].
- [19] A. Roth, Einführung und Umsetzung von Industrie 4.0, 1. Auflage Hrsg., Leinfelden-Echterdingen: Springer Gabler, 2016.
- [20] D. S. Döring, „Industrie 4.0 erklärt | explainIT auf München-Digital-Portal,“ 22 Februar 2020. [Online]. Available: <https://muenchen.digital/blog/explainit-industrie-4-0-erklaert/>. [Zugriff am 19 September 2023].
- [21] R. Langmann, Taschenbuch der Automatisierung, Solingen: Carl Hanser Verlag München, 2017.
- [22] W. Babel, Internet of Things und Industrie 4.0, Wiesbaden: Springer Vieweg Wiesbaden, 2023.
- [23] J. Rech, Ethernet: Technologien und Protokolle für die Computervernetzung, 3. Auflage Hrsg., Heise Verlag, 2014.
- [24] D. Bodnar, „TCP/IP: Was ist das TCP/IP-Modell und wie funktioniert es? | AVG,“ Avast Deutschland GmbH, 4 Juni 2021. [Online]. Available: <https://www.avg.com/de/signal/what-is-tcp-ip>. [Zugriff am 1 Oktober 2023].
- [25] M. Schleipen et. al., Praxishandbuch OPC UA Grundlagen-Implementierung-Nachrüstung-Praxisbeispiele, Würzburg: Vogel Communications Group, 2020.
- [26] Sheldon, „TCPIP vs. OSI: Was ist der Unterschied? | FS Community,“ 6 Juli 2022. [Online]. Available: <https://community.fs.com/de/blog/tcpip-vs-osi-whats-the-difference-between-the-two-models.html>. [Zugriff am 1 Oktober 2023].
- [27] Wikimedia Foundation Inc., „Topologie (Rechnernetz) - Wikipedia,“ 20 März 2013. [Online]. Available: https://de.wikipedia.org/wiki/Topologie_%28Rechnernetz%29. [Zugriff am 1 Oktober 2023].

- [28] P. Helm, *Gebäudesystemtechnik Industrie- und Internet- Kommunikationssysteme, Vorlesungsunterlagen zum Modul Gebäudesystemtechnik der HS Merseburg*, 2019.
- [29] A. Ortwein, *Bussysteme, Vorlesungsunterlagen zum Modul Prozessleittechnik der HS Merseburg*, 2020.
- [30] G. Schnell, *Bussysteme in der Automatisierungs- und Prozesstechnik*, 9. Auflage Hrsg., B. Wiedemann, Hrsg., Wiesbaden: Springer Vieweg, 2019.
- [31] KUNBUS GmbH, „Eine Übersicht zum Feldbussystem PROFIBUS,“ [Online]. Available: <https://www.kunbus.de/eine-uebersicht-zum-feldbussystem-profibus>. [Zugriff am 1 Oktober 2023].
- [32] KUNBUS GmbH, „PROFINET – Bedeutung und Merkmale,“ [Online]. Available: <https://www.kunbus.de/profinet-grundlagen>. [Zugriff am 1 Oktober 2023].
- [33] KUNBUS GmbH, „AS-Interface / AS-i (actuator sensor interface),“ [Online]. Available: <https://www.kunbus.de/as-interface>. [Zugriff am 1 Oktober 2023].
- [34] Bihl+Wiedemann GmbH, „ASi Kabel einfachste Anschlusstechnik - Bihl+Wiedemann GmbH,“ 2023. [Online]. Available: <https://www.bihl-wiedemann.de/de/applikationen/as-interface/asi-kabel-einfach-effizient-und-kostenguenstig#c-id--scroll--top>. [Zugriff am 2 November 2023].
- [35] T. Riedelsheimer et al., *DIGITAL TWIN READINESS ASSESSMENT*, M. Samarajiwa et al., Hrsg., Berlin: Fraunhofer Verlag, 2020.
- [36] J. Krauß, A. Schmetz und A. Fitzner et al., *Der Digitale Zwilling in der Batteriezellfertigung*, F. F. B. FFB, Hrsg., Münster, 2023.
- [37] M. Kurpicz-Briki et al., „Wie Digitale Zwillinge in der Industrie eingesetzt werden,“ *SocietyByte*, p. 5, 2023.
- [38] Siemens AG, „Flyer: SIMIT Simulation Platform,“ Karlsruhe, 2019.
- [39] Bucher Netzwerke GmbH, „Digitale Verhaltensmodelle mit SIMIT,“ 2022. [Online]. Available: <https://www.bucher-netzwerke.de/simulation-vibn/digitale-verhaltensmodelle-mit-simit.html>. [Zugriff am 13 10 2023].
- [40] Siemens AG, „Bedienhandbuch SIMIT Simulation Platform (V10.1),“ Nürnberg, 2019.
- [41] Siemens AG, „Konfiguration einer SIMIT UNIT im PCS 7 Umfeld,“ 2019.
- [42] MHJ-Software GmbH & Co. KG, „FACTORY I/O | Software | MHJ-Software GmbH & Co. KG,“ 2023. [Online]. Available: <https://www.mhj-online.de/software/factory-io/>. [Zugriff am 31 Oktober 2023].
- [43] Real Games, „I/O Drivers | Factory I/O,“ 2023. [Online]. Available: <https://docs.factoryio.com/manual/drivers/>. [Zugriff am 31 Oktober 2023].
- [44] b. S. G. d. & ident, „RFID | bsi Systeme GmbH data & ident,“ 2023. [Online]. Available: <https://bsisysteme.de/de/bsiident/rfid/>. [Zugriff am 30 Oktober 2023].

- [45] M. Schneider, „RFID im Einzelhandel: Lesegeräte und Etiketten | Logiscenter | Logiscenter,“ 12 April 2022. [Online]. Available: <https://www.logiscenter.de/news/rfid-mode-einzelhandel>. [Zugriff am 30 Oktober 2023].
- [46] RFID, Inc., „UHF 433 MHz Active RFID Tags | RFID, Inc.,“ 2023. [Online]. Available: <https://www.rfidinc.com/uhf-433-mhz-active-rfid-tags>. [Zugriff am 30 Oktober 2023].
- [47] CODESYS GmbH, „Unternehmen,“ 2023. [Online]. Available: <https://de.codesys.com/unternehmen.html>. [Zugriff am 24 Oktober 2023].
- [48] CODESYS GmbH, „CODESYS - IEC 61131-3 Programmierwerkzeug für SPS & Industrie-Steuerungen,“ 2023. [Online]. Available: <https://de.codesys.com/produkte/codesys-engineering/development-system.html>. [Zugriff am 24 Oktober 2023].
- [49] S. Manemann, „Projektdateien: Dateien Projekt 06: „Eine Sortieranlage analysieren, simulieren und erweitern“,“ 6 April 2014. [Online]. Available: <https://www.xplordna.net/mod/folder/view.php?id=791>. [Zugriff am 29 Oktober 2023].

Anlagen

Anlage 1 Ausschnitt Treiberbaustein Factory IO

Anlage 2 Codesys Programm lokale Variablenliste

Anlage 3 Codesys Programm Netzwerk 1 bis 6

Anlage 4 Codesys Programm Netzwerk 7 und 8

Anlage 5 Codesys Programm Netzwerk 9 und 10

Anlage 6 Codesys Programm Netzwerk 11

Anlage 7 Codesys Programm Netzwerk 12 und 13

Anlage 8 Codesys Programm Netzwerk 14 bis 18

Anlage 9 Codesys Programm Netzwerk 19 bis 21

Anlage 10 Codesys Programm Netzwerk 22 bis 26

Anlage 11 Codesys Programm Netzwerk 27 bis 33

Anlage 12 Codesys Programm Netzwerk 34 bis 36

Anlage 13 Codesys globale Variablenliste „FIO“

Anlage 1 Ausschnitt Treiberbaustein Factory IO

Server: opc.tcp://DESKTOP-LHIF4FE:4 840 (58)		
Aufzug fährt	iAufzugfaehrt	
Auto	iAuto	
Gewicht	iGewicht	
Klemme zu	iKlemmeZu	
Lichtschranke	iLichtschranke	
Manuell	iManuell	
Not Aus	iNotAus	
Paket Eingang	iPaketEingang	
Palette Ausgang	iPaletteAusgang	
Palette bereit	iPaletteBereit	
Palette Eingang	iPaletteEingang	
Reset	iReset	
RFID2 Daten lesen	iRFID2Lesen	
FACTORY I/O (Running)	iRUNNING_Knopf	
Schieber limit	iSchieberLimit	
Start	iStart	
An Station1	iStation1	
An Station2	iStation2	
Stop	iStop	
Tor_geöffnet	iTorGeoeffnet	
Tür limit	iTuerLimit	
	oAufzugBisLimit	fahre bis Limit
	oAufzugHoch	Aufzug hoch
	oAufzugRunter	Aufzug runter
	oBand1	! Band1
	oBand2	Band2
	oBand3	Band3

Anlage 2 Codesys Programm lokale Variablenliste

POU: PLC_PRG

```
1  PROGRAM PLC_PRG
2  VAR
3      SR_0          : SR ;
4      SR_1          : SR ;
5      SR_2          : SR ;
6      SR_4          : SR ;
7      SR_3          : SR ;
8      SR_5          : SR ;
9      SR_6          : SR ;
10     SR_7          : SR ;
11     SR_8          : SR ;
12     SR_9          : SR ;
13     SR_10         : SR ;
14
15     RS_0          : RS ;
16     RS_1          : RS ;
17     RS_2          : RS ;
18     RS_3          : RS ;
19     RS_4          : RS ;
20     RS_5          : RS ;
21     RS_6          : RS ;
22     RS_7          : RS ;
23     RS_8          : RS ;
24
25     TON_0         : TON ;
26     T0            : TIME ;
27     TON_1         : TON ;
28     T1            : TIME ;
29     TON_2         : TON ;
30     T2            : TIME ;
31     TON_3         : TON ;
32     T3            : TIME ;
33     TON_4         : TON ;
34     T4            : TIME ;
35     TON_5         : TON ;
36     T5            : TIME ;
37     TON_6         : TON ;
38     T6            : TIME ;
39     TON_7         : TON ;
40     T7            : TIME ;
41     TON_8         : TON ;
42     T8            : TIME ;
43     TON_9         : TON ;
44     T9            : TIME ;
45     TON_10        : TON ;
46     T10           : TIME ;
47     TON_11        : TON ;
48     T11           : TIME ;
49     TON_12        : TON ;
50     T12           : TIME ;
51     TON_13        : TON ;
52     T13           : TIME ;
53     TON_14        : TON ;
54     T14           : TIME ;
55     TON_15        : TON ;
56     T15           : TIME ;
57     TON_16        : TON ;
58     T16           : TIME ;
59     TON_17        : TON ;
60     T17           : TIME ;
61     TON_18        : TON ;
62     T18           : TIME ;
63     TON_19        : TON ;
64     T19           : TIME ;
65     TON_20        : TON ;
66     T20           : TIME ;
67
68     TOF_0         : TOF ;
69     To0           : TIME ;
70     TOF_1         : TOF ;
71     To1           : TIME ;
72     TOF_2         : TOF ;
73     To2           : TIME ;
74     TOF_3         : TOF ;
75     To3           : TIME ;
76
77     START         : BOOL ;
78     Manuell_trigger : BOOL ;
79     Umlenker_trigger : BOOL ;
80     Schieber_trigger : BOOL ;
81
82     CTU_0         : CTU ;
83     Zaehlwert     : WORD ;
84     CTU_1         : CTU ;
85     TuerAuf       : WORD ;
86     CTU_2         : CTU ;
87     AufzugFaehrt  : WORD ;
88     CTU_3         : CTU ;
89     Schieberzahl  : WORD ;
90     CTU_4         : CTU ;
```

Anlage 3 Codesys Programm Netzwerk 1 bis 6

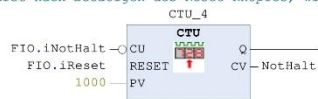
POU: PLC_PRG

```

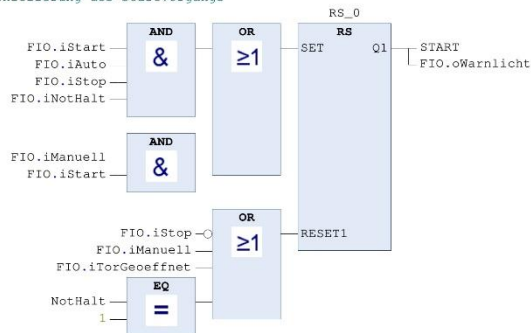
91     NotHalt      : WORD ;
92
93     R_TRIG_0     : R_TRIG ;
94
95     KartonM      : INT := 100 ;
96     KartonL      : INT := 200 ;
97
98     END_VAR
99

```

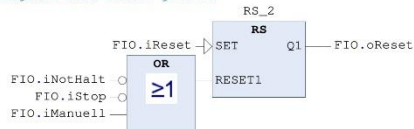
1 Zähler Not-Halt
wird Not-Halt betätigt, zählt Zähler 1 hoch und sperrt somit den START Baustein.
Erst nach Betätigen des Reset-Knopfes, wird der Zähler zurückgesetzt und START Baustein wird freigegeben.



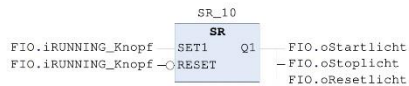
2 Startbaustein
Initiierung des Startvorgangs



3 Resetbedingung
Programm wird zurück gesetzt



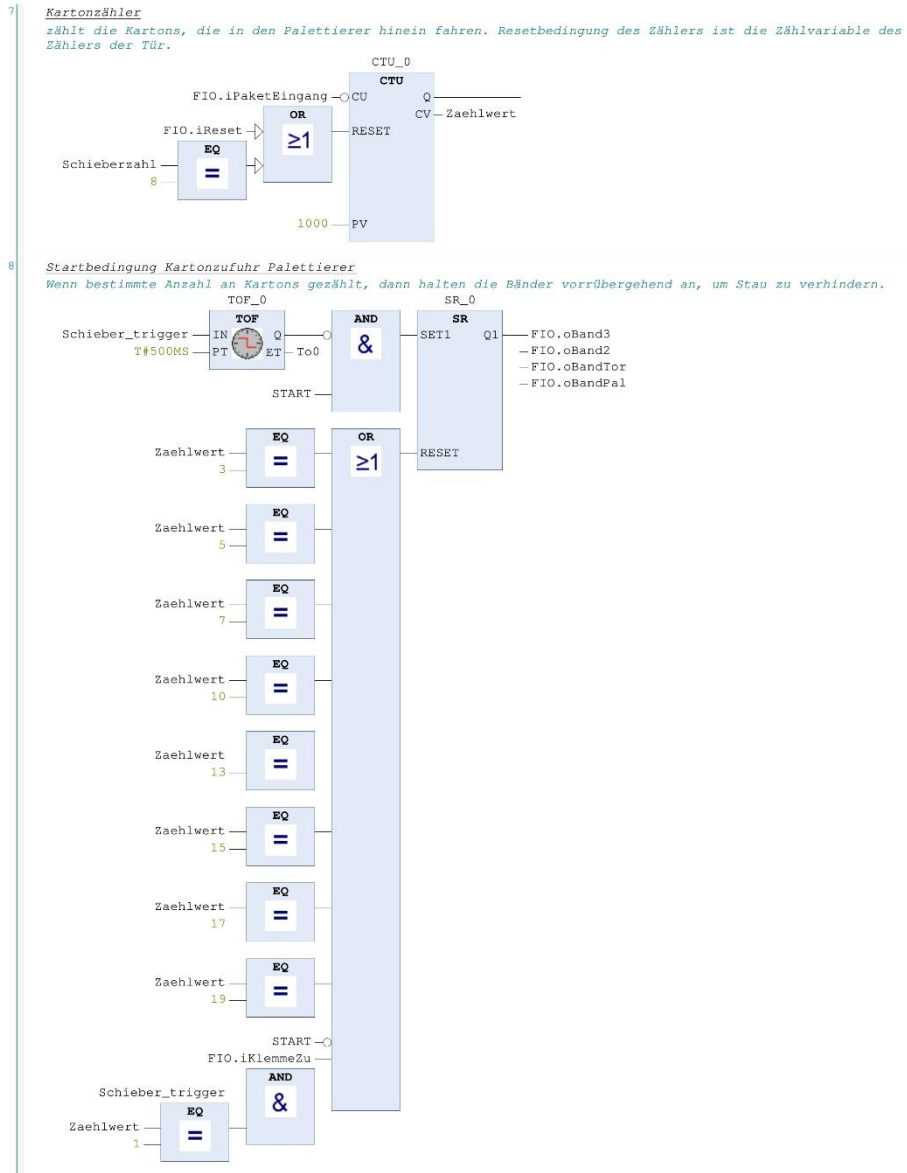
4 Knopflichter



5
6 1. Palettierer

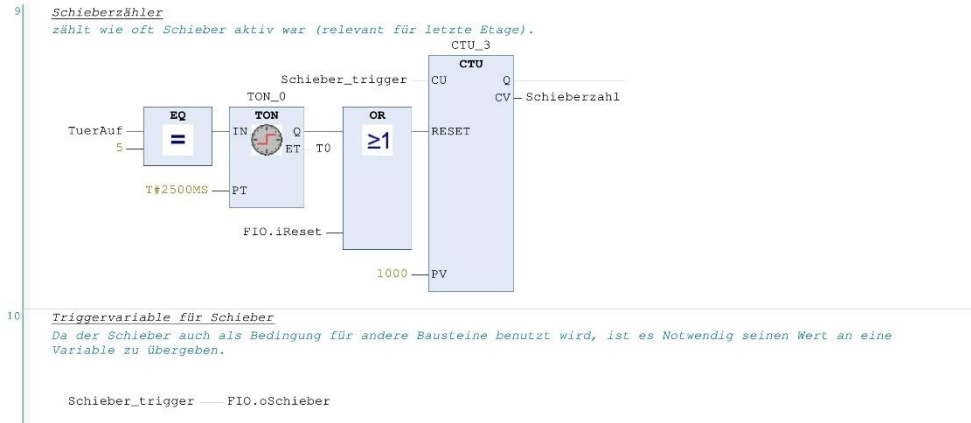
Anlage 4 Codesys Programm Netzwerk 7 und 8

POU: PLC_PRG



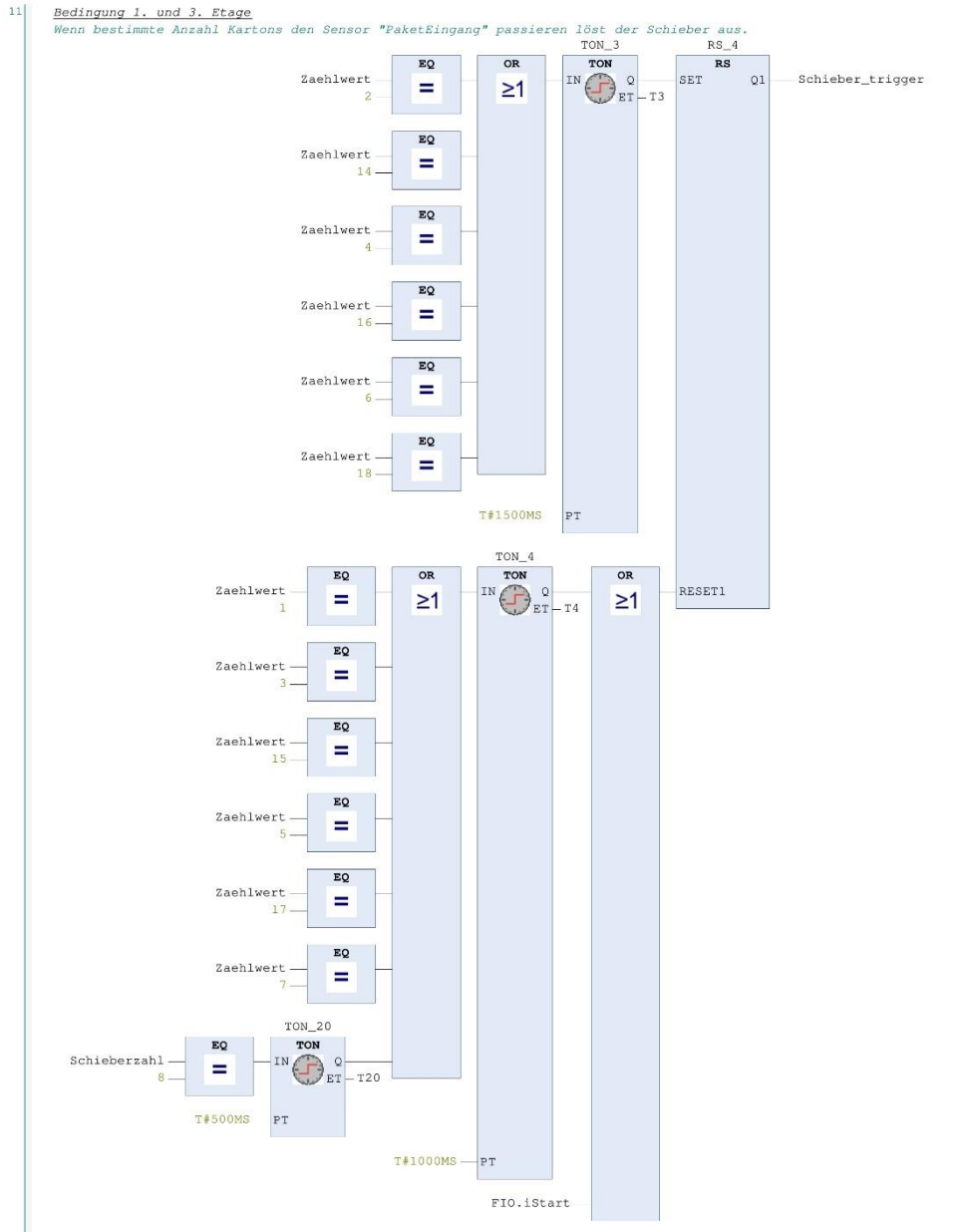
Anlage 5 Codesys Programm Netzwerk 9 und 10

POU: PLC_PRG



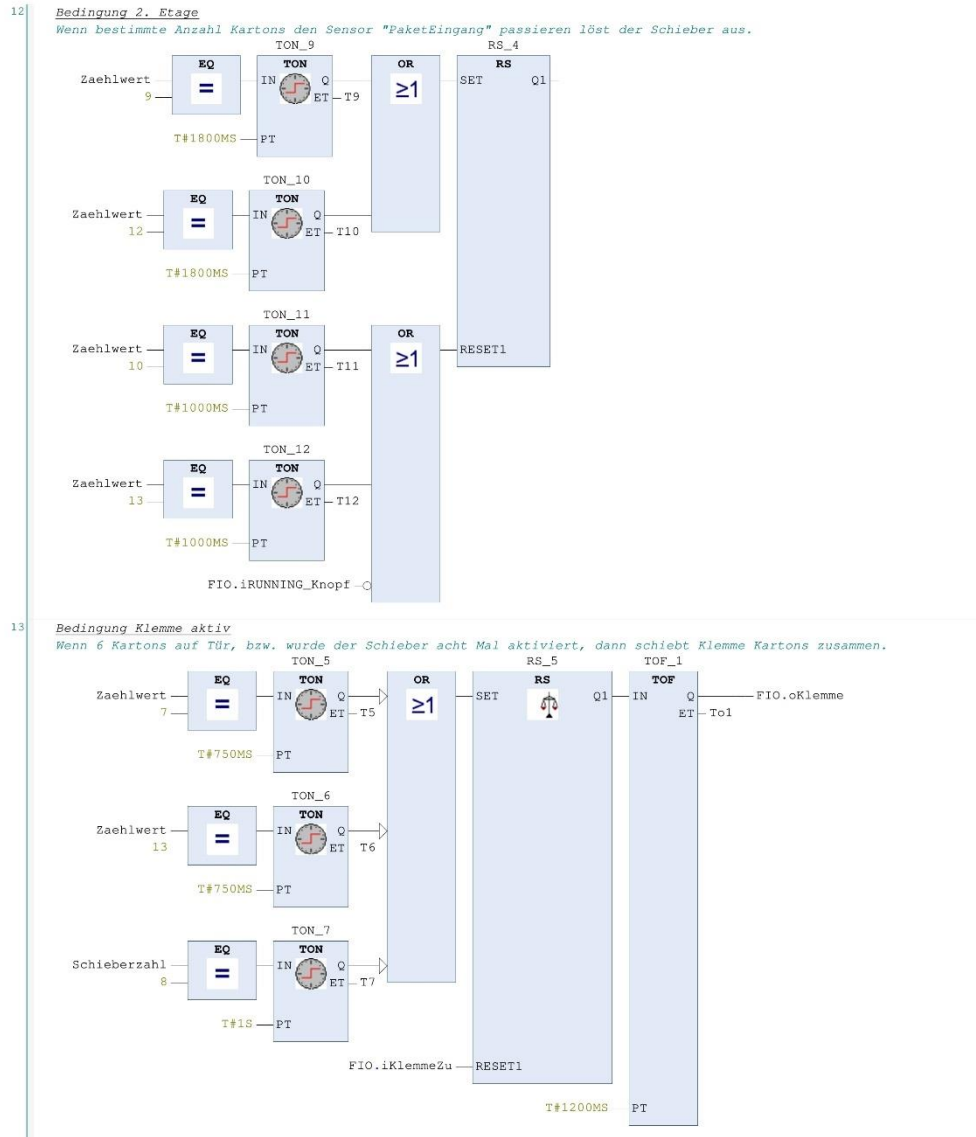
Anlage 6 Codesys Programm Netzwerk 11

POU: PLC_PRG



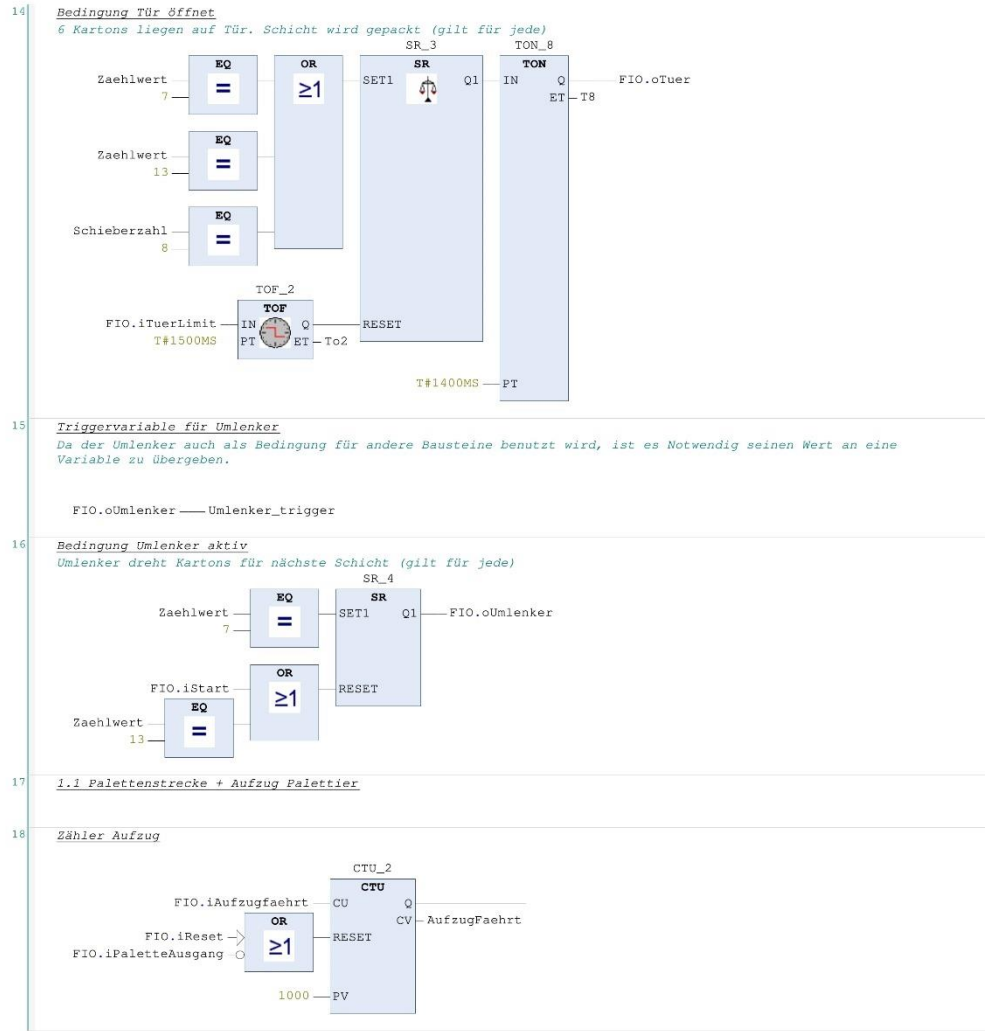
Anlage 7 Codesys Programm Netzwerk 12 und 13

POU: PLC_PRG



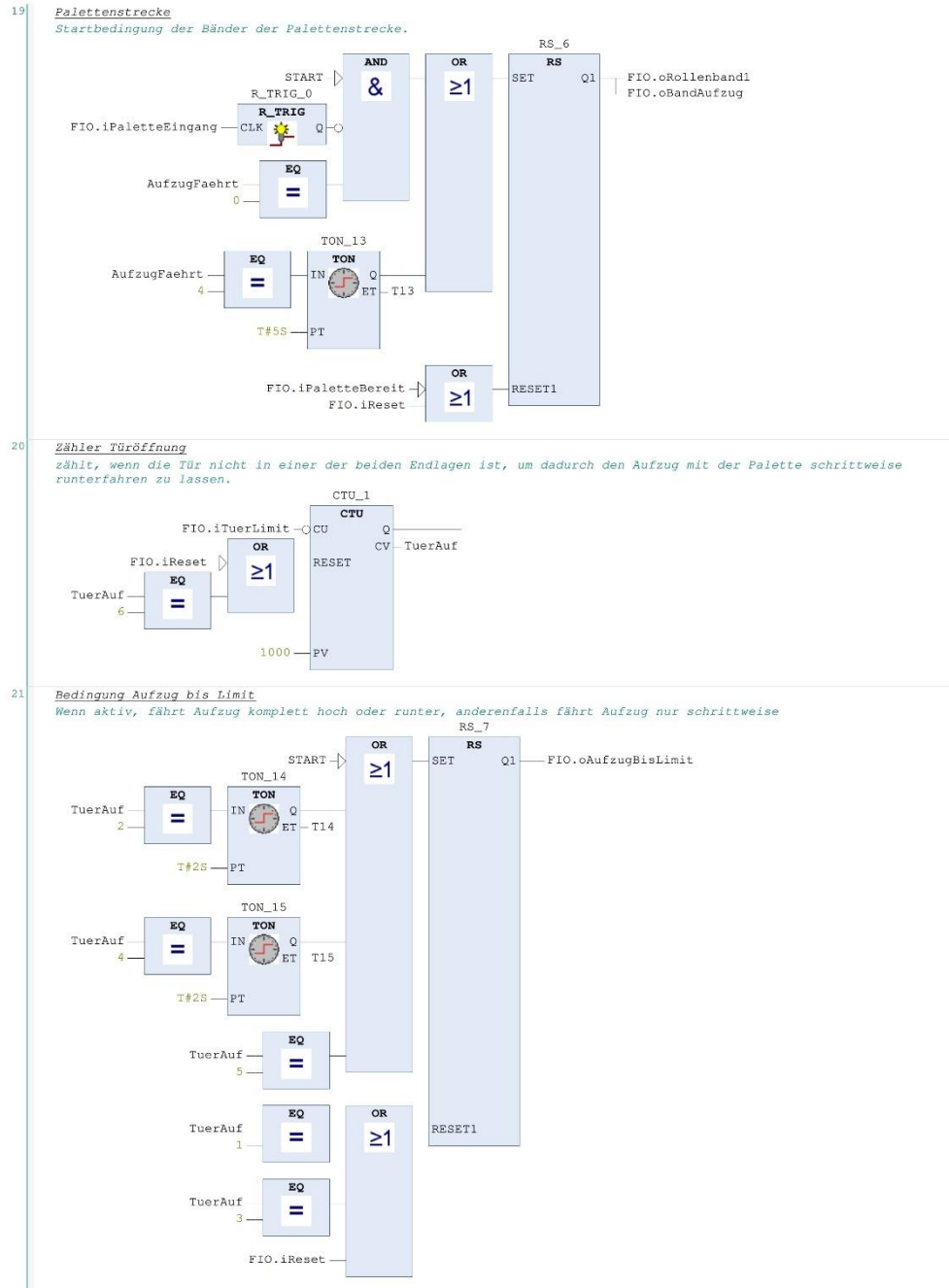
Anlage 8 Codesys Programm Netzwerk 14 bis 18

POU: PLC_PRG



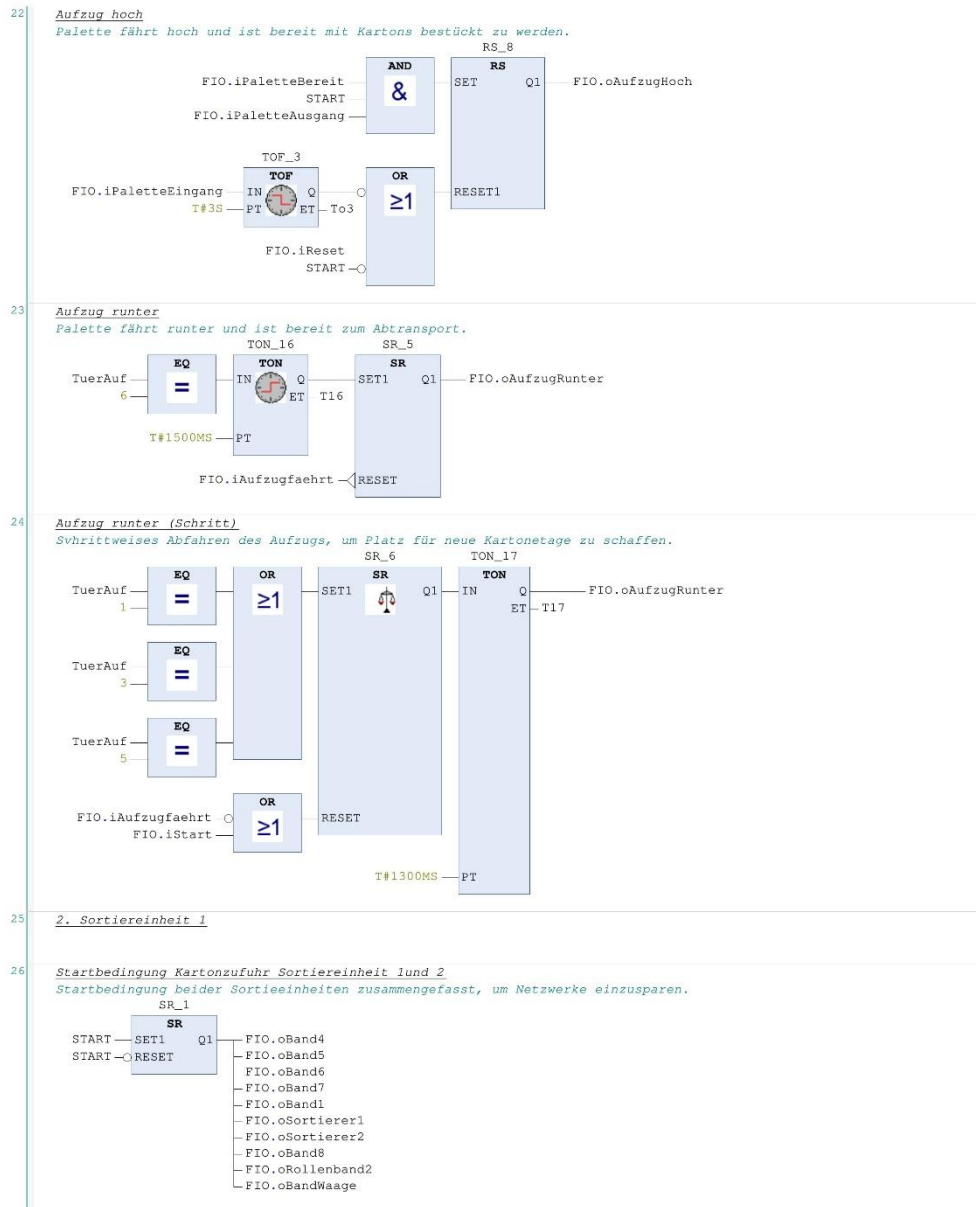
Anlage 9 Codesys Programm Netzwerk 19 bis 21

POU: PLC_PRG



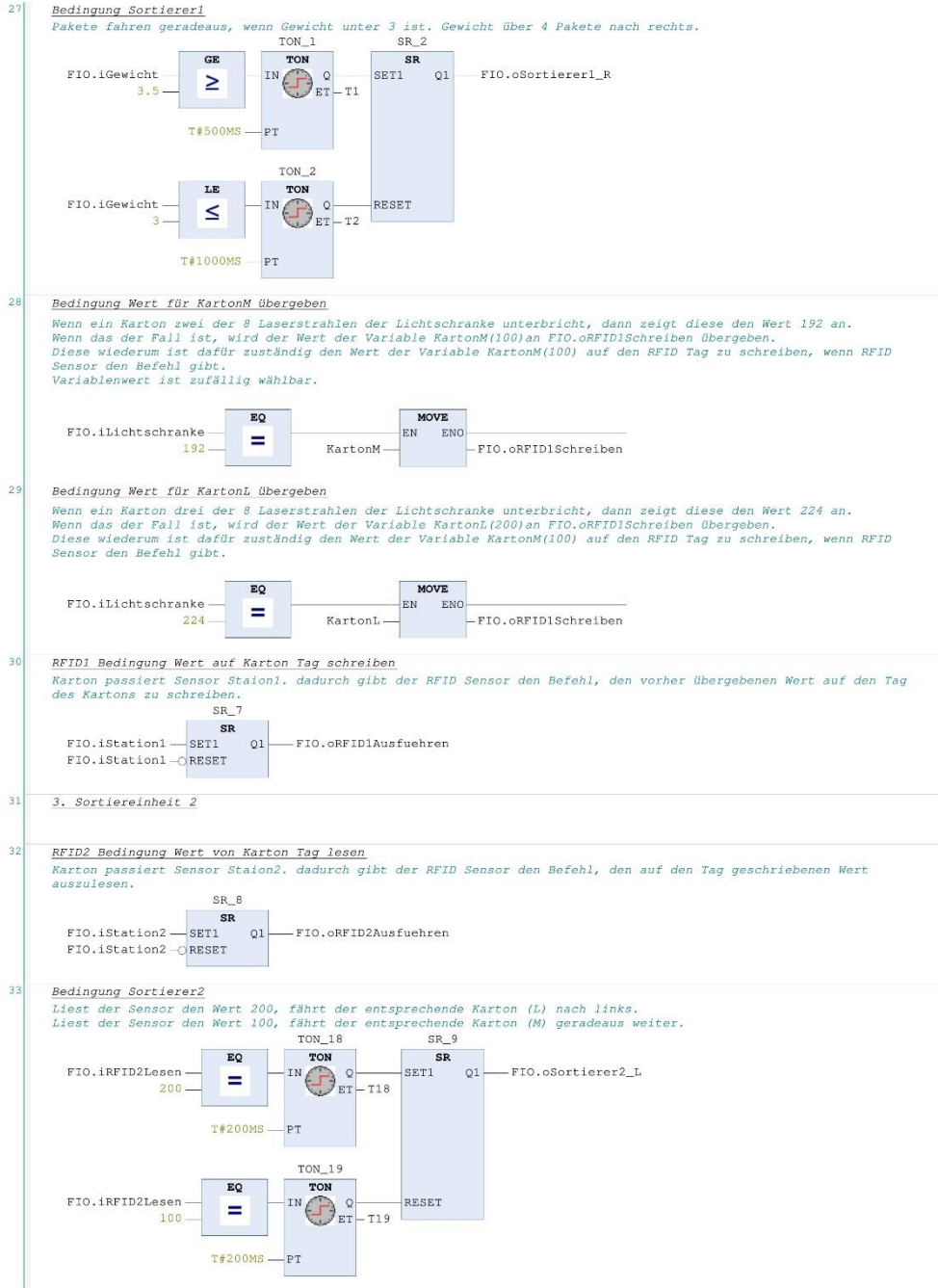
Anlage 10 Codesys Programm Netzwerk 22 bis 26

POU: PLC_PRG

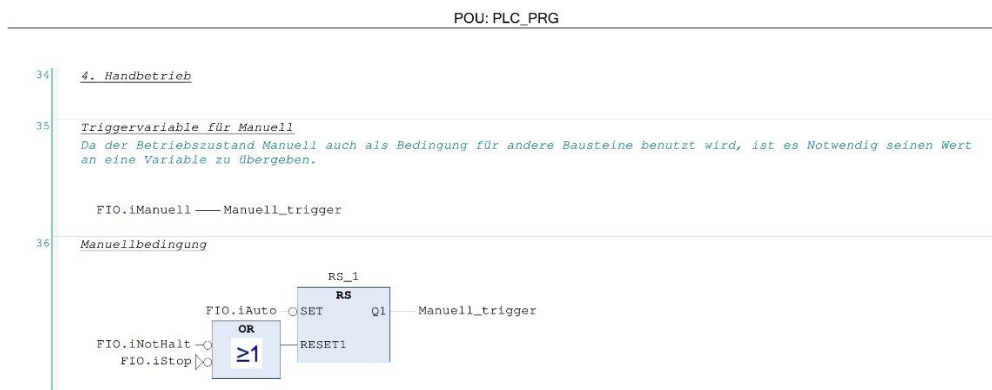


Anlage 11 Codesys Programm Netzwerk 27 bis 33

POU: PLC_PRG



Anlage 12 Codesys Programm Netzwerk 34 bis 36



Anlage 13 Codesys globale Variablenliste „FIO“

```
1      {attribute 'qualified_only'}
2      VAR_GLOBAL
3          iStart          :   BOOL ;
4          iStop           :   BOOL ;
5          iReset          :   BOOL ;
6          iNotHalt        :   BOOL ;
7          iAuto           :   BOOL ;
8          iManuell        :   BOOL ;
9          iStation1       :   BOOL ;
10         iStation2       :   BOOL ;
11         iAufzugfaehrt   :   BOOL ;
12         iGewicht        :   REAL ;
13         iKlemmeZu       :   BOOL ;
14         iPaketEingang   :   BOOL ;
15         iSchieberLimit  :   BOOL ;
16         iTuerLimit      :   BOOL ;
17         iPaletteEingang :   BOOL ;
18         iPaletteAusgang :   BOOL ;
19         iPaletteBereit  :   BOOL ;
20         iRUNNING_Knopf  :   BOOL ;
21         iTorGeoeffnet   :   BOOL ;
22         iRFID2Lesen     :   DINT ;
23         iLichtschranke  :   DINT ;
24
```

```

25      oStartlicht      :   BOOL ;
26      oStoplicht      :   BOOL ;
27      oResetlicht     :   BOOL ;
28      oZaehler        :   INT  ;
29      oWarnlicht      :   BOOL ;
30      oBand1          :   BOOL ;
31      oBand2          :   BOOL ;
32      oBand3          :   BOOL ;
33      oBand4          :   BOOL ;
34      oBand5          :   BOOL ;
35      oBand6          :   BOOL ;
36      oBand7          :   BOOL ;
37      oBand8          :   BOOL ;
38      oBandPal        :   BOOL ;
39      oBandWaage      :   BOOL ;
40      oBandTor        :   BOOL ;
41      oKlemme         :   BOOL ;
42      oSchieber       :   BOOL ;
43      oSortierer1     :   BOOL ;
44      oSortierer1_R   :   BOOL ;
45      oSortierer2     :   BOOL ;
46      oSortierer2_L   :   BOOL ;
47      oTuer           :   BOOL ;
48      oReset          :   BOOL ;
49      oUmlenker       :   BOOL ;
50      oPause          :   BOOL ;
51      oRollenband1    :   BOOL ;
52      oRollenband2    :   BOOL ;
53      oBandAufzug     :   BOOL ;
54      oAufzugHoch     :   BOOL ;
55      oAufzugRunter   :   BOOL ;
56      oAufzugBisLimit :   BOOL ;
57      oRFID1Schreiben :   DINT ;
58      oRFID1Befehl    :   INT  ;
59      oRFID1Ausfuehren :  BOOL ;
60      oRFID2Befehl    :   INT  ;
61      oRFID2Ausfuehren :  BOOL ;
62      END_VAR

```