

Diplomarbeit

Visualisierung und Simulation einer Kleinserienfertigungsanlage

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Diplom-Ingenieurs

am

Institut für Konstruktionswissenschaften und Technische Logistik
der Fakultät für Maschinenwesen und Betriebswissenschaften
der Technischen Universität Wien

unter der Leitung von
Ass.Prof. Dipl.-Ing. Dr.techn. Manfred Grafinger

durch

Wolfgang Meißnitzer
Matrikel Nr.: 0126097
Flurschützstraße 15/10
1120 Wien

Wien, September 2008

Danksagung

An dieser Stelle möchte ich mich bei all jenen bedanken, die mich bei der Erstellung der vorliegenden Diplomarbeit unterstützt haben.

Ich danke Herrn Ass.Prof. Dipl.-Ing. Dr.techn. Manfred Grafinger, der diese Diplomarbeit hervorragend betreut und mich bei Fragen immer wieder unterstützt hat. Auch war es mir durch ihn möglich, den auf dem Institut vorhandenen leistungsstarken Rechner für die Simulation zu nutzen.

Weiters bedanke ich mich bei meinen Eltern, durch deren Unterstützung es möglich wurde, mein Studium erfolgreich zu beenden. Auch allen Freunden, die mich durch meine Studienzeit begleitet haben, möchte ich an dieser Stelle danken. Zum Schluss möchte ich mich ganz besonders bei meiner Frau Ute für ihre Unterstützung bedanken.

Kurzfassung

Der Einsatz von Simulationen ist heute auch bei Entwicklungen und Optimierungen von technischen Systemen nicht mehr wegzudenken. Die Simulation ist eine möglichst realitätsnahe Abbildung vom Geschehen der Wirklichkeit. So wird durch Abstraktion ein Modell geschaffen, an dem zielgerichtet experimentiert werden kann. Ausgehend von diesen Ergebnissen wird auf das Verhalten der realen Vorgänge geschlossen.

Der Hauptteil dieser Diplomarbeit beschäftigt sich mit der Erstellung einer Simulation einer Kleinserienfertigungsanlage. Der physische Aufbau dieser Kleinserienfertigung wird im Rahmen einer Lehr- und Lernproduktion an der HTBL Wien 10 errichtet. Die Simulation ahmt die Montageabläufe der Kleinserienfertigung nach.

In der Lehr- und Lernproduktion wird der Unternehmensalltag eines produzierenden Betriebes nachgebildet, durch die Simulation wird es möglich, dabei eine größere Variantenvielfalt virtuell zu erzeugen. So wird den Lernenden die Möglichkeit geboten, Handlungs- und Entscheidungsfähigkeiten in lebensnahen, jedoch entlasteten Situationen zu entwickeln.

Im Hauptteil wird die simulierte Arbeitszelle beschrieben und die vorhandenen Montagemodule samt Verknüpfung mit ihren steuerbaren Mechanismen erklärt.

Im letzten Kapitel erfolgt die Beschreibung der Roboterprogrammierstellung und exemplarisch wird ein Unterprogramm mittels eines Flussdiagramms verdeutlicht. Um ein Abspielen der Simulation auch ohne die Software COSIMIR zu ermöglichen, wurde die Simulation als Video-File abgespeichert. Dadurch kann sie einfach von einer DVD abgespielt werden.

Abstract

The main part of this thesis deals with the creation of a simulation of a small batch production plant. The physical structure of this small batch production is built up within a didactic production at the HTBL Wien 10. There the everyday working life of a producing company is imitated. Due to the simulation it is possible to virtually create a bigger variety within the didactic production.

The simulation replicates the installation process of the small batch production. This thesis describes the simulated working cell and the assembly devices including their linkage with controllable mechanisms.

In order to play the simulation even without the software COSIMIR it was necessary to save it in form of a video file. Thus the simulation can be played simply by a DVD.

Inhaltsverzeichnis

Abbildungsverzeichnis	3
Tabellenverzeichnis.....	4
Abkürzungen	4
1 Einleitung	6
1.1 Aufgabenstellung.....	6
1.2 Wahl der Software	6
1.3 Ziel der Diplomarbeit	6
2 Kooperationspartner	8
2.1 Technik Akademie	8
2.2 Höhere Technische Bundeslehranstalt Wien 10.....	8
2.3 Forschungsgruppe Maschinenbauinformatik und Virtuelle Produktentwicklung	9
2.4 Festo Ges.m.b.H.	9
3 Lehr- und Lernproduktion.....	10
4 Von der Simulation zum Abbild der Realität.....	12
4.1 Begriffsbestimmung	12
4.2 Die Simulation.....	13
4.3 Digitale Fabrik.....	16
4.3.1 Von der Simulation zur Digitalen Fabrik.....	16
4.3.2 Virtuelle Inbetriebnahme – Die Abkürzung.....	18
5 COSIMIR Professional	22
6 Beschreibung der Arbeitszelle	25
6.1 Einlesen der Daten in COSIMIR.....	25
6.2 Montagevorrichtungen	26

Inhalt	2
6.2.1 Montagemodul Kolben.....	26
6.2.2 Montagemodul Deckel	29
6.2.3 Montagemodul Kolbenstange	30
6.2.4 Montagemodul Zylinder.....	32
6.3 Modellierung steuerbarer Mechanismen	34
6.3.1 Roboter.....	34
6.3.2 Greifer	35
6.3.3 Zweiwegeschubzylinder.....	35
6.3.4 Drehzylinder.....	36
6.3.5 Replikator.....	36
6.3.6 Fließband.....	37
6.4 Simulationsumgebung	38
7 Programm- und Videoerstellung.....	41
7.1 Programmierung.....	41
7.2 Programmbeschreibung.....	42
7.3 Aufzeichnen der Simulation als Video.....	46
7.4 Aufbau der Ordner-Struktur	49
8 Zusammenfassung und Ausblick	51
9 Anhang	53
9.1 Verwendete Befehle	53
9.2 Programmcode.....	54
10 Literaturverzeichnis.....	91

Abbildungsverzeichnis

Abbildung 1: Lehr- und Lernproduktion HTBL Wien 10	10
Abbildung 2: Pneumatikzylinder	11
Abbildung 3: Ausbildung anhand einer Simulation im virtuellen Raum.....	14
Abbildung 4: Anwendungsfelder der Simulation	15
Abbildung 5: Inbetriebnahme im Produktionsanlauf	17
Abbildung 6: Gegensätzliche Vorgehensweise bei Planung und Inbetriebnahme...	18
Abbildung 7: Einordnung der neuen Meilensteine	19
Abbildung 8: Fabrikplanungszyklus	20
Abbildung 9: Aufwand und Produktionsstart	21
Abbildung 10: Hauptfenster COSIMIR	22
Abbildung 11: Montagemodul Kolben	27
Abbildung 12: Montageschritte Kolben.....	28
Abbildung 13: Montagemodul Deckel.....	29
Abbildung 14: Montageschritte Abschlussdeckel.....	30
Abbildung 15: Montageschritte Lagerdeckel.....	30
Abbildung 16: Montagemodul Kolbenstange	31
Abbildung 17: Montageschritte Kolbenstange.....	31
Abbildung 18: Montagemodul Zylinder	32
Abbildung 19: Montageschritte Pneumatikzylinder	33
Abbildung 20: Mitsubishi Roboter	34
Abbildung 21: Reichweite des Roboters.....	39
Abbildung 22: Reflexlichttaster	39
Abbildung 23: Arbeitszelle	40
Abbildung 24: Allgemeine Vorgehensweise zur Programmerstellung.....	42
Abbildung 25: Projektverwaltung	45
Abbildung 26: Schaltflächen Kamerafahrt.....	46
Abbildung 27: Dialogfeld Kamerafahrt	47
Abbildung 28: Dialogfeld Einstellungen – Video.....	47
Abbildung 29: Simulationseinstellungen	48

Tabellenverzeichnis

Tabelle 1: Ein-/Ausgänge Montagemodul Kolben	27
Tabelle 2: Ein-/Ausgänge Montagemodul Deckel	29
Tabelle 3: Ein-/Ausgänge Montagemodul Kolbenstange	31
Tabelle 4: Ein-/Ausgänge Montagemodul Zylinder	33
Tabelle 5: Greifer	35
Tabelle 6: Zweiwegeschubzylinder.....	35
Tabelle 7: Drehzylinder.....	36
Tabelle 8: Replikator.....	37
Tabelle 9: Fließband.....	37
Tabelle 10: Programmzuordnung.....	44
Tabelle 11: Ordner-Struktur.....	49

Abkürzungen

Die folgenden Abkürzungen werden in der Diplomarbeit und der Simulation verwendet.

AD	Abschlussdeckel
ADmD	Abschlussdeckel mit Dichtung
GB	Gleitbuchse
GPK	Greifpunkt
IBN	Inbetriebnahme
K	Kolben
KmD	Kolben mit Dichtung
KS	Kolbenstange
KSmK	Kolbenstange mit Kolben
LD	Lagerdeckel
LDmD	Lagerdeckel mit Dichtung
MM	Montagemodul
NR	Nutring
OR	O-Ring

SchraubenAD	Schrauben für Abschlussdeckel
SchraubenLD	Schrauben für Lagerdeckel
SOP	Start-Of-Production
SR	Schraube
vIBN	virtuelle Inbetriebnahme
vSOP	virtueller Start-Of-Production
ZR	Zylinderrohr
ZYL6x80	Zylinder mit 6 mm Stangendurchmesser und 80 mm Hub
ZYL6x100	Zylinder mit 6 mm Stangendurchmesser und 100 mm Hub
ZYL6x120	Zylinder mit 6 mm Stangendurchmesser und 120 mm Hub
ZYL8x80	Zylinder mit 8 mm Stangendurchmesser und 80 mm Hub
ZYL8x100	Zylinder mit 8 mm Stangendurchmesser und 100 mm Hub
ZYL8x120	Zylinder mit 8 mm Stangendurchmesser und 120 mm Hub

1 Einleitung

1.1 Aufgabenstellung

Für den Aufbau einer Lehr- und Lernproduktion der Technik Akademie¹ soll eine Kleinserienfertigungsanlage von pneumatischen Zylindern visualisiert und simuliert werden. So wird den Auszubildenden die Möglichkeit gegeben Rückschlüsse zwischen der Simulation und der realen Fertigung zu ziehen. Beispielsweise können Fehler im Zusammenbau des Zylinders in der Simulation nachgeahmt werden, diese und die Konsequenzen daraus werden dann auf den realen Arbeitsplätzen besprochen. Es kommt zu einer Interaktion zwischen der Simulation und der Wirklichkeit, wobei immer aus der Simulation in die Wirklichkeit verwiesen wird und dort weiter gelehrt, analysiert und entschieden wird. Die virtuelle Umgebung unterstützt das Lernen, anstelle eines Textes tritt der virtuelle Raum. Die visuelle Darstellung am PC motiviert und fördert den Lernprozess. So wird es möglich, verschiedene Fehlerszenarien durchzuspielen. Die Lernenden sollen darauf reagieren und so die Konsequenzen ihres Handelns erkennen.

1.2 Wahl der Software

Die Basis der Simulation bildet die Simulationssoftware COSIMIR Professional Version 4.2, welche eine Modellierung, Visualisierung und Simulation von Einzelplätzen bis hin zu ganzen Fabrikstraßen ermöglicht. Die gesamte Simulation der Kleinserienfertigungsanlage wurde mit dieser Software erstellt. Die Software wurde vom Kooperationspartner Festo Didactic zur Verfügung gestellt.

1.3 Ziel der Diplomarbeit

Ziel dieser Diplomarbeit ist die Modellierung und die Programmierung der Montageabläufe von sechs Zusammenbauvarianten der Pneumatikzylinder in der Simulationssoftware COSIMIR.

¹ Erklärungen zu den Kooperationspartnern siehe Seite 8.

Die Simulation sollte als Video-File abgespeichert werden, um sie unabhängig von der Simulationssoftware nutzen zu können.

Die Simulationen soll zu einem späteren Zeitpunkt ergänzt und erweitert werden können, deshalb muss darauf geachtet werden, dass die nötige Einarbeitungszeit einer betrauten Person für spätere Ergänzungen minimal gehalten wird.

2 Kooperationspartner

Die wesentlichen Kooperationspartner sind

- die Technik Akademie, welche die Lehr- und Lernproduktion initiiert hat,
- die HTBL Wien 10, an welcher die Anlage installiert wird,
- die Forschungsgruppe Maschinenbauinformatik und Virtuelle Produktentwicklung, welche die Module konstruiert hat und
- die Festo Ges.m.b.H., welche die Simulationssoftware COSIMIR und viele Komponenten für den physischen Aufbau der Lehr- und Lernproduktion zur Verfügung stellt.

2.1 Technik Akademie

Die Technik Akademie Vienna Region (TAVR) ist in den letzten Jahren ein international anerkannter Partner für Aus- und Weiterbildungsaufgaben geworden. Das Bildungsangebot zieht sich von Fachkräften und -technikern hin zu Betriebsingenieuren. Die Abschlüsse reichen vom Fachabschluss bis zum Master of Business Administration (MBA) an der Donau Universität Krems.

Über Schwesterngesellschaften baute die TAVR ihre Verbindungen zu den Regionen Brno (CZ), Bratislava (SK) und Győr (H) aus und kann dadurch auch grenzüberschreitende Ausbildungen anbieten.²

2.2 Höhere Technische Bundeslehranstalt Wien 10

Die HTBL Wien 10 ist in Wien Favoriten angesiedelt und es werden ca. 1.100 Schüler in fünf Abteilungen unterrichtet.

Die Lehr- und Lernproduktion wird in den Räumen der HTBL Wien 10 installiert. Hier werden die Module und Produkte nach den Konstruktionszeichnungen der TU Wien gebaut und später auch weiterentwickelt.

² Vgl. Leitbild TAVR, www.tavr.at, [26. 05. 2008].

2.3 Forschungsgruppe Maschinenbauinformatik und Virtuelle Produktentwicklung

Die Forschungsgruppe Maschinenbauinformatik und Virtuelle Produktentwicklung ist in das Institut für Konstruktionswissenschaften und Technische Logistik der TU Wien eingebunden. Der Schwerpunkt der Forschung liegt in der Anwendung von rechnergestützten Verfahren im Rahmen der virtuellen Produktentwicklung. Hier geht es um eine möglichst vollständige Beschreibung und Abbildung realer Produkte mit ihren Eigenschaften als Modell im Rechner. So soll eine lückenlose Dokumentation eines Produktes und seines Entstehungsprozesses gewährleistet werden.³

Die Forschungsgruppe ist für die Entwicklung und die Konstruktion der Produkte und Montagemodule verantwortlich. In Rahmen der Diplomarbeit setzt sie die Simulation der Module und des Zusammenbaus der Pneumatikzylinder mit dem Softwaretool COSIMIR um.

2.4 Festo Ges.m.b.H.

Festo gehört zu den führenden Automatisierungsunternehmen weltweit und ist mit seiner Tochter Festo Didactic führend in der industriellen Weiterbildung. Das Unternehmen wurde 1925 gegründet und liefert einen weltweiten Umsatz von 1,65 Milliarden EUR (2006) mit rund 12.800 Mitarbeitern. Festo Didactic bietet Lösungen zur Ausbildung im gesamten Spektrum der Fertigungs- und Prozessautomatisierung und vertreibt die Simulationssoftware COSIMIR.

Festo Österreich und Festo Didactic liefern die gesamten Druckluftkomponenten für den Aufbau der Lehr- und Lernproduktion und die Simulationssoftware COSIMIR Professional.

³ Vgl. <http://www.mivp.tuwien.ac.at> [26. 06. 2008].

3 Lehr- und Lernproduktion

Die TAVR entwickelte eine Lehr- und Lernproduktion (LLP), welche in der HTBL Wien 10 untergebracht wird. Mit Hilfe der LLP können in der Ausbildung stehende Teilnehmer Fertigungs-, Montage- und Kommunikationsprozesse auf Basis modernster Hardware- und Software-Strukturen gestalten und optimieren. Die Lehr- und Lernproduktion ist in Abbildung 1 dargestellt. Sie besteht aus drei Bereichen:

- **Werkstattbereich**
Hier sind die Fertigung, die Qualitätssicherung mit Messtechnik, die Lagersysteme und die physische Kleinserienmontage 1 vorhanden.
- **Animations- und Visualisierungsbereich**
In diesem sind die Arbeitsplätze für die Kleinserienmontage 2 (KSM 2) (Animation COSIMIR), die Vormontage (VM), die Serienmontage 1 und 2 (SM 1 und 2) sowie die der Arbeitsvorbereitung (AV) vorhanden.
- **Bürobereich**
In diesem Raum befinden sich die Arbeitsplätze der Unternehmensleitung (UL), der Auftragsabwicklung (AA) und der Logistik (Log).

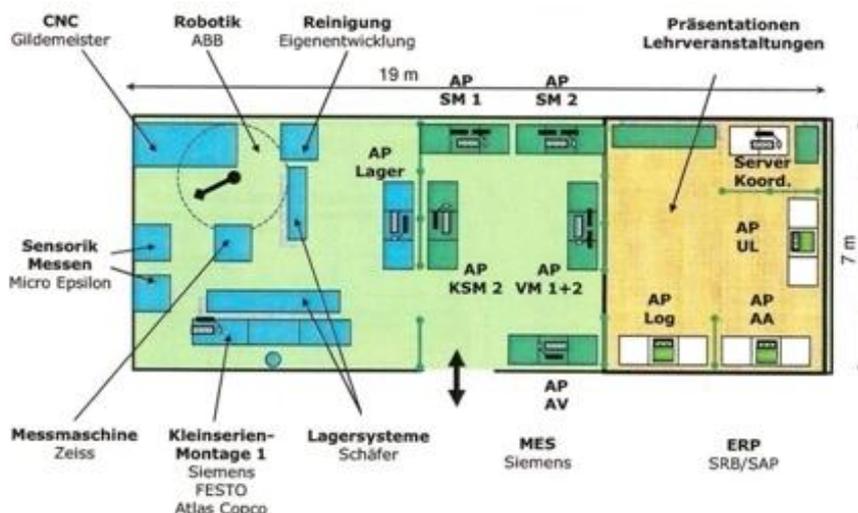


Abbildung 1: Lehr- und Lernproduktion HTBL Wien 10⁴

⁴ Abbildung: o. V.: Partnertag 30.5.2007, technik akademie, S. 3.

Die Kleinserienmontage in der Lehr- und Lernproduktion stellt als Produkt einen Pneumatikzylinder her. Dieses Produkt bietet sich durch die Einfachheit und die vielfältigen Variationsmöglichkeiten an. Es werden insgesamt sechs verschiedene Variationen in drei Hublängen und zwei Kolbenstangendurchmessern zusammengebaut. In der Abbildung 2 sind alle benötigten Lager- und Zukaufsteile dargestellt.

Die Montage erfolgt in vier Montagemodulen. Diese Module und die Montage der verschiedenen Pneumatikzylinder werden in der Simulation dargestellt.

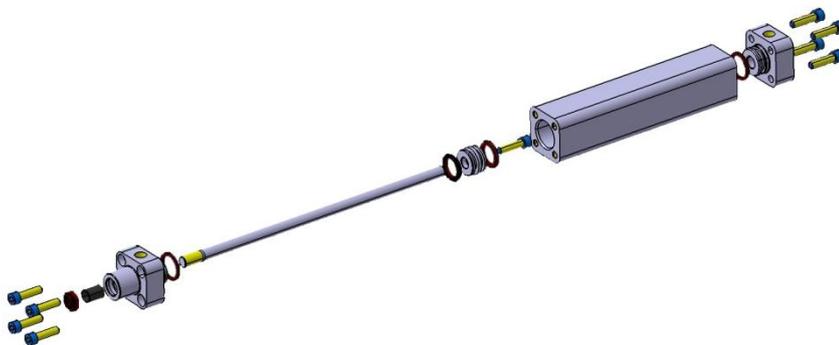


Abbildung 2: Pneumatikzylinder⁵

⁵ Explosionsdarstellung des Pneumatikzylinders.

4 Von der Simulation zum Abbild der Realität

4.1 Begriffsbestimmung

Die Begriffsdefinitionen erfolgen nach der VDI-Richtlinie 3633 Blatt 1 und Blatt 11, da sie dem anerkannten Stand der Technik entsprechen.

Simulation

Simulation ist das Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind.

Virtual Reality

Die Virtual Reality ist eine Mensch-Maschine-Schnittstelle, die es erlaubt, in eine computergenerierte, dreidimensionale virtuelle Welt einzutauchen, diese als Realität wahrzunehmen, Bestandteil dieser zu sein und mit ihr zu interagieren. Als Teil der computergenerierten Welt kann man diese von verschiedenen Positionen aus betrachten und sie unmittelbar verändern.

Visualisierung

Die Visualisierung umfasst die Erzeugung der grafischen Veranschaulichung von Daten und Sachverhalten durch Transformation ermittelter Rohdaten in symbolische und geometrische Information.

4.2 Die Simulation

Die Bedeutung der Simulation nahm in den letzten 10 bis 15 Jahren zu. Dies liegt einerseits an den immer größer werdenden Anforderungen an die Unternehmen in der Wirtschaft und andererseits an der rasanten Entwicklung im Hard- und Softwarebereich.

So führen die Tendenzen in der Wirtschaft, wie

- steigende Anforderungen an die Qualität und an die Flexibilität bei Sonderwünschen,
- steigende Produktkomplexität und Variantenvielfalt,
- schnellere Produktwechsel,
- kleinere Losgrößen und
- steigender Kostendruck

zu immer komplexeren Zusammenhängen und Abläufen in der Entwicklungs- und Planungsphase.⁶

Diese Tendenzen verstärken die Notwendigkeit einer Simulation, da sie es ermöglicht eben diese Zusammenhänge und Abläufe zu analysieren bzw. diese sichtbar zu machen.

Früher wurden Simulationen dafür eingesetzt um Planungen im Nachhinein abzusichern, doch nun wird „innerhalb der Planung und Realisierung von logistischen Systemen (...) die ereignisorientierte Simulation zu einem unabdingbaren Werkzeug mit vielfältigen Aufgaben. Sie wird der Erklärung der Systemfunktion und dem Nachweis der Leistungsfähigkeit ebenso dienen wie der Ausbildung und der Betriebsführung.“⁷

Die Ausbildung anhand einer Simulation kann soweit gehen, dass man die zukünftigen Anlagenbetreiber, wie z. B. Meister oder Kranführer, frühzeitig, also noch vor Bauende der Anlage, an der virtuellen Anlage (siehe Abbildung 3) schulen kann. Die Schulung erfolgt mit Hilfe von Filmsequenzen und interaktiven Szenen. Dies führt zum Erlernen der Bedienung der neuen Anlage in einer anschaulichen und realitätsnahen Form.⁸

⁶ Vgl. VDI 3633 Blatt 1, 2000, S. 2.

⁷ Müller/Hrdliczka, 2003, S. 213.

⁸ Vgl. VDI 3633 Blatt 11, 2003, S. 3.

Die frühzeitige Schulung trägt unter anderem dazu bei, dass eine steilere Anlaufkurve bei neuen Produktionsanlagen erreicht werden kann.

Diesem Ansatz der realitätsnahen Ausbildung folgt auch die Idee der Lehr- und Lernproduktion, für die in dieser Diplomarbeit die Simulation erstellt wird.



Abbildung 3: Ausbildung anhand einer Simulation im virtuellen Raum⁹

Neben der Entwicklung im Hard- und Softwarebereich sind auch noch folgende Punkte¹⁰ für die immer breiter werdende Akzeptanz von Simulationen in Unternehmen verantwortlich:

- Die Simulationssoftware ist anwendungsfreundlicher geworden (schnell zu erlernen und effektiv einzusetzen).
- Das Investitionsrisiko bei Neuentwicklungen von Systemen und Komponenten kann stark verringert werden.
- Die Untersuchung eines Systemverhaltens ist überschaubar und mit vergleichsweise geringem finanziellen Aufwand verbunden und sie kann
- in wesentlich kürzerer Zeit und wiederholt durchgeführt werden, was in der Realität häufig aufwendig ist.
- Parametervariationen sind schneller und einfacher durchzuführen.

⁹ Abbildung: <http://www.virtual-environments.com> [19. 06. 2008].

¹⁰ Vgl. Brychta / Müller, 2004, S. 9.

Wo liegen die Anwendungsfelder der Simulation?

Die Simulation wird heute in allen Phasen des Planungs- und Realisierungsprozesses und zur Prozesssteuerung eingesetzt (siehe Abbildung 4).

In der Planung liegen die Vorteile der Simulation in der Möglichkeit real nicht existierende Systeme zu untersuchen und so z. B. mehrere Gestaltungsvarianten eines Fabrik-Layouts mit geringem Arbeitsaufwand zu erstellen. Unterdessen sind in der Realisierungsphase vor allem Anlaufvorgänge (simulierter Probebetrieb) und Einschwingungsphasen interessant. Auch die oben genannten Schulungssysteme kommen in dieser Phase zur Anwendung.¹¹

Ein weiterer großer Vorteil ist die begleitende Simulation während des Betriebes. So wird es möglich mittels Parameterstudien die operative Planung zu unterstützen oder standortübergreifende Abläufe zu untersuchen.¹² Man kann so ohne direkten Betriebseingriff Abläufe oder Prozesse optimieren.

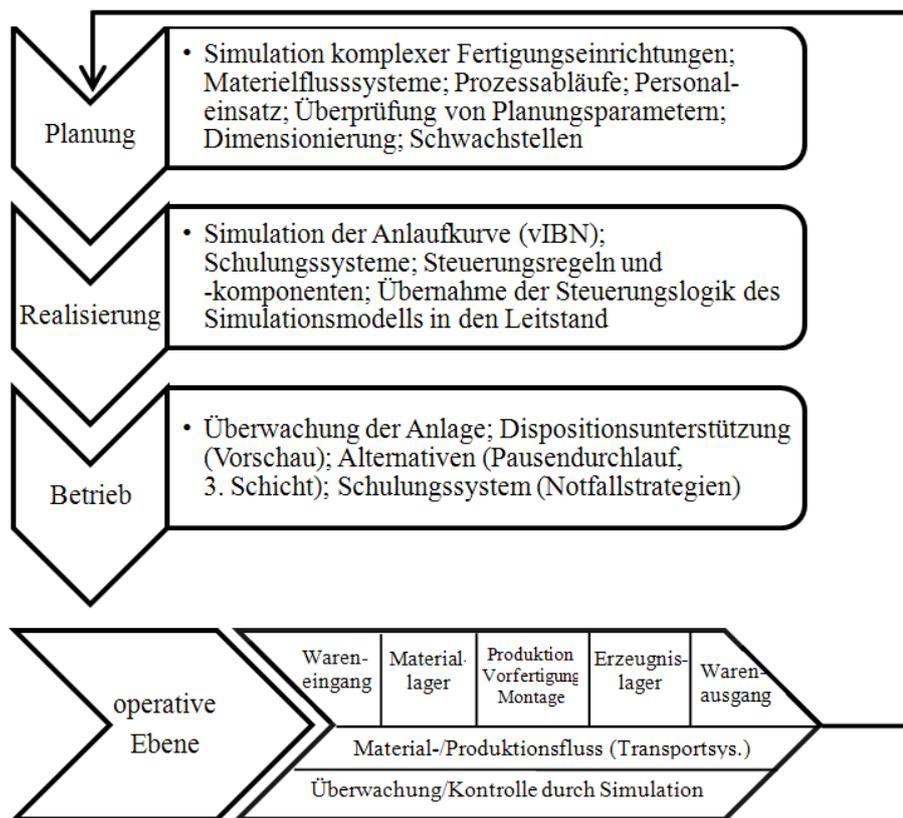


Abbildung 4: Anwendungsfelder der Simulation¹³

¹¹ VDI-Richtlinie 3633, Blatt 1, S. 2 - 4.

¹² Vgl. Spieckermann / Wortmann, 2003, S. 47.

¹³ Vgl. Abbildung: VDI 3633, Blatt 1, 2000, S. 4.

4.3 Digitale Fabrik

Die Digitale Fabrik wird definiert als ein „Oberbegriff für ein umfassendes Netzwerk von digitalen Modellen, Methoden und Werkzeugen – unter anderem der Simulation und 3D-Visualisierung. (...) Ihr Ziel ist die ganzheitliche Planung, Evaluierung und laufende Verbesserung aller wesentlichen Strukturen, Prozesse und Ressourcen der realen Fabrik in Verbindung mit dem Produkt.“¹⁴

Die Digitale Fabrik ist unter anderem zur Unterstützung des Anlaufmanagements notwendig. Es fehlt heute immer noch die vollständige Sicht auf die Elemente der Fabrik. Diese lückenlose Sicht soll sich über die Beschreibung der einzelnen Fertigungsstellen mit

- geometrischer Integration (3D-Layout, Kollisionsprüfung etc.)
- funktionaler Integration (Logistik, Materialfluss, Steuerung) und der
- produkttechnischen Integration des Produkts in die Anlage (Prozesse, Abläufe) erstrecken.

Die Methoden der Digitalen Fabrik vereinen all diese Punkte und tragen so zur Unterstützung des Anlaufmanagements bei.¹⁵

4.3.1 Von der Simulation zur Digitalen Fabrik

Die Fahrzeugindustrie ist im Bereich der Digitalen Fabrik ein Pionier, wie sie es schon damals beim Einsatz von CAD-Software und bei der 3D-Simulation gewesen ist. Sie erkannte frühzeitig das Potenzial und den Bedarf der Digitalen Fabrik.

Der Einsatz der Simulationstechnik ist heute noch stark segmentiert und von einer Durchgängigkeit der Daten weit entfernt. Die Digitale Fabrik wird es schaffen, die vorhandenen Datenmodelle während des gesamten Lebenszyklus, von der ersten Grobauslegung über die Anlaufphase bis hin zur Nutzungsphase der Produktionssysteme, durchgängig zu verwalten.¹⁶

Dies ist auch nötig um im globalen Wettbewerb zu bestehen. Denn in diesem gibt es zwei wesentliche Strömungen mit erheblichen Auswirkungen auf die

¹⁴ VDI-Richtlinie 4499, Blatt 1, S. 3.

¹⁵ Vgl. Breitenbach / Schwab, 2004, S. 2-9.

¹⁶ Vgl. Bayer / Collisi / Wenzel, 2003, S. 3.

Produktion. Wie bereits weiter oben erwähnt, versuchen sich die Hersteller einerseits durch eine große Variantenvielfalt von der Konkurrenz abzuheben und andererseits werden die Innovationszyklen immer kürzer. Dies bedeutet für den Automobilbau einen Produktionszyklus von derzeit vier Jahren. Darum wird das Augenmerk besonders auf den Produktionsanlauf gelegt. Dieser muss verkürzt werden, ohne dabei die Kosten wesentlich zu erhöhen.

Die Abbildung 5 zeigt die Inbetriebnahme im Produktionsanlauf. In dieser Phase hat die Reifegradkurve ihren höchsten Anstieg, das heißt, hier ist der höchste Aufwand im Lebenszyklus erforderlich, da alle Teilsysteme erstmals zu einem Ganzen zusammengefügt werden und anschließend interagieren müssen.¹⁷

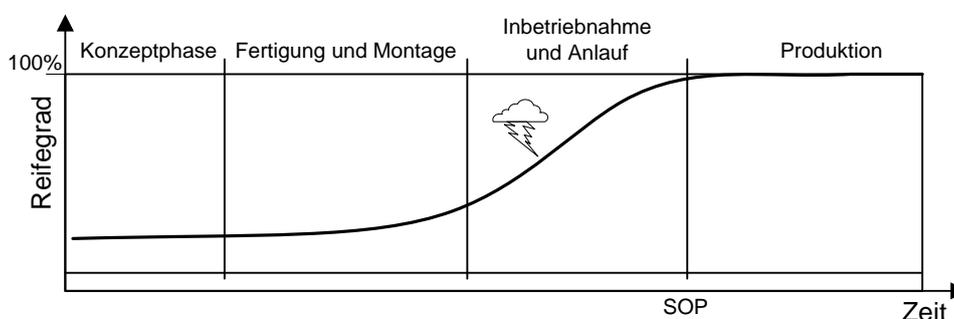


Abbildung 5: Inbetriebnahme im Produktionsanlauf¹⁸

Die Inbetriebnahmekosten liegen im Bereich von 5 bis 20% der Investitionskosten. Daran sieht man, dass diese Phase eine sehr sensible ist, denn oft werden hier die in der Planungs- und Montagephase eingesparten Finanzmittel durch Verzögerungen wieder aufgebraucht.¹⁹ Erfahrungsgemäß entstehen Störungen und Schäden während der Inbetriebnahme von verfahrenstechnischen Anlagen zu über 85% aus Fehlern und Mängeln aus den Vorphasen.²⁰

Bei der Betrachtung der Planungs- und der Inbetriebnahmephase (siehe Abbildung 6) stellt man einen grundsätzlichen Unterschied fest. In der Planung wird nach dem Top-Down-Ansatz vorgegangen. Blickt man auf die Hierarchieebenen, so sieht man, dass während der Planung die Ebenen von oben nach unten (Unternehmensnetzwerk – Fabrik – Produktion – Leitung – Anlagen- bzw.

¹⁷ Vgl. Zäh u.a., 2004, S. 1-2f.

¹⁸ Vgl. Abbildung aus: Zäh u.a., 2004, S. 1-3.

¹⁹ Vgl. Bernecker, 1984, (zit. nach: Weber, Klaus H., 2002, S. 93).

²⁰ Vgl. Backhaus, 1982, (zit. nach: Weber, Klaus H., 2002, S. 6).

Prozessebene) durchschritten werden. Es wird also vom Groben ins Feine detailliert.

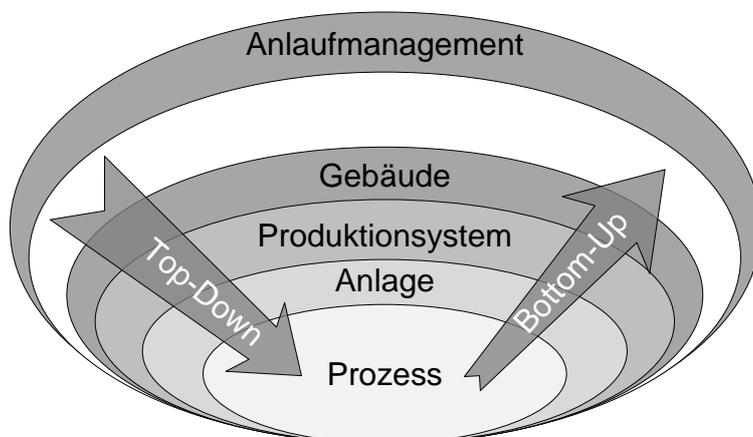


Abbildung 6: Gegensätzliche Vorgehensweise bei Planung und Inbetriebnahme²¹

Bei der Inbetriebnahme jedoch wird die unterste Ebene zuerst in Betrieb genommen und dann stufenweise auch die höher gelagerten Systeme. Der Weg geht also von den einzelnen Anlagenkomponenten über Stationen bzw. Zellen bis hin zu den Ebenen der Leitsysteme.

Um den Projekterfolg feststellen zu können, muss das Projekt zuerst die vollständige Planungsphase für alle Anlagenteile bis ins Detail durchlaufen und in der realen Inbetriebnahme wieder den umgekehrten Weg gehen.²²

4.3.2 Virtuelle Inbetriebnahme – Die Abkürzung

Eine virtuelle Inbetriebnahme (vIBN) der Fabrik kann diesen Weg beschleunigen, da er dazu beiträgt, die reale Inbetriebnahme zu verkürzen. Es werden drei neue Meilensteine (siehe Abbildung 7) hinzugefügt.

- virtueller Start-Of-Production I (vSOP I): Zu diesem Meilenstein stellen alle an der Anlagenentwicklung beteiligten Unternehmen Planungs- und Produktdaten zur Verfügung. Somit steht zum ersten Mal ein ganzheitliches digitales Modell der gesamten Produktion zur Verfügung. Hierbei handelt es sich jedoch um ein reines Geometriemodell, d. h. die zur Verfügung gestellten Daten sind nicht auf ihre Funktionserfüllung hin abgesichert.

²¹ Vgl. Abbildung aus: Breitenbach / Schwab, 2004, S. 2-6.

²² Vgl. Zäh u.a., 2004, S. 1-4.

- virtueller Start-Of-Production II (vSOP II): Bei diesem zweiten Synchronisationspunkt steht ein vollständiges, funktionell abgesichertes digitales Gesamtmodell der Produktion zur Verfügung. Somit können die beteiligten Unternehmen in die Phase der realen Inbetriebnahme übergehen.
- virtuelle Inbetriebnahme (vIBN): Ist die Simulation der Inbetriebnahme unter Verwendung der realen Roboter- und Steuerungsprogramme. Die Phase der vIBN wird zwischen der Konzeptphase und der realen IBN gesetzt. Sie wird begrenzt von den beiden vSOP Meilensteinen und dient der Funktionsabsicherung, d. h. sie bezieht sich auf die Interaktion von Anlage, Bauteil und Prozess.

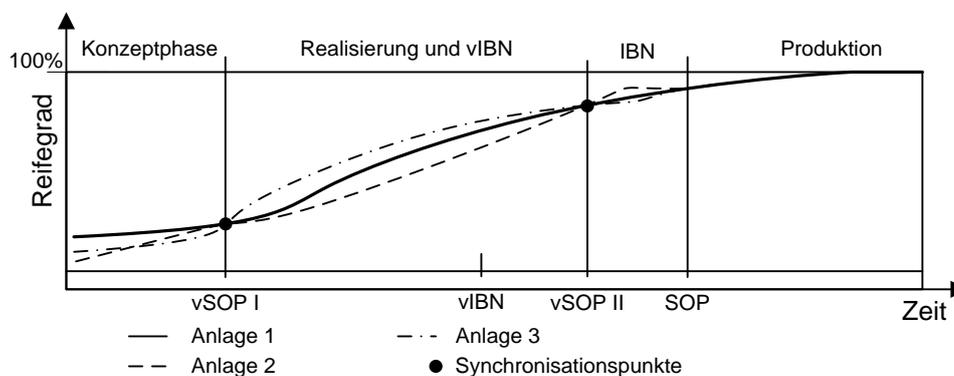


Abbildung 7: Einordnung der neuen Meilensteine ²³

Der vSOP I zwingt die beteiligten Partner (Anlagenlieferanten) sich schon frühzeitig Gedanken über die Anlaufthematik zu machen, da sie zu diesem Zeitpunkt digitale Modelle ihrer Anlagen zur Verfügung stellen müssen.

Bei der vIBN liegt der Fokus auf der Funktionalität, der Verkettung und dem Automatikbetrieb der Anlage im Produktionsumfeld. Dem Anlagenlieferanten ist es so möglich, schon im Vorfeld viele Fehler (Steuerung, Verkettung) zu erkennen, die er sonst erst bei der realen IBN erkannt hätte. Somit werden die Fehler schon vor der Anlieferung behoben und die IBN erfolgt reibungsloser und dadurch schneller.²⁴

Diese schnellere IBN ist im globalisierten und turbulenten Umfeld der Produktion auch nötig um am Markt bestehen zu können.

²³ Vgl. Abbildung aus: Breitenbach / Schwab, 2004, S. 2-10.

²⁴ Vgl. Breitenbach / Schwab, 2004, S. 2-11 ff.

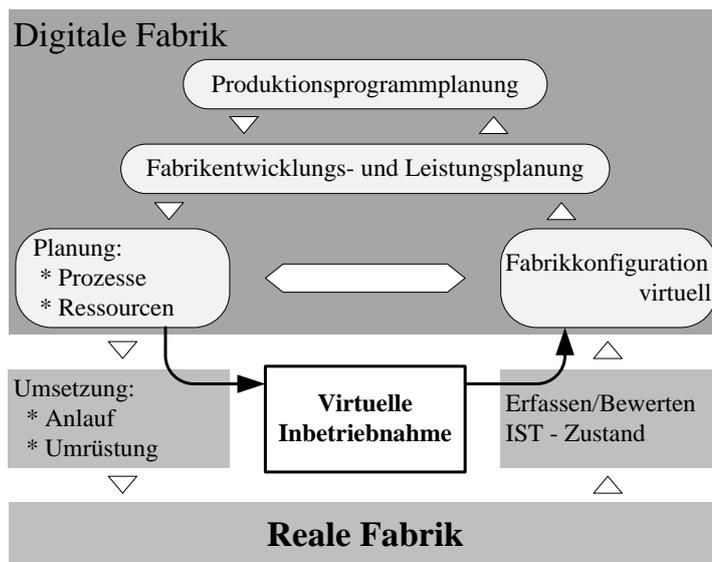


Abbildung 8: Fabrikplanungszyklus ²⁵

Die Digitale Fabrik und die vIBN lassen sich auch als Regelkreis darstellen. In Abbildung 8 kann man erkennen, dass durch die Kombination der beiden ein Fabrikplanungszyklus entsteht. So lassen sich Erkenntnisse aus der Inbetriebnahme im Labor vorwegnehmen und man kann diese über einen schnelleren Regelkreis der Planung zuführen. Ohne die vIBN geht der Weg über die reale Fabrik und es kommt dadurch zu sehr langen Zykluszeiten.²⁶

Durch die Planung einer Fertigungsanlage mit Hilfe der Digitalen Fabrik können Aufwand und Zeit bis zum SOP um bis zu 50% gesenkt werden.

²⁵ Vgl. Abbildung aus: Zäh u.a., 2004, S. 1-8.

²⁶ Vgl. Zäh u.a., 2004, S. 1-8.

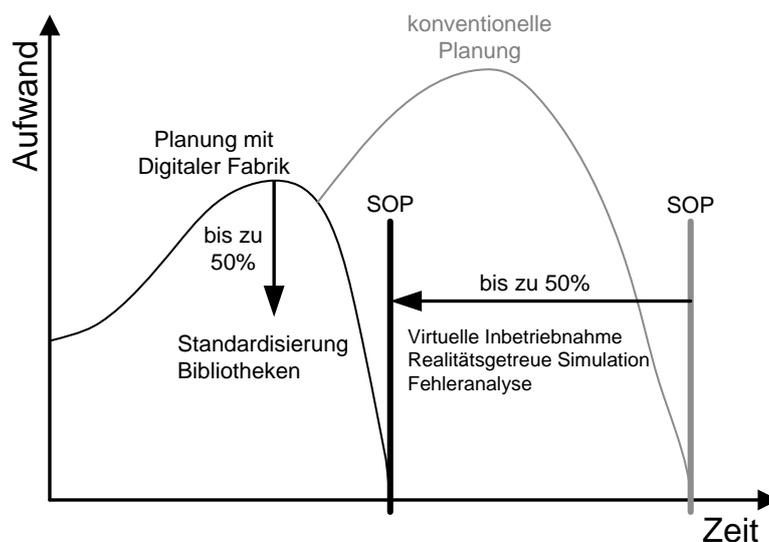


Abbildung 9: Aufwand und Produktionsstart ²⁷

Mit der Simulationssoftware COSIMIR ist die Erstellung eines digitalen Modells einer Produktionsanlage ausführbar. Es ist ein durchgängiger Arbeitsfluss vom Import der CAD-Modelle bis hin zur virtuellen Inbetriebnahme möglich. Dieser Weg erstreckt sich von der 3D-Visualisierung, der Materialfluss-/Ablaufsimulation, der Prozess- und Robotersimulation bis hin zur Offline-Programmierung der Roboter und der Erstellung lauffähiger SPS- und Roboterprogramme. So kann eine schrittweise Inbetriebnahme der Hardware erfolgen. Wie in Abbildung 9 ersichtlich führt dies zu einer deutlichen Beschleunigung des Planungsprozesses und Reduzierung des Planungsaufwandes bei gleichzeitiger Verbesserung der Planungsergebnisse.

²⁷ Vgl. Abbildung aus: Heinze, Frank: Durchgängige Virtuelle Inbetriebnahme von Fertigungsanlagen, 2008, Online im WWW unter: www.berufsbildungstage.de/download/Vortrag_Frank_Heinze.pdf [19. 07. 2008].

5 COSIMIR Professional

COSIMIR (Cell Oriented Simulation of Industrial Robots) ist eine PC-basierte 3D-Simulationssoftware der EF-Robotertechnik GmbH in Schwerte, mit der es möglich ist

- eine robotergestützte Arbeitszelle zu planen,
- die Erreichbarkeit aller Positionen zu überprüfen und
- die Roboter- und Steuerungsprogramme in den originalen Programmiersprachen zu entwickeln.

Die grafische Darstellung der Arbeitszelle reicht von der einfachen Drahtrahmendarstellung bis hin zur schattierten Darstellung unter Berücksichtigung mehrerer Lichtquellen.

Das Hauptfenster von COSIMIR ist wie in Abbildung 10 folgendermaßen unterteilt:

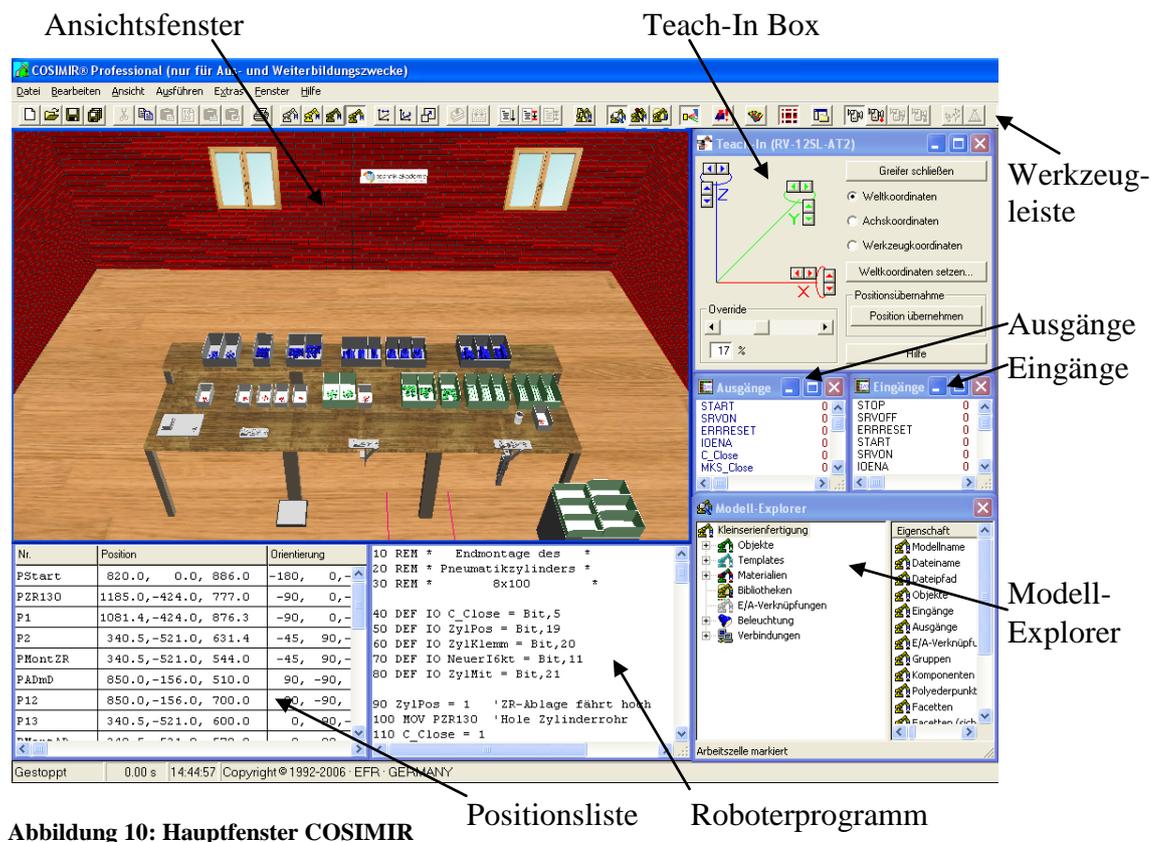


Abbildung 10: Hauptfenster COSIMIR

- Ansichtsfenster: Zeigt die 3D-Darstellung der geladenen Arbeitszelle.
- Teach-In Box: Mittels der Teach-In Box wird der Roboter in den Achsrichtungen manuell bewegt. So werden die verschiedenen Positionen angefahren und diese kann man in der Positionsliste speichern.
- Ein-/Ausgänge: Sie zeigen an, welche Signale an den Ein-/Ausgängen der simulierten Steuerung anliegen.
- Modell-Explorer: Mittels des Modell-Explorers kann auf alle Elemente und deren Eigenschaften zugegriffen werden.
- Roboterprogramm: Das Programm besteht aus einer Folge von Deklarationen und Anweisungen.
- Positionsliste: Jedes Programm hat seine eigene Positionsliste. Die Zuordnung erfolgt über den Namen der Liste, der gleich dem Programmnamen sein muss. Im Programm selbst können die einzelnen Positionen über ihre Nummer (=Namen) aufgerufen werden.

In der folgenden Liste ist ersichtlich, welche Programmiersprachen durch COSIMIR Professional unterstützt werden:

- IRL (DIN 66312)
- KRL (Kuka)
- Rapid (ABB)
- V+ (Adept und Stäubli)
- Movemaster Command (Mitsubishi)
- Melfa Basic III und IV (Mitsubishi)

Dank der reichhaltigen Bibliothek ist es so möglich alle gängigen Robotersysteme der oben genannten Firmen in die Arbeitszelle einzubauen und zu steuern.

In der modellierten Arbeitszelle lassen sich alle Bewegungsabläufe und Handhabungsvorgänge simulieren, damit ist eine Optimierung der Erreichbarkeit und der Zykluszeit möglich. So können das Layout und die Roboterbahn optimiert werden und es kommt zu einer effizient gestalteten Anlage. Weiters ist die Taktzeitanalyse für Steuerungsprogramme und die Durchsatzanalyse für Produktionslinien möglich.

Offline-Programmierung von Robotern

Bei der Offline-Programmierung wird das Programm unabhängig vom realen Roboter erstellt, das heißt, der Roboter kann weiter betrieben werden und es gibt so keine Stillstandzeiten. Bei der Offline-Programmierung mit COSIMIR wird der gesamte Bewegungsablauf in der Simulationsumgebung festgelegt. Dazu müssen allerdings die Umgebung und die Werkzeuge des Roboters in die Simulation importiert werden, was einen zusätzlichen Aufwand darstellt.

Das in der Simulation erstellte Programm wird per Datenträger oder Netzwerk zum Industrieroboter übertragen und kann sofort ausgeführt werden. Meist sind aber noch einige Kalibrierungen erforderlich, da die Simulation nie genau mit der Realität übereinstimmt.

Die Vorteile der Offline-Programmierung sind:

- Reduzierung der Stillstandzeiten
- Planungs- und Konstruktionsfehler können frühzeitig erkannt werden
- Vereinfachung der Prozessoptimierung und somit eine
- Erhöhung der Produktivität
- Überprüfung der Erreichbarkeit der Arbeitspunkte
- Frühzeitige Kollisionserkennung
- Ermittlung von Zyklus- und Taktzeiten

Demgegenüber stehen die Nachteile der Offline-Programmierung:

- Die Umgebung existiert oft nicht als 3D-Datenmodell und muss erst erstellt werden.
- Flexible Versorgungsleitungen können am Computer nur unzureichend abgebildet werden, bewirken aber eine starke Bewegungseinschränkung des realen Industrieroboters.
- Die in der 3D-Simulation interpolierten Bahnen stimmen nicht zu 100% mit den in der Steuerung berechneten Bahnen überein.²⁸

²⁸ Vgl. <http://de.wikipedia.org/wiki/Industrieroboter> [20. 07. 2008].

6 Beschreibung der Arbeitszelle

Die Modellierung aller Module, Behälter, Arbeitstische und der Zylinderbestandteile erfolgte mit der CAD-Konstruktionssoftware CATIA V5.

6.1 Einlesen der Daten in COSIMIR

Es ist viel manuelle Arbeit nötig um eine statische 3D-Szene zu modellieren. Für eine möglichst realistische Darstellung müssen Geometriedaten zu Gebäuden und Anlagen sowie weiteren Einrichtungsgegenständen (z. B. Regale, Container) beschafft werden. Kann man auf vorhandene 3D-Geometrien aus der Anlagen- und Layoutplanung zurückgreifen, so darf man den Aufwand zur Konvertierung nicht unterschätzen. So zeigte sich nach den Erfahrungen der beiden Autoren Kremer und Liewald, dass es manchmal effektiver ist, Geometrien in vereinfachter Form nachzumodellieren, als mit viel Aufwand CAD-Daten in eine geeignete Form zu bringen.²⁹

Da die Zylindermodelle in CATIA konstruiert wurden, lagen diese Daten bereit. Aufgrund einer fehlenden direkten Schnittstelle zwischen CATIA und COSIMIR musste ein Umweg gefunden werden. Dieser bestand in der Umwandlung der CATIA Files in das STEP Format. Die STEP Files wurden in OneSpaceDesigner eingelesen, welcher über eine gemeinsame Schnittstelle mit COSIMIR verfügt. Diese gemeinsame Schnittstelle ist das VRML-Format. Die Virtual Reality Modelling Language (VRML) ist eine Beschreibungssprache für 3D-Geometrien. Da die meisten 3D-Modellierungswerkzeuge den Im- und Export von VRML-Dateien ermöglichen, etablierte sich das Dateiformat als Austauschformat von 3D-Modellen.³⁰

Die Grenzen die durch so genannte Insellösungen der virtuellen Welt, d. h. es gibt zwischen manchen Produkten keine entsprechende Schnittstelle, bestehen, müssen in Zukunft aufgelöst werden. Da in der Überbrückung der Schnittstellen viel Zeit, Aufwand und Informationsverlust steckt, kann man sich nicht mehr nur auf die

²⁹ Vgl. Kremer/Liewald, 2003, S. 190.

³⁰ Vgl. <http://de.wikipedia.org/wiki/VRML> [25. 06. 2008].

Prozesse konzentrieren. Daher muss es dem Anwender künftig möglich sein, in einer kompatiblen Welt seine Daten leichter austauschen zu können.³¹

Nachdem alle Daten erfolgreich in COSIMIR implementiert wurden, bekamen die Montagevorrichtungen ihre steuerbaren Mechanismen (siehe dazu Kapitel 6.2 und Kapitel 6.3).

6.2 Montagevorrichtungen

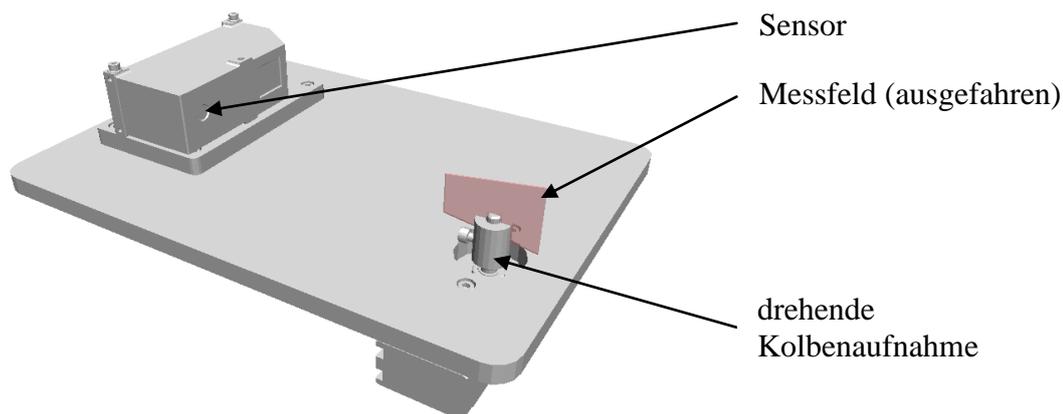
Wie auch in der realen LLP werden die Montagevorrichtungen in der Simulation mit Zweiwegeschubzylindern ausgestattet um die Klemm- und Positioniervorgänge zu simulieren. Nur das Montagemodul Kolben besitzt in der Simulation einen zusätzlichen Zweiwegeschubzylinder um das Messfeld zu visualisieren.

6.2.1 Montagemodul Kolben

Das Montagemodul Kolben (MMK) dient zur Montage der beiden Dichtungen auf den Kolben. Es befindet sich zusätzlich ein Sensor³² auf der Vorrichtung, der es ermöglicht die richtige Montage der Dichtungen zu kontrollieren. Das MMK (siehe Abbildung 11) wurde mit einem Drehzylinder versehen, um den Kolben während der Messung durch den Sensor drehen zu können. Die Prüfung wird mittels eines dargestellten Messfeldes visualisiert. Der Kolben führt eine ganze Umdrehung im Messfeld aus, wobei das Messfeld vor der Drehung aktiviert, d. h. durch den Zweiwegeschubzylinder ausgefahren und nach der Umdrehung des Kolbens um die eigene Achse wieder deaktiviert (= eingefahren) wird.

³¹ Vgl. Griesbach u.a., 2004, S. 4-26.

³² Laser-Linienscanner scanCONTROL 2800 der Firma MICRO-EPSILON.

Abbildung 11: Montagemodul Kolben³³

Die Ansteuerung des Drehkolbens erfolgt über den Ausgang *KolbenDreh* [16]. Das Messfeld wird über den Ausgang *Messfeld* [15] aktiviert bzw. deaktiviert, da der Ausgang auch invertiert mit dem Eingang *MessfeldAus* verbunden ist.

Objekt	Ein-/Ausgang	Name	Index	Beschreibung	Verknüpft mit
RV-12SL	Eingang	MessungAn	15	Zustand	
		MessungAus	16	Messung	
		KolbenDrehAus	17	Zustand Drehkolben	
	Ausgang	Messfeld	15	Steuersignal Messfeld	MessfeldAn !MessfeldAus
		KolbenDreh	16	Steuersignal Drehkolben	StartBewegung
MMK	Eingang	MessfeldAn		Steuerung des	
		MessfeldAus		Messfelds	
		StartBewegung		Steuerung Drehkolben	
	Ausgang	MessungAktiv		Zustand der	MessungAn
		MessungAus		Messung	MessungAus
		EndPosErreicht		Zustand des Drehkolbens	KolbenDrehAus

Tabelle 1: Ein-/Ausgänge Montagemodul Kolben

Die graphische Darstellung der Montageschritte für das Montagemodul Kolben ist in Abbildung 12 ersichtlich.

³³ Darstellung ohne Kolben.

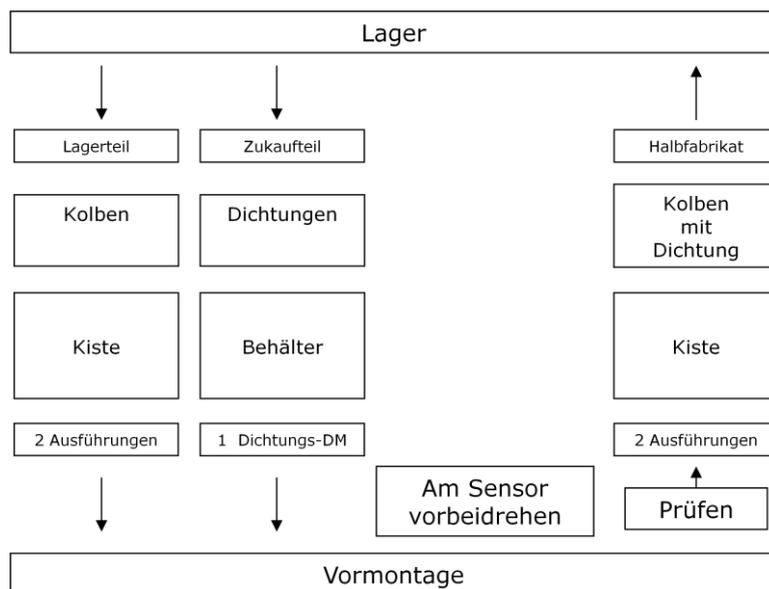


Abbildung 12: Montageschritte Kolben

Hier werden die einzelnen Montageschritte abgebildet. Der Lagerteil Kolben wird der Kiste entnommen und auf der Kolbenaufnahme befestigt. Die Zusatzbemerkung weist darauf hin, dass es zwei verschiedene Ausführungen des Kolbens gibt. Weiters werden zwei Dichtungen, welche Zukaufteile sind, aus dem jeweiligen Behälter herausgenommen und auf dem Kolben montiert. Die Prüfung der richtigen Montage der Dichtungen auf dem Kolben erfolgt über den Sensor, das heißt der Sensor tastet den Umfang des Kolbens mit einem Messfeld ab. Anschließend wird das fertig montierte Halbfabrikat Kolben mit Dichtung in das Zwischenlager gelegt.

Alle folgenden Abbildungen der Montageschritte folgen demselben Prinzip.

6.2.2 Montagemodul Deckel

Auf dieser Vorrichtung (siehe Abbildung 13) werden der Abschlussdeckel und der Lagerdeckel montiert. Am Abschlussdeckel wird eine Dichtung und am Lagerdeckel eine Dichtung sowie eine Gleithülse eingebaut. Dazu muss der Lagerdeckel einmal gewendet werden.

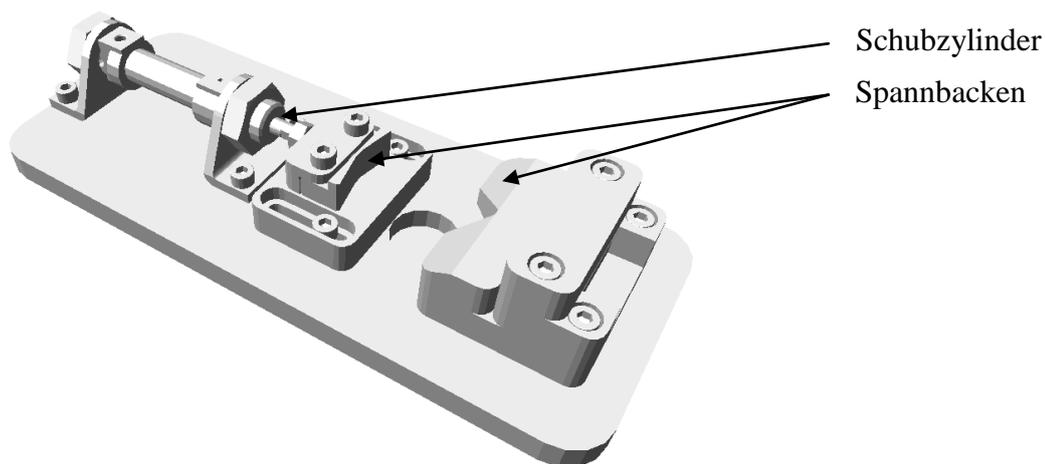


Abbildung 13: Montagemodul Deckel

Der Zweiwegeschubzylinder simuliert hier die Auf- und Zubewegung der Klemmvorrichtung. Die Hublänge wurde so eingestellt, dass die Backen den Deckel optisch berühren. Angesteuert wird der Schubzylinder über den Ausgang *DeckelKlemm* [17].

Objekt	Ein-/Ausgang	Name	Index	Beschreibung	Verknüpft mit
RV-12SL	Ausgang	DeckelKlemm	17	Steuersignal Schubzylinder	Ausfahren !Einfahren
MMD	Eingang	Ausfahren		Steuerung des Schubzylinders	
		Einfahren			

Tabelle 2: Ein-/Ausgänge Montagemodul Deckel

Abbildung 14 zeigt die Montageschritte des Abschlussdeckels und in Abbildung 15 sind die Montageschritte des Lagerdeckels ersichtlich. Die beiden Halbfabrikate werden nach der Montage einer optischen Prüfung unterzogen und in den jeweiligen Zwischenlagerbehälter gelegt.

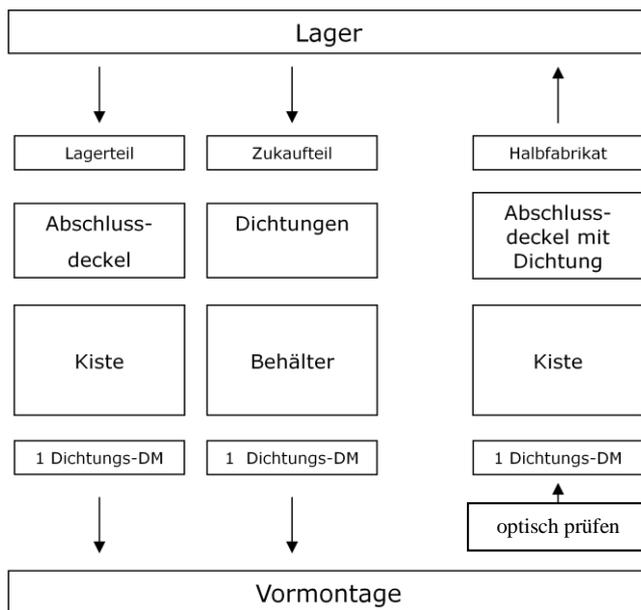


Abbildung 14: Montageschritte Abschlussdeckel

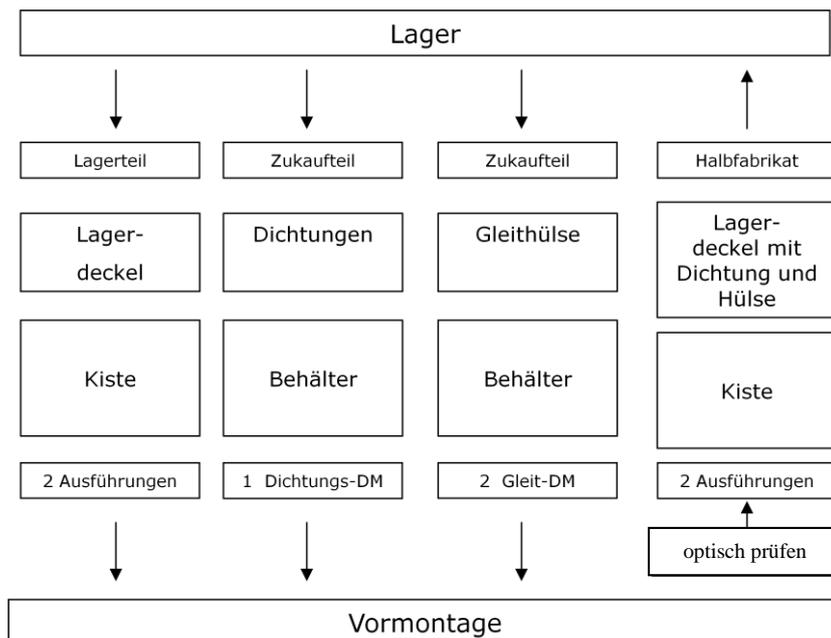


Abbildung 15: Montageschritte Lagerdeckel

6.2.3 Montagemodul Kolbenstange

Das Montagemodul Kolbenstange (siehe Abbildung 16) dient dem Zusammenbau der Kolbenstange mit dem Halbfabrikat Kolben mit Dichtung. Die Höhenverstellung der Auflagenfläche für die Kolbenstange erfolgt bei diesem Modul manuell.

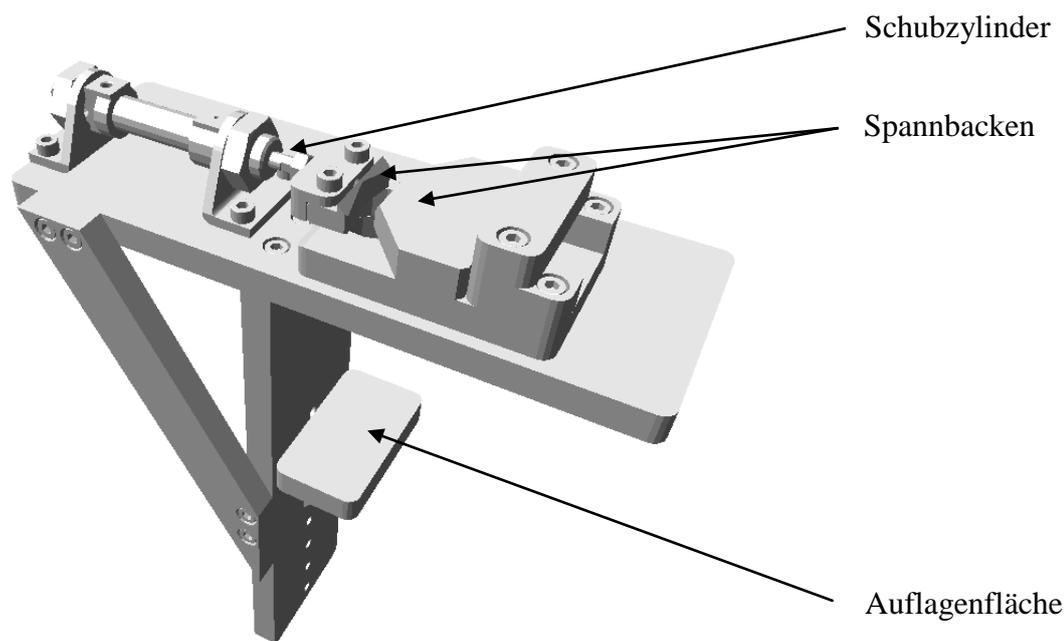


Abbildung 16: Montagemodul Kolbenstange

Auch hier simuliert der Zweiwegeschubzylinder die Auf- und Zubewegung der Klemmvorrichtung. Angesteuert wird er über den Ausgang *KSKlemm* [18].

Objekt	Ein-/Ausgang	Name	Index	Beschreibung	Verknüpft mit
RV-12SL	Ausgang	KSKlemm	18	Steuersignal Schubzylinder	Ausfahren !Einfahren
MM-KS	Eingang	Ausfahren		Steuerung des Schubzylinders	
		Einfahren			

Tabelle 3: Ein-/Ausgänge Montagemodul Kolbenstange

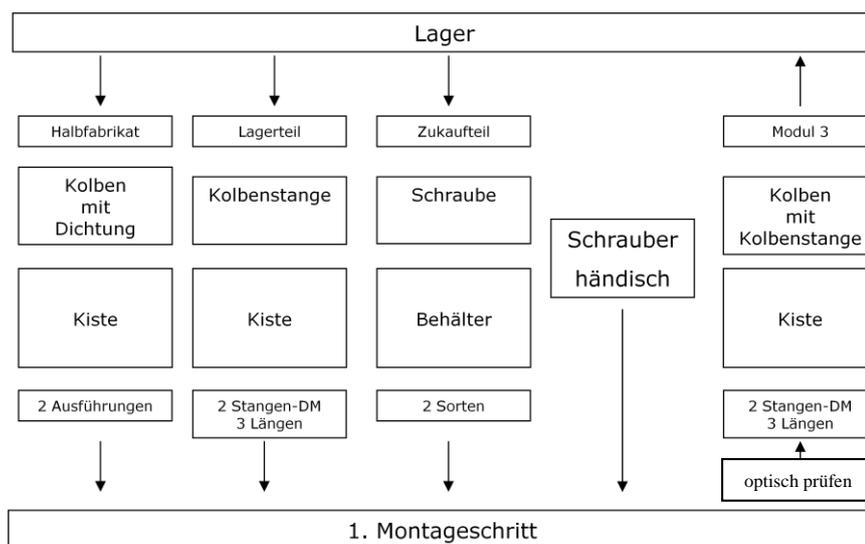


Abbildung 17: Montageschritte Kolbenstange

6.2.4 Montagemodul Zylinder

Diese Vorrichtung (siehe Abbildung 18) dient der Endmontage des Pneumatikzylinders. Alle in den vorherigen Schritten gefertigten Halbfabrikate werden am Zylinderrohr zusammengefügt.

Das Modul besitzt drei ansteuerbare Mechanismen. Diese wären der Zweiwegeschubzylinder für die Auf-/Zubewegung der Spannbacken, der Positionierer für die Höhenverstellung der Auflagenfläche und der Sauggreifer, der im Bereich der Auflagenfläche installiert wurde um eine Mitnahme des Zylinderrohres mit dem Positionierer zu ermöglichen. In der realen Vorrichtung wird dieser Sauggreifer nicht benötigt, da seine Aufgabe die Schwerkraft übernimmt.

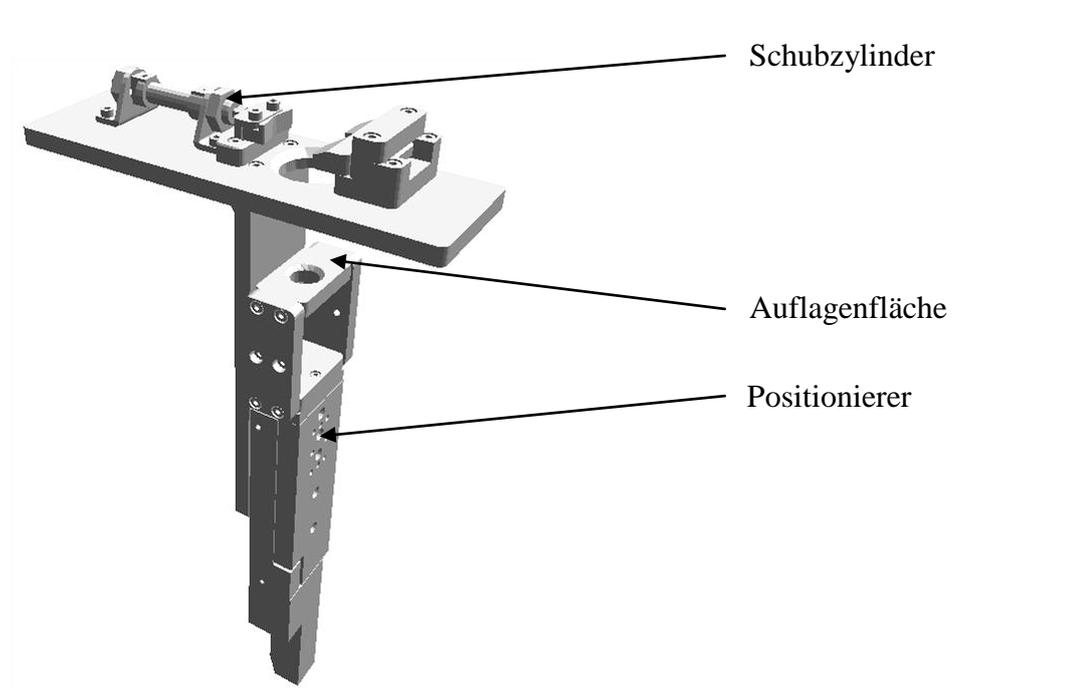


Abbildung 18: Montagemodul Zylinder

Die Ansteuerung der Mechanismen sieht wie folgt aus: Der Ausgang *ZylPos* [19] wird zuerst aktiviert, damit die Auflagenfläche nach oben fährt, dann wird das Zylinderrohr in die Vorrichtung gestellt, nun wird *ZylMit* [21] aktiviert um das Zylinderrohr mit nach unten nehmen zu können. Wenn der Positionierer seine Endposition erreicht hat, wird der Klemmvorgang mit *ZylKlemm* [20] gestartet.

Objekt	Ein-/Ausgang	Name	Index	Beschreibung	Verknüpft mit
RV-12SL	Eingang	ZylOben	18	Zustand	
		ZylUnten	19	Positionierer	

	Ausgang	ZylPos	19	Steuersignal Positionierer	Hochfahren !Einfahren.Pos
		ZylKlemm	20	Steuersignal Schubzylinder	Ausfahren !Einfahren.Klemm
		ZylMit	21	Steuersignal Sauggreifer	Saugen
MM-Zyl	Eingang	Hochfahren		Steuerung des Positionierers	
		Einfahren.Pos			
		Ausfahren		Steuerung des Schubzylinders	
		Einfahren.Klemm			
	Ausgang	Hochgefahren		Zustand des Positionierers	ZylOben
		Eingefahren			ZylUnten

Tabelle 4: Ein-/Ausgänge Montagemodul Zylinder

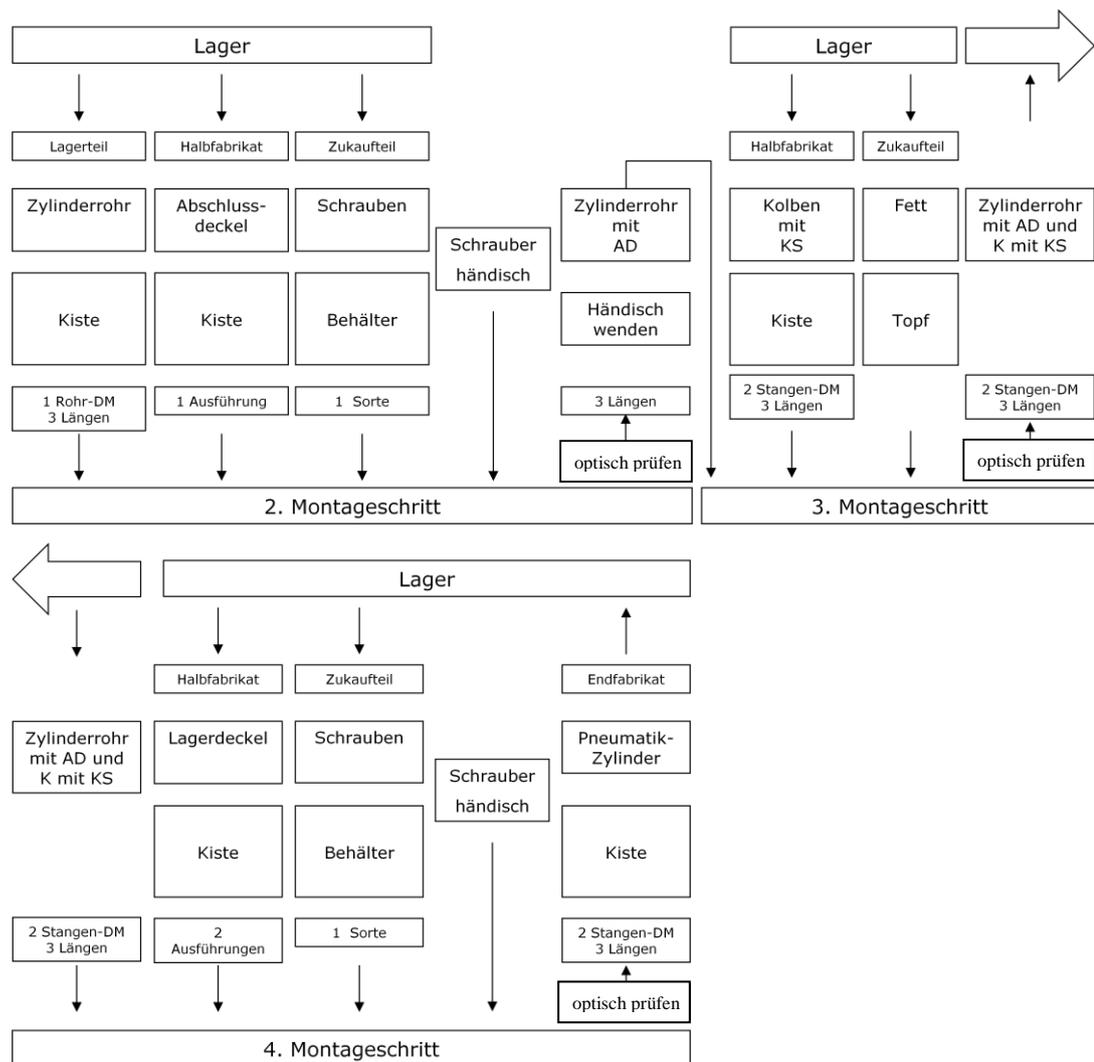


Abbildung 19: Montageschritte Pneumatikzylinder

6.3 Modellierung steuerbarer Mechanismen

Um die dynamischen Prozesse in der Arbeitszelle zu simulieren bedarf es ansteuerbarer Mechanismen. So wird einem Objekt über den Objekttyp ein Mechanismus zugewiesen. Die Objektstruktur (Anzahl der Ein-/Ausgänge, Konfiguration der Gruppen) ist je nach Art des Mechanismus vorgegeben. Diese muss für eine einwandfreie Simulation eingehalten werden.

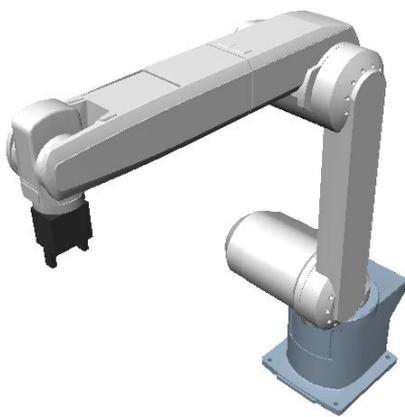
Die nachfolgenden Mechanismen wurden in der vorliegenden Diplomarbeit verwendet und die Beschreibung dieser Mechanismen erfolgte anhand des Hilfemenüs von COSIMIR.

6.3.1 Roboter

In der Simulation wird ein Roboter für den Zusammenbau des Pneumatikzylinders eingesetzt. In der realen LLP übernehmen diese Aufgaben die Lernenden.

Der in der Simulation verwendete Roboter ist von der Firma Mitsubishi Electric und besitzt die Typenbezeichnung RV-12SL. Er ist durch seine programmierbaren Ausgänge die Steuereinheit aller im Anschluss beschriebenen ansteuerbaren Mechanismen.

Die Technischen Daten des Roboters sind laut Hersteller:



Konstruktion:	Knickarm
Freiheitsgrade:	6 Achsen (ohne Hand)
Elektrischer Antrieb:	AC - Servomotoren
Reichweite:	1385 mm
Geschwindigkeit:	9500 mm/s
Hebekraft:	12,0 kg incl. Hand
Wiederholgenauigkeit:	±0,05 mm
Installation:	horizontal, vertikal, hängend
Umgebungstemperatur:	0°C - 40°C
Gewicht:	98 kg

Abbildung 20: Mitsubishi Roboter

Der Roboter ist in Abbildung 20 mit dem montierten Greifer (siehe nächste Seite) dargestellt.

6.3.2 Greifer

Der Greifer wird auf den Roboterarm gesetzt und mit der Robotersteuerung verbunden. Er ersetzt in der Simulation der LLP die Hände einer realen Person, die den Zylinder fertigt. Durch den Greifer wird es möglich die vorhandenen Objekte zu bewegen. Den Objekten muss dazu ein Greifpunkt eingefügt werden, den der Greifer greifen kann.

Mechanismus				
Greifer				
System-Ein-/Ausgänge				
Eingänge	Index	Typ	Wert	Beschreibung
		000	digital	1
	0			Öffnet den Greifer.
Ausgänge	Index	Typ	Wert	Beschreibung
	000	digital	0	Greifer ist nicht geschlossen.
1			Greifer ist geschlossen.	

Tabelle 5: Greifer

6.3.3 Zweiwegeschubzylinder

Die Ansteuerung des Zweiwegeschubzylinders erfolgt über eine invertierte Schaltung der beiden Eingänge. Der Zylinder fährt eine ihm definierte Hublänge aus und nimmt alle ihm zugewiesenen Objekte, wie z. B. Spannbacken, mit. Die zwei System-Ausgänge geben den aktuellen Zustand des Zweiwegeschubzylinders wieder.

Anwendung findet dieser Mechanismus in allen Montagevorrichtungen.

Mechanismus				
Zweiwegeschubzylinder				
System-Ein-/Ausgänge				
Eingänge	000	digital	1	Ausfahren des Schubzylinders.
			0	Kein Ausfahren des Schubzylinders.
	001	digital	1	Einfahren des Schubzylinders.
			0	Kein Einfahren des Schubzylinders.
Ausgänge	000	digital	0	Zylinder ist nicht ausgefahren.
			1	Zylinder ist ausgefahren.
	001	digital	0	Zylinder ist nicht eingefahren.
			1	Zylinder ist eingefahren.

Tabelle 6: Zweiwegeschubzylinder

6.3.4 Drehzylinder

Der Drehzylinder ist ähnlich einem Schubzylinder, nur wird hier die translatorische Bewegung in eine rotatorische Bewegung umgesetzt.

Zur Anwendung kommt der Mechanismus im Montagemodul Kolben, damit der gesamte Umfang des Kolbens durch das Messfeld gescannt werden kann.

Mechanismus				
Drehzylinder				
System-Ein-/Ausgänge				
Eingänge	Index	Typ	Wert	Beschreibung
	000	digital	1	Drehzylinder dreht in die Endposition.
0			Drehzylinder dreht in die Anfangsposition.	
Ausgänge	Index	Typ	Wert	Beschreibung
	000	digital	0	Drehzylinder ist nicht in Endposition.
1			Drehzylinder ist in Endposition.	

Tabelle 7: Drehzylinder

6.3.5 Replikator

Der Replikator dient der gezielten Erzeugung von neuen Objekten auf Basis von Templates. Die Zuordnung der Templates zu den System-Eingängen erfolgt in den erweiterten Eigenschaften des Replikator-Objekts (Beispiel: template0= „I6kt“). Durch das Setzen des System-Eingangs auf 1, wird ein neues Objekt, basierend auf dem Template, an der Position des Greifpunkts des Replikator-Objektes erzeugt.

Anwendung findet der Replikator in der wiederholten Erzeugung der Schrauben für die Endmontage des Pneumatikzylinders und der beiden Nutringe für den Kolben.

Der Mechanismus Papierkorb ist das Gegenteil eines Replikators, er lässt Objekte aus der Arbeitszelle verschwinden. Dieser Mechanismus findet in der Diplomarbeit keine Anwendung, sei aber der Vollständigkeit halber erwähnt.

Die System-Ein- und Ausgänge des Replikators sind in der Tabelle 8 aufgelistet.

Mechanismus				
Replikator				
System-Ein-/Ausgänge				
Eingänge	Index	Typ	Wert	Beschreibung
000	digital		1	Erzeugt erstes konfiguriertes Objekt.
			0	-
001	digital		1	Erzeugt zweites konfiguriertes Objekt.
			0	-
...
00n	digital		1	Erzeugt n-tes konfiguriertes Objekt.
			0	-

Tabelle 8: Replikator

6.3.6 Fließband

Das Fließband befördert ein Objekt, das einen freien Greifpunkt im Greifbereich der aktiven Fläche des Fließbandes besitzt. Dieses Objekt wird mit der konfigurierten Transportgeschwindigkeit entlang des Fließbandes bewegt, wenn der Eingang 000 auf 1 gesetzt ist. Der Ausgang zeigt an, ob ein Objekt das Fließbandende erreicht hat.

Angewendet wird das Fließband zur Reichweitenerhöhung des Roboters. Genauer wird dies in Kapitel 6.4 beschrieben.

Mechanismus				
Fließband				
System-Ein-/Ausgänge				
Eingänge	Index	Typ	Wert	Beschreibung
000	digital		1	Schaltet das Fließband ein.
			0	Schaltet das Fließband aus.
001	digital		1	Fließband fährt rückwärts.
			0	Fließband fährt vorwärts.
Ausgänge	Index	Typ	Wert	Beschreibung
000	digital		0	Es ist kein Teil am Fließbandende.
			1	Es ist ein Teil am Fließbandende.

Tabelle 9: Fließband

6.4 Simulationsumgebung

Die Darstellung und die Position der Behälter, der Arbeitstische und der Montagevorrichtungen wurde mit Herrn Prof. Grafinger abgesprochen. Sie sollten der realen Umgebung der Werkstätte der HTBL Wien 10 nahe kommen um eine möglichst realistische Simulation erstellen zu können.

Da der Zusammenbau der Pneumatikzylinder in der Schule durch die Schüler erfolgt und nicht wie in der Simulation durch einen Roboter, muss dieser unsichtbar dargestellt werden. Ansonsten würde der Wiedererkennungseffekt sinken.

Die aus CATIA übernommenen Einzelteile werden in drei Gruppen eingeteilt, die farblich in den Behältern wie folgt dargestellt werden:

- Lagerteile → blau
Kolben, Abschlussdeckel, Lagerdeckel, Kolbenstange, Zylinderrohr

- Zukaufteile → rot
Nutting, Dichtungen, O-Ringe, Gleithülsen, Schrauben

- Halbfabrikate → grün
Kolben, Lager-, Abschlussdeckel (alle mit Dichtung), Kolbenstange mit Kolben

Die begrenzte Reichweite des Roboters wurde durch ein Förderband gelöst, auf das der Roboter gestellt wurde. Das Förderband besitzt drei „Haltestellen“, von denen der Roboter den Zusammenbau der einzelnen Halbfabrikate und zuletzt des gesamten pneumatischen Zylinders bewerkstelligen kann (siehe Abbildung 21).

In der Anfangsposition werden der Kolben und der Abschlussdeckel mit den jeweiligen Dichtungen versehen, danach befördert das Förderband den Roboter zur zweiten Haltestelle. Hier werden die Ablage des Abschlussdeckels in das Zwischenlager und die Montage des Lagerdeckels ausgeführt. Danach wechselt der Roboter zum dritten Haltepunkt. Auf diesem erfolgt das Zusammensetzen der Kolbenstange mit dem Kolben und die Fertigmontage des Pneumatikzylinders.

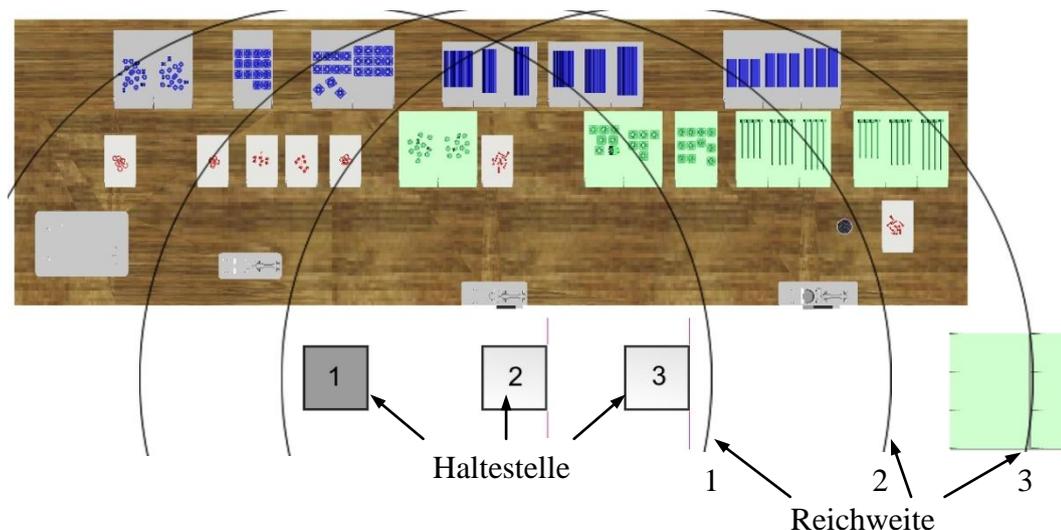


Abbildung 21: Reichweite des Roboters

Zwei Sensoren zeigen an, wenn das Förderband die Positionen erreicht hat und stellen dieses ab. So wird es möglich, dass die Haltestellen immer exakt eingehalten werden und der Roboter die ihm vorgegebenen Positionen erreicht. Die Sensoren arbeiten wie Reflexlichttaster (Abbildung 22). Hierbei wird der Strahl vom detektierten Objekt (im speziellen Fall die Transportplatte vom Roboter) zurückgeworfen.



Abbildung 22: Reflexlichttaster

Wie der Roboter ist auch das Förderband unsichtbar dargestellt, nur die Transportplatte muss sichtbar bleiben, da sie sonst nicht von den Sensoren erkannt wird. Die Positionen der Montagevorrichtungen auf dem Arbeitstisch ergeben sich aus den Montageprozessen.

Die Abbildung 23 auf der nächsten Seite zeigt die ganze Arbeitszelle und die Bezeichnung der einzelnen Behälter und der Module. Der Boden, die Arbeitstischplatte und die Wand wurden mit Texturen belegt, so dass sie möglichst realistisch dargestellt werden können.

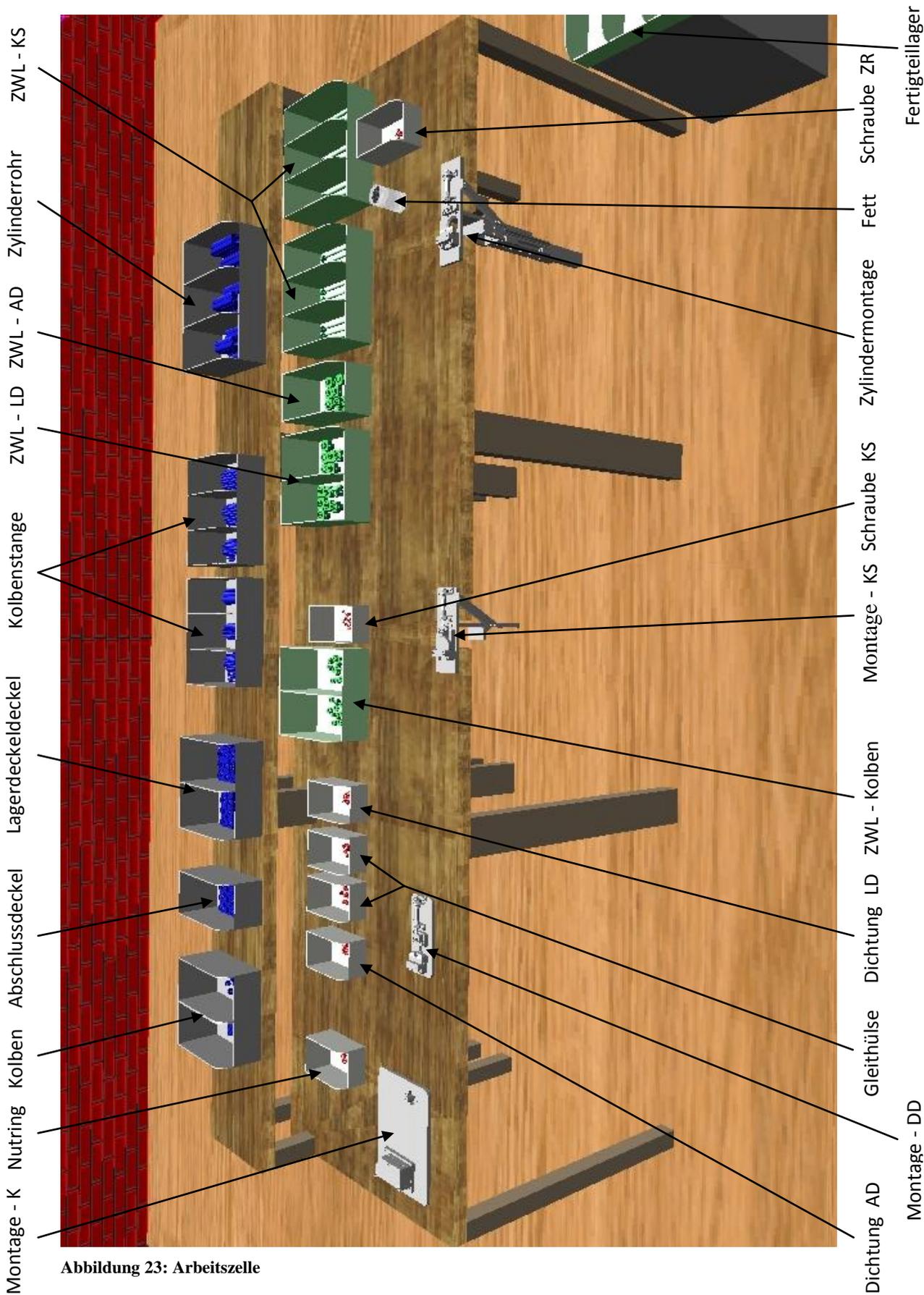


Abbildung 23: Arbeitszelle

7 Programm- und Videoerstellung

7.1 Programmierung

Es war wichtig für die Programmierung der Aufgaben eine einfache und schnell zu erlernende Programmiersprache zu verwenden, da das Programm an der HTBL Wien 10 weiterentwickelt und angepasst wird. So fiel die Wahl auf die Roboterprogrammiersprache MELFA-BASIC und in weiterer Folge auf einen Roboter der Firma Mitsubishi, welcher mit dieser Sprache programmiert werden kann.

Die Roboterprogrammiersprache MELFA-BASIC baut auf der universellen Sprache BASIC (Beginner's All-purpose Symbolic Instruction Code) auf. Das Entwicklungsziel der Sprache, eine für Anfänger leicht zu lernende und erweiterbare Sprache zu sein, wurde erreicht.³⁴ Da diese Sprache nicht standardisiert ist, gibt es eine Vielzahl an BASIC-Dialekten, auch MELFA-BASIC stellt so einen Dialekt dar. Dieser wurde um spezielle Befehle zur Bewegungssteuerung und Kommunikation mit anderen Komponenten in Arbeitszellen erweitert. Das in der Sprache implementierte „Worker-Subset“ ist eine Gruppe von einfachen Anweisungen, die leicht zu erlernen und anzuwenden sind, und so auch von Nicht-Spezialisten für Handhabungsaufgaben verwendet werden können.

³⁴ Vgl. <http://de.wikipedia.org/wiki/BASIC> [06. 05. 2008].

7.2 Programmbeschreibung

Bei umfangreicheren Programmieraufgaben gibt es eine allgemeine Vorgehensweise, welche auch bei dieser Diplomarbeit angewendet wurde.

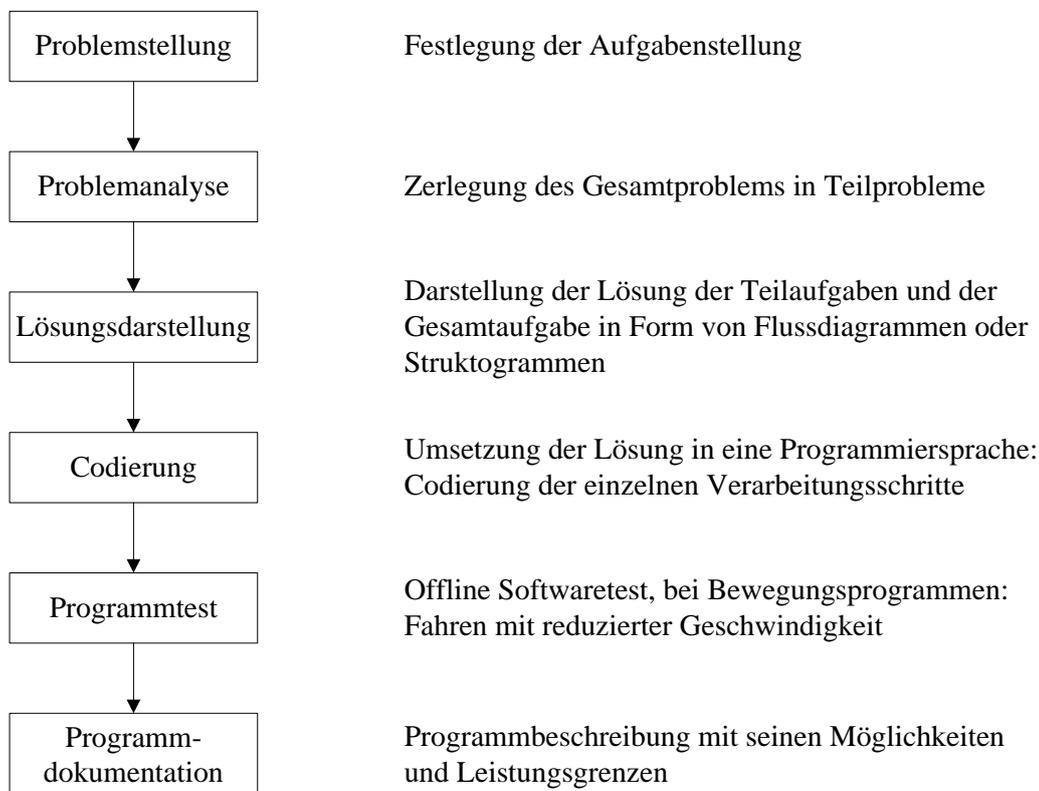


Abbildung 24: Allgemeine Vorgehensweise zur Programmerstellung

Die Aufgabe der Herstellung eines Pneumatikzylinders wurde stufenweise in immer kleinere Probleme zerlegt (top-down), dadurch erhält man eine verfeinerte Problemstruktur und kommt so zu immer einfacher strukturierten Teilproblemen. Diese Teilprobleme bildete ich in eigenen Programmen ab, wodurch das Hauptprogramm in verschiedene Programmteile zerfällt. Diese modulare Programmierung hilft das Programm wartbar zu halten, da die einzelnen Programme übersichtlich und abgeschlossen sind. Außerdem ist es nun möglich einzelne Programme für die verschiedenen Varianten der Pneumatikzylinder wieder zu verwenden, so wird das Programm für den Abschlussdeckel z. B. bei allen sechs Varianten verwendet.

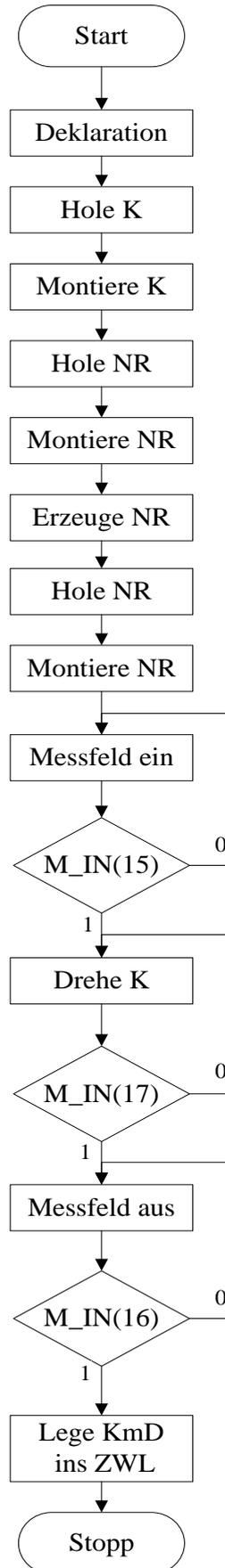
Auf der folgenden Seite ist exemplarisch das Flussdiagramm als Darstellung der Lösung für das Programm Kolben mit Dichtung abgebildet.

KmD6

10 REM * Zusammenbau Kolben DM6 *

```

20 DEF IO C_Close = Bit,5
30 DEF IO Messfeld = Bit,15
40 DEF IO KolbenDreh = Bit,16
50 C_Close = 0
60 JOVRD 10
70 MOV PK6      'Hole Kolben DM6
80 C_Close = 1
90 MOV P1
100 MOV P2
110 JOVRD 2
120 MOV PMontK  'Montagepos
130 JOVRD 10    'Kolben
140 C_Close = 0
150 JOVRD 100
160 MOV PNR     'Hole Nutring
170 C_Close = 1
180 JOVRD 10
190 MOV PNR, -80
200 MOV PMontNR, -60
210 JOVRD 1
220 MOV PMontNR  'Montiere NR
230 JOVRD 10
240 C_Close = 0
250 M_OUT(10)=1  'Neuer Nutring
260 MOV PNR     'Hole 2ten NR
270 C_Close = 1
280 MOV PNR, -80
290 MOV PMontNR, -60
300 JOVRD 1
310 MOV PMontNR, -5 'Montiere 2ten
320 JOVRD 10      'NR
330 C_Close = 0
340 Messfeld = 1  'Messfeld ein
350 WAIT M_IN(15) = 1
360 KolbenDreh = 1 'Drehe Kolben
370 WAIT M_IN(17) = 1
380 Messfeld = 0  'Messfeld aus
390 WAIT M_IN(16) = 1
400 MOV P7       'Lege K mit
410 C_Close = 1  'Dichtung ins ZWL
420 MOV P7, -60
430 OVRD 3
440 MVS PKmD6, -150
450 OVRD M_NOVRD
460 MOV PKmD6    'Lagerpos KmD6
470 C_Close = 0
480 MOV PStart
490 END
    
```



Die Zuordnung der Programme zu den jeweiligen Zylindersimulationen ist in Tabelle 10 ersichtlich. Oben waagrecht sind die sechs Zylindervarianten aufgelistet und senkrecht in den Spalten alle Unterprogramme. Das „x“ kennzeichnet nun, welches Unterprogramm in welcher Variante vorkommt.

	6x80	6x100	6x120	8x80	8x100	8x120
Main_6_80	x					
Main_6_100		x				
Main_6_120			x			
Main_8_80				x		
Main_8_100					x	
Main_8_120						x
KmD6	x	x	x			
KmD8				x	x	x
ADmD	x	x	x	x	x	x
LDmD6	x	x	x			
LDmD8				x	x	x
KSmK6x140	x					
KSmK6x165		x				
KSmK6x185			x			
KSmK8x140				x		
KSmK8x165					x	
KSmK8x185						x
SchraubenLD110	x			x		
SchraubenLD130		x			x	
SchraubenLD150			x			x
SchraubenAD110	x			x		
SchraubenAD130		x			x	
SchraubenAD150			x			x
ZYL6x80	x					
ZYL6x100		x				
ZYL6x120			x			
ZYL8x80				x		
ZYL8x100					x	
ZYL8x120						x

Tabelle 10: Programmzuordnung

Mit der COSIMIR Werkzeug Projektverwaltung werden die einzelnen Roboterprogramme den jeweiligen Zusammenbauvarianten zugeteilt. Es ist zudem

möglich eine Projektinfo zu hinterlegen. Die Dialogseite Dateien (siehe Abbildung 25) stellt alle zum Projekt gehörigen Programmdateien und Positionslisten dar.

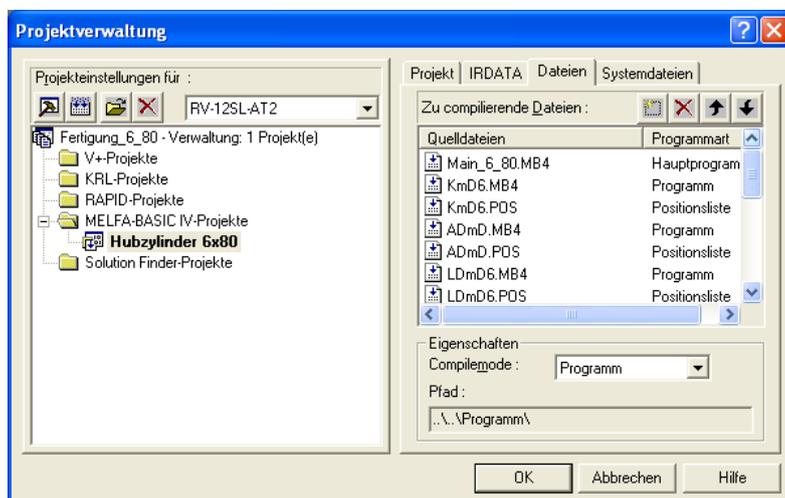


Abbildung 25: Projektverwaltung

Ein Programm muss immer als Hauptprogramm definiert werden. Bei allen Varianten ist dies das Programm mit den für die jeweilige Variante geltenden Namen (z. B.: Main_6_80.MB4). Dieses wird vom Compiler als Hauptprogramm übersetzt, da die Programmiersprache MELFA BASIC IV kein eigentliches Startprogramm besitzt. Positionslisten – Erkennbar an der Endung .POS – werden hingegen nicht übersetzt.

7.3 Aufzeichnen der Simulation als Video

COSIMIR ermöglicht durch das Tool Kamerafahrt die Aufzeichnung einer Simulation. Man speichert dazu verschiedene Ansichten eines Arbeitszellenfensters. Bei einer laufenden Simulation werden diese Ansichten der Reihe nach angefahren, dazwischen wird linear interpoliert, um so eine gleichmäßige Bewegung der Kamerafahrt zu suggerieren.

Die Kamerafahrt ist mit der Simulationszeit synchronisiert, daher erfolgt die Veränderung der Ansicht synchron mit der simulierten Arbeitszelle.

Für die Steuerung der Kamerafahrt stehen in der Werkzeugleiste, wie in Abbildung 26 ersichtlich, verschiedene Schaltflächen zur Verfügung.

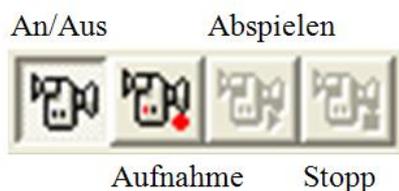


Abbildung 26: Schaltflächen Kamerafahrt

- Kamerafahrt An/Aus: Ein-/Ausschalten der Kamerafahrt. Die Kamerafahrt wird nur bei einer laufenden Simulation durchgeführt.
- Kamerafahrt Aufnahme: Aufnahme der Kamerafahrt starten. Die Ansichten der Kamerafahrt werden in eine Datei gespeichert.
- Kamerafahrt Abspielen: Abspielen der aufgenommenen Kamerafahrt.
- Kamerafahrt Stopp: Stoppen der Aufnahme oder der Wiedergabe der Kamerafahrt.

Unter dem Menübefehl Extras/Einstellungen/Kamerafahrt kann man eine Kamerafahrt für ein Simulationsmodell konfigurieren. Wie man in Abbildung 27 sehen kann, sind in dieser Liste alle Ansichten der Kamerafahrt aufgeführt. Die Haltezeit gibt an, wie lange die eingestellte Position gehalten werden soll. Die Zoomzeit ist jene Zeit zwischen dem Verlassen der einen und dem Erreichen der nächsten Position. All diese Zeiten und Einstellungen wurden bei der Diplomarbeit so abgestimmt, dass sie den Bewegungen des unsichtbaren Roboterarmes folgen. Es sollte so der Eindruck vermittelt werden, als sei der Betrachter derjenige, der den Zylinder zusammenbaut.



Abbildung 27: Dialogfeld Kamerafahrt

Mit der Schaltfläche Einstellungen gelangt man zum Dialogfeld Einstellungen – Video (Abbildung 28). Die getroffenen Einstellungen für das Video sind folgende:

Bei der Aufnahme sollte jedes Bild aufgenommen werden um das Video später ruckelfrei abspielen zu können. Die Dauer der Aufnahme wurde mit „Genau einen Zyklus“ gewählt, das heißt sie stoppt nach Beendigung der Simulation und aufgenommen werden soll nur das Arbeitszellenfenster.



Abbildung 28: Dialogfeld Einstellungen – Video

Zur Erstellung des Videos wurden noch die Simulationseinstellungen (siehe Abbildung 29) geändert. So mussten der Simulationstakt und der maximale Simulationstakt auf 0,05 Sekunden gesenkt werden, um eine möglichst gleichförmige, nicht stockende Bewegung zu erhalten. Der Simulationstakt gibt an, nach welcher simulierten Zeit die grafische Darstellung aktualisiert werden soll. Das heißt, je kleiner dieser Wert gewählt wird desto fließender wird die Anzeige der Bewegungen, aber auch umso langsamer, da rechenaufwändiger, wird die Simulation.

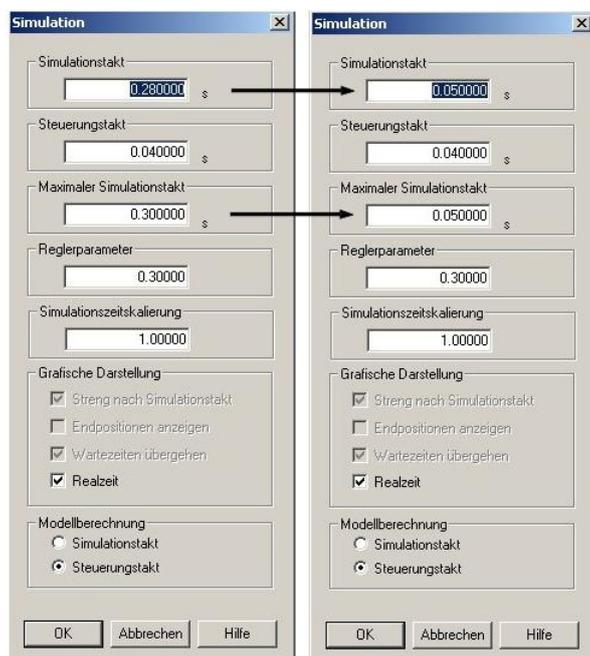


Abbildung 29: Simulationseinstellungen

Der Steuerungstakt ist der Takt, der zur Berechnung von Zwischenpositionen bei Robotersteuerungen und als Zykluszeit bei speicherprogrammierbaren Steuerungen benutzt wird. Dieser Wert muss auf 0,04 Sekunden verbleiben, da sonst die zuvor programmierten Positionen und Kameraeinstellungen nicht mehr übereinstimmen.

Die Komprimierung des Videos erfolgte mit dem Komprimierer „Microsoft Video 1“ bei einer Komprimierungsqualität von 100. Das so erhaltene Video wurde im avi-Format abgespeichert.

7.4 Aufbau der Ordner-Struktur

Wie auch bei der Programmierung war es wichtig eine gewisse Ordnung bei den Verzeichnissen einzuhalten, dies auch unter dem Aspekt der Wartbarkeit.

Die nachfolgende Verzeichnisstruktur wurde bei der Erstellung der Diplomarbeit verwendet:

<Installationsverzeichnis>\...\

Verzeichnis	Beschreibung
\...\	Modellverzeichnis
Import	3D-CAD-Daten in verschiedenen Grafikaustauschformaten
Modell	Hauptverzeichnis für die Modelldatei
Parts	Teilmodelle, die als Bibliotheken eingebunden werden
Positionen	Positionslisten für die Roboterprogrammierung
Programm	Roboterprogramme
Texturen	Texturen für Materialien

Tabelle 11: Ordner-Struktur

- **Import**
Hier befinden sich alle relevanten VRML-Files. Diese Objekte können in ein 3D-Konstruktionsprogramm eingelesen und bei Bedarf geändert werden. So ist die Möglichkeit gegeben, im Nachhinein konstruktive Änderungen am Modell vorzunehmen.
- **Modell**
In den diversen Unterordnern befinden sich alle sechs Simulationsvarianten. Die Unterordner beinhalten die Arbeitszelle und die gespeicherten Kamerapositionen und -zeiten.
- **Parts**
Unter den jeweiligen Unterordnern sind Behälter, Lager-, Zukaufteile und Halbfabrikate gespeichert.
- **Positionen**
Hier befinden sich alle Positionslisten. Der Name der Positionsliste muss gleich dem Programmnamen sein, da die Zuordnung über den Namen stattfindet.

- Programm
Alle Roboterprogramme wurden hier gespeichert. Der Dateiname bezieht sich auf das zusammenzubauende Objekt.
- Texturen
In diesem Ordner befinden sich die Bilder für die Oberflächentexturen.

8 Zusammenfassung und Ausblick

In der Lehr- und Lernproduktion wird der Normalbetrieb eines Unternehmens nachgespielt. So kann man dort unternehmerische Herausforderungen wie zum Beispiel Auftragsänderungen, Qualitätsprobleme, Personalausfälle usw. ablaufen lassen. Den Lernenden wird es mit der Lehr- und Lernproduktion ermöglicht Entscheidungen zu treffen und aus den Konsequenzen zu lernen.

Die Lehr- und Lernproduktion versucht den Unternehmensalltag möglichst realistisch darzustellen. Dies macht es nötig, in gewissen Bereichen auf Simulationen zurückzugreifen. Ziel dieser Diplomarbeit war es, eine größere Variantenvielfalt in der Herstellung von Pneumatikzylindern zu simulieren. Dazu wurde die von der Firma Festo Didactic bereitgestellte Simulationssoftware COSIMIR Professional verwendet.

Alle Module und die Bestandteile des Zylinders waren bereits als CATIA Modelle vorhanden. Diese wurden so aufbereitet, dass sie in die Simulationssoftware eingespielt werden konnten. Den Modulen wurden nun die beweglichen Mechanismen zugewiesen, wodurch es möglich wurde einzelne Modulbestandteile in der Simulation zu bewegen. Den Bestandteilen des Zylinders wurden Greifpunkte zugefügt, um sie mit dem Roboter fassen zu können. Dieser Roboter wurde für die Simulation nötig, da kein humanoider Dummy zur Verfügung stand.

Danach wurde der reale Arbeitsplatz in COSIMIR nachgebildet. Dazu wurden die Arbeitstische, die Behälter der Lager- und Zukaufteile sowie die der Halbfabrikate erstellt und platziert, um so möglichst realitätsnahe Umstände zu schaffen.

Als nächstes wurden die einzelnen Roboterprogramme für den Zusammenbau der Halbfabrikate und schließlich des gesamten Pneumatikzylinders in der Programmiersprache MELFA-BASIC IV erstellt. Mit dieser modularen

Programmierung wurde es möglich, verschiedene Varianten des Zusammenbaus zu simulieren.

Um ein Abspielen der Simulation auch ohne die Software COSIMIR zu ermöglichen, wurde sie als Video-File abgespeichert – so kann die Simulation einfach von einer DVD abgespielt werden.

Für die LLP werden in nächster Zukunft so genannte Drehbücher geschrieben. In diese Drehbücher werden verschiedene Fehlerszenarien und die Möglichkeit zur Korrektur eingebaut. Mit dieser Diplomarbeit wurde die Basis für die Simulation mehrerer Szenarien in der Lehr- und Lernproduktion geschaffen und es wird möglich, diese Drehbücher in der Simulation wiederzugeben. Um die Simulation ohne lange Einarbeitungszeit und großen Aufwand an die Drehbücher anpassen zu können, wurde darauf geachtet, eine schnell zu erlernende Programmiersprache zu verwenden.

9 Anhang

9.1 Verwendete Befehle

<i>Befehle</i>	<i>Beschreibung</i>
CALLP	Programm aufrufen
DEF IO	Ein-/Ausgangsvariablen definieren
DLY	Verzögerung einstellen
END	Programmende
JOVRD	Geschwindigkeitsübersteuerung für Gelenk-Interpolation festlegen
MOV	Bewegung mit Gelenk-Interpolation
MVS	geradlinige Bewegung
OVRD	Übersteuerung
REM	Kommentar
SPD	Geschwindigkeit festlegen
WAIT	Wartestatus definieren

9.2 Programmcode

Nachfolgend sind alle Programme aus allen sechs Simulationen abgebildet. Die Reihenfolge ist die gleiche wie in Tabelle 10.

Main_6_80

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD6"
30 CALLP "ADmD"
40 CALLP "LDmD6"
50 CALLP "KSmK6x140"
60 CALLP "ZYL6x80"

70 End
```

Main_6_100

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD6"
30 CALLP "ADmD"
40 CALLP "LDmD6"
50 CALLP "KSmK6x165"
60 CALLP "ZYL6x100"

70 End
```

Main_6_120

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD6"
30 CALLP "ADmD"
40 CALLP "LDmD6"
50 CALLP "KSmK6x185"
60 CALLP "ZYL6x120"

70 End
```

Main_8_80

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD8"
30 CALLP "ADmD"
40 CALLP "LDmD8"
50 CALLP "KSmK8x140"
60 CALLP "ZYL8x80"

70 End
```

Main_8_100

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD8"
30 CALLP "ADmD"
40 CALLP "LDmD8"
50 CALLP "KSmK8x165"
60 CALLP "ZYL8x100"

70 End
```

Main_8_120

```
10 REM *Aufrufen der
Unterprogramme*

20 CALLP "KmD8"
30 CALLP "ADmD"
40 CALLP "LDmD8"
50 CALLP "KSmK8x185"
60 CALLP "ZYL8x120"

70 End
```

KmD6

10 REM * Zusammenbau Kolben DM6 *

20 DEF IO C_Close = Bit,5

30 DEF IO Messfeld = Bit,15

40 DEF IO KolbenDreh = Bit,16

50 C_Close = 0

60 JOVRD 10

70 MOV PK6 *'Hole Kolben DM6*

80 C_Close = 1

90 MOV P1

100 MOV P2

110 JOVRD 2

120 MOV PMontK *'Montagepos Kolben*

130 JOVRD 10

140 C_Close = 0

150 JOVRD 100

160 MOV PNR *'Hole Nutring*

170 C_Close = 1

180 JOVRD 10

190 MOV PNR, -80

200 MOV PMontNR, -60

210 JOVRD 1

220 MOV PMontNR *'Montiere NR*

230 JOVRD 10

240 C_Close = 0

250 M_OUT(10)=1 *'Neuer Nutring*

260 MOV PNR *'Hole 2ten NR*

270 C_Close = 1

280 MOV PNR, -80

290 MOV PMontNR, -60

300 JOVRD 1

310 MOV PMontNR, -5 *'Montiere 2ten NR*

320 JOVRD 10

330 C_Close = 0

340 Messfeld = 1 *'Messfeld ein*

350 WAIT M_IN(15) = 1

360 KolbenDreh = 1 *'Drehe Kolben*

370 WAIT M_IN(17) = 1

380 Messfeld = 0 *'Messfeld aus*

390 WAIT M_IN(16) = 1

400 MOV P7 *'Lege K mit Dichtung ins ZWL*

410 C_Close = 1

420 MOV P7, -60

430 OVRD 3

440 MVS PKmD6, -150

```
450 OVRD M_NOVRD
460 MOV PKmD6   'Lagerpos KmD6
470 C_Close = 0
480 MOV PStart
490 END
```

KmD8

```
10 REM *Zusammenbau Kolben DM8*
```

```
20 DEF IO C_Close = Bit,5
30 DEF IO Messfeld = Bit,15
40 DEF IO KolbenDreh = Bit,16

50 C_Close = 0
60 JOVRD 100
70 MOV PK8       'Hole Kolben DM8
80 C_Close = 1
90 JOVRD 10
100 MOV P1
110 OVRD 2
120 MVS P2
130 OVRD M_NOVRD
140 JOVRD 2
150 MOV PMontK   'Montagepos Kolben
160 JOVRD 10
170 C_Close = 0
180 JOVRD 100
190 MOV PNR      'Hole Nutring
200 C_Close = 1
210 JOVRD 10
220 MOV PNR, -80
230 MOV PMontNR, -60
240 JOVRD 1
250 MOV PMontNR 'Montiere NR
260 JOVRD 10
270 C_Close = 0
280 M_OUT(10)=1 'Neuer Nutring
290 MOV PNR      'Hole 2ten Nutring
300 C_Close = 1
310 MOV PNR, -80
320 MOV PMontNR, -60
330 JOVRD 1
340 MOV PMontNR, -5 'Montiere 2ten NR
350 JOVRD 10
360 C_Close = 0
370 Messfeld = 1   'Messfeld ein
380 WAIT M_IN(15) = 1
```

```
390 KolbenDreh = 1  'Drehe Kolben
400 WAIT M_IN(17) = 1
410 Messfeld = 0    'Messfeld aus
420 WAIT M_IN(16) = 1
430 MOV P7          'Lege KmD ins ZWL
440 C_Close = 1
450 MOV P7, -60
460 OVRD 3
470 MVS PKmD8, -150
480 OVRD M_NOVRD
490 MOV PKmD8      'Lagerpos KmD8
500 C_Close = 0
510 MOV PStart
520 END
```

ADmD

```
10 REM *Zusammenbau Abschlussdeckel*

20 DEF IO C_Close = Bit,5
30 DEF IO DeckelKlemm = Bit,17
40 DEF IO BandAn = Bit,7

50 C_Close = 0
60 JOVRD 100
70 MOV PAD      'Hole Abschlussdeckel
80 C_Close = 1
90 JOVRD 10
100 MOV PAD, -100
110 OVRD 3
120 MVS PMontAD,-60
130 OVRD M_NOVRD
140 JOVRD 1
150 MOV PMontAD 'Montagepos AD
160 C_Close = 0
170 DeckelKlemm = 1 'Klemme Deckel
180 JOVRD 100
190 MOV POR1     'Hole Dichtung AD
200 C_Close = 1
210 JOVRD 10
220 MOV POR1, -80
230 MOV PMontOR1,-40
240 JOVRD 1
250 MOV PMontOR1 'Montiere Dichtung
260 JOVRD 10
270 C_Close = 0
280 DeckelKlemm = 0 'Löse Deckel
```

```
290 MOV PMontAD   'Lege AD mit Dichtung
300 C_Close = 1   'ins ZWL
310 MOV PMontAD,-100
320 BandAn = 1    'Starte Förderband
330 WAIT M_IN(9) = 1 'Stopp MM2
340 BandAn = 0
350 OVRD 5
360 MVS PADmD, -150
370 OVRD M_NOVRD
380 JOVRD 3
390 MOV PADmD   'Lagerpos ADmD
400 JOVRD 100
410 C_Close = 0
420 MOV PStart
430 JOVRD 10
450 END
```

LDmD6

```
10 REM *Zusammenbau Lagerdeckel DM6*

20 DEF IO C_Close = Bit,5
30 DEF IO DeckelKlemm = Bit,17

40 C_Close = 0
50 JOVRD 30
60 MOV PLD6   'Hole Lagerdeckel
70 JOVRD 10
80 C_Close = 1
90 MOV P1
100 MOV P2
110 JOVRD 1
120 MOV PMontLD 'Montagepos LD
130 C_Close = 0
140 DeckelKlemm = 1 'Klemme Deckel
150 JOVRD 100
160 MOV POR2   'Hole Dichtung LD
170 C_Close = 1
180 JOVRD 10
190 MOV POR2, -90
200 MOV PMontOR,-40
210 JOVRD 1
220 MOV PMontOR 'Montiere Dichtung
230 C_Close = 0
240 DeckelKlemm = 0
250 JOVRD 100
260 MOV PMontLD
270 C_Close = 1
```

```
280 JOVRD 10
290 MOV P2
300 MOV PWenden 'Deckel wenden für
310 MOV PMontLD2 'GB-Montage
320 C_Close = 0
330 DeckelKlemm = 1
340 JOVRD 100
350 MOV PGB6 'Hole Gleitbuchse 6
360 C_Close = 1
370 JOVRD 10
380 MOV PGB6, -80
390 MOV PMontGB,-40
400 JOVRD 1
410 MOV PMontGB 'Montiere GB
420 C_Close = 0
430 DeckelKlemm = 0
440 JOVRD 100
450 MOV PT12 'Lege LDmD ins ZWL
460 C_Close = 1
470 JOVRD 10
480 MOV PT12, -100
490 OVRD 3
500 MVS PT13
510 OVRD M_NOVRD
520 MOV PLDmD6 'Lagerpos LD mit
530 C_Close = 0 'Dichtung
540 JOVRD 100
550 MOV PStart
560 JOVRD 10

570 END
```

LDmD8

```
10 REM *Zusammenbau Lagerdeckel DM8*

20 DEF IO C_Close = Bit,5
30 DEF IO DeckelKlemm = Bit,17

40 C_Close = 0
50 JOVRD 50
60 MOV PLD8 'Hole Lagerdeckel
70 JOVRD 10
80 C_Close = 1
90 MOV P1
100 MOV P2
110 JOVRD 1
120 MOV PMontLD 'Montagepos LD
```

```
130 C_Close = 0

140 DeckelKlemm = 1 'Klemme Deckel
150 JOVRD 100
160 MOV POR2 'Hole Dichtung LD
170 C_Close = 1
180 JOVRD 10
190 MOV POR2, -90
200 MOV PMontOR, -40
210 JOVRD 1
220 MOV PMontOR 'Montiere Dichtung
230 C_Close = 0
240 DeckelKlemm = 0
250 JOVRD 100
260 MOV PMontLD
270 C_Close = 1
280 JOVRD 10

290 MOV P2
300 MOV PWenden 'Deckel wenden für
310 MOV PMontLD2 'GB-Montage
320 C_Close = 0
330 DeckelKlemm = 1
340 JOVRD 100
350 MOV PGB8 'Hole Gleitbuchse 8
360 C_Close = 1
370 JOVRD 10
380 MOV PGB8, -80
390 MOV PMontGB, -40
400 JOVRD 1
410 MOV PMontGB 'Montiere GB
420 C_Close = 0
430 DeckelKlemm = 0
440 JOVRD 100
450 MOV P12 'Lege LDmD ins ZWL
460 C_Close = 1
470 JOVRD 10
480 MOV P12, -100
490 OVRD 3
500 MVS P13
510 OVRD M_NOVRD
520 MOV PLDmD8 'Lagerpos LD mit Dichtung
530 C_Close = 0
540 JOVRD 100
550 MOV PStart
560 JOVRD 10

570 END
```

KSmK6x140

```
10 REM * Zusammenbau Kolbenstange*
20 REM *           mit Kolben           *
30 REM *           6x140                 *

40 DEF IO C_Close = Bit,5
50 DEF IO KSKlemm = Bit,18
60 DEF IO BandAn = Bit,7

70 BandAn = 1           'Weiterfahrt auf MM3
80 WAIT M_IN(10)= 1
90 BandAn = 0

100 C_Close = 0
110 JOVRD 100
120 MOV PKS140   'Hole Kolbenstange
130 C_Close = 1
140 JOVRD 10
150 MOV P1
160 OVRD 5
170 MVS P2
180 OVRD 2
190 MVS PMontKS 'Montagepos KS
200 OVRD M_NOVRD
210 C_Close = 0
220 KSKlemm = 1   'Klemme KS

230 JOVRD 100
240 MOV PKmD6   'Hole Kolben mit Dichtung
250 C_Close = 1
260 JOVRD 10
270 MOV PKmD6, -100
280 MOV PMontKmD,-50
290 JOVRD 2
300 MOV PMontKmD 'Montiere KmD
310 JOVRD 100
320 C_Close = 0

330 MOV PI6kt   'Hole Schraube
340 C_Close = 1
350 JOVRD 10
360 MOV P4
370 MOV P2
380 MOV P5
390 JOVRD 1
400 MOV PMontI6k 'Montiere Schraube
410 C_Close = 0
```

420 JOVRD 10

430 MOV PMontKS *'Lege KS mit Kolben ins ZWL*

440 KSKlemm = 0

450 C_Close = 1

460 OVRD 1

470 MVS P2

480 MVS P3

490 OVRD M_NOVRD

500 MOV PKSmK *'Lagerpos KSmK*

510 C_Close = 0

520 JOVRD M_NJOVRD

530 MOV PStart

540 END

KSmK6x165

10 REM * Zusammenbau Kolbenstange*

20 REM * mit Kolben *

30 REM * 6x165 *

40 DEF IO C_Close = Bit,5

50 DEF IO KSKlemm = Bit,18

60 DEF IO BandAn = Bit,7

70 BandAn = 1 *'Weiterfahrt auf MM3*

80 WAIT M_IN(10)= 1

90 BandAn = 0

100 C_Close = 0

110 JOVRD 100

120 MOV PKS165 *'Hole Kolbenstange*

130 C_Close = 1

140 JOVRD 10

150 MOV P1

160 OVRD 4

170 MVS P2

180 OVRD 1

190 MVS PMontKS *'Montagepos KS*

200 OVRD M_NOVRD

210 C_Close = 0

220 KSKlemm = 1 *'Klemme KS*

230 JOVRD 100

240 MOV PKmD6 *'Hole Kolben mit Dichtung*

250 C_Close = 1

```

260 JOVRD 10
270 MOV PKmD6, -100
280 MOV PMontKmD,-50
290 JOVRD 2
300 MOV PMontKmD 'Montiere KmD
310 JOVRD 100
320 C_Close = 0

330 MOV PI6kt      'Hole Schraube
340 C_Close = 1
350 JOVRD 10
360 MOV P4
370 MOV P2
380 MOV P5
390 JOVRD 1
400 MOV PMontI6k 'Montiere Schraube
410 C_Close = 0
420 JOVRD 10

430 MOV PHolKS    'Lege KS mit Kolben ins ZWL
440 KSKlemm = 0
450 C_Close = 1
460 OVRD 2
470 MVS P6
480 MVS P7
490 OVRD M_NOVRD
500 JOVRD 1000
510 C_Close = 0
520 MOV P8        'Umgreifen
530 C_Close = 1
540 JOVRD 10
550 MOV P3
560 MOV PKSmK    'Lagerpos KSmK
570 C_Close = 0

580 JOVRD 100
590 MOV PStart

600 END

```

KSmK6x185

```

10 REM * Zusammenbau Kolbenstange*
20 REM *           mit Kolben      *
30 REM *           6x185          *

40 DEF IO C_Close = Bit,5
50 DEF IO KSKlemm = Bit,18

```

```
60 DEF IO BandAn = Bit,7

70 BandAn = 1      'Weiterfahrt auf MM3
80 WAIT M_IN(10)= 1
90 BandAn = 0
100 C_Close = 0
110 JOVRD 100
120 MOV PKS185    'Hole Kolbenstange
130 C_Close = 1
140 JOVRD 10
150 MOV P1
160 OVRD 4
170 MVS P2
180 OVRD 1
190 MVS PMontKS  'Montagepos KS
200 OVRD M_NOVRD
210 C_Close = 0
220 KSKlemm = 1  'Klemme KS

230 JOVRD 100
240 MOV PKmD6    'Hole Kolben mit Dichtung
250 C_Close = 1
260 JOVRD 10
270 MOV PKmD6, -100
280 MOV PMontKmD,-50
290 JOVRD 2
300 MOV PMontKmD 'Montiere KmD
310 JOVRD 100
320 C_Close = 0

330 MOV PI6kt    'Hole Schraube
340 C_Close = 1
350 JOVRD 10
360 MOV P4
370 MOV P2
380 MOV P5
390 JOVRD 1
400 MOV PMontI6k 'Montiere Schraube
410 C_Close = 0
420 JOVRD 100

430 MOV PHolKS   'Lege KSmK ins ZWL
440 KSKlemm = 0
450 C_Close = 1
460 JOVRD 10
470 OVRD 2
480 MVS P6
490 MVS P7
```

```
500 OVRD M_NOVRD
510 JOVRD 1000
520 C_Close = 0
530 MOV P8      'Umgreifen
540 C_Close = 1
550 JOVRD 10
560 MOV P3
570 MOV PKSmK   'Lagerpos KSmK
580 C_Close = 0

590 JOVRD 100
600 MOV PStart

610 END

KSmK8x140

10 REM * Zusammenbau Kolbenstange*
20 REM *      mit Kolben      *
30 REM *      8x140          *

40 DEF IO C_Close = Bit,5
50 DEF IO KSKlemm = Bit,18
60 DEF IO BandAn = Bit,7

70 BandAn = 1      'Weiterfahrt auf MM3
80 WAIT M_IN(10)= 1
90 BandAn = 0

100 C_Close = 0
110 JOVRD 100
120 MOV PKS8x140 'Hole Kolbenstange
130 C_Close = 1
140 JOVRD 10
150 MOV P1
160 OVRD 5
170 MVS P2
180 OVRD 2
190 MVS PMontKS 'Montagepos KS
200 OVRD M_NOVRD
210 C_Close = 0
220 KSKlemm = 1   'Klemme KS

230 JOVRD 100
240 MOV PKmD8     'Hole Kolben mit Dichtung
250 C_Close = 1
260 JOVRD 10
270 MOV PKmD8, -100
```

```

280 OVRD 3
290 MVS PMontKmD,-50
300 OVRD M_NOVRD
310 JOVRD 2
320 MOV PMontKmD 'Montiere KmD
330 JOVRD 100
340 C_Close = 0

350 MOV PI6kt      'Hole Schraube
360 C_Close = 1
370 JOVRD 10
380 MOV P4
390 MOV P2
400 MOV P5
410 JOVRD 1
420 MOV PMontI6k 'Montiere Schraube
430 C_Close = 0
440 JOVRD 100

450 MOV PHolKS    'Lege KS mit Kolben ins ZWL
460 KSKlemm = 0
470 C_Close = 1
480 JOVRD 10
490 OVRD 2
500 MVS P6
510 MVS P7
520 OVRD M_NOVRD
530 JOVRD 1000
540 C_Close = 0
550 MOV P8        'Umgreifen
560 C_Close = 1
570 JOVRD 10
580 MOV P3
590 MOV PKSmK     'Lagerpos KSmK
600 C_Close = 0

610 JOVRD 100
620 MOV PStart

630 END

```

KSmK8x165

```

10 REM * Zusammenbau Kolbenstange*
20 REM *           mit Kolben      *
30 REM *           8x165           *

```

```

40 DEF IO C_Close = Bit,5

```

```
50 DEF IO KSKlemm = Bit,18
60 DEF IO BandAn = Bit,7

70 BandAn = 1      'Weiterfahrt auf MM3
80 WAIT M_IN(10)= 1
90 BandAn = 0
100 C_Close = 0
110 JOVRD 100
120 MOV PKS8x165 'Hole Kolbenstange
130 C_Close = 1
140 JOVRD 10
150 MOV P1
160 OVRD 5
170 MVS P2
180 OVRD 2
190 MVS PMontKS 'Montagepos KS
200 OVRD M_NOVRD
210 C_Close = 0
220 KSKlemm = 1   'Klemme KS

230 JOVRD 100
240 MOV PKmD8    'Hole Kolben mit Dichtung
250 C_Close = 1
260 JOVRD 10
270 MOV PKmD8, -100
280 OVRD 3
290 MVS PMontKmD,-50
300 OVRD M_NOVRD
310 JOVRD 2
320 MOV PMontKmD 'Montiere KmD
330 JOVRD 100
340 C_Close = 0

350 MOV PI6kt    'Hole Schraube
360 C_Close = 1
370 JOVRD 10
380 MOV P4
390 MOV P2
400 MOV P5
410 JOVRD 1
420 MOV PMontI6k 'Montiere Schraube
430 C_Close = 0
440 JOVRD 100

450 MOV PHolKS   'Lege KSmK ins ZWL
460 KSKlemm = 0
470 C_Close = 1
480 JOVRD 10
```

```

490 OVRD 2
500 MVS P6
510 MVS P7
520 OVRD M_NOVRD
530 JOVRD 1000
540 C_Close = 0
550 MOV P8      'Umgreifen
560 C_Close = 1
570 JOVRD 10
580 MOV P3
590 MOV PKSmK  'Lagerpos Kolbenstange mit Kolben
600 C_Close = 0

610 JOVRD 100
620 MOV PStart

630 END

```

KSmK8x185

```

10 REM * Zusammenbau Kolbenstange*
20 REM *      mit Kolben      *
30 REM *      8x185          *

40 DEF IO C_Close = Bit,5
50 DEF IO KSKlemm = Bit,18
60 DEF IO BandAn = Bit,7

70 BandAn = 1      'Weiterfahrt auf MM3
80 WAIT M_IN(10)= 1
90 BandAn = 0

100 C_Close = 0
110 JOVRD 100
120 MOV PKS8x185 'Hole Kolbenstange
130 C_Close = 1
140 JOVRD 10
150 MOV P1
160 OVRD 5
170 MVS P2
180 OVRD 2
190 MVS PMontKS 'Montagepos KS
200 OVRD M_NOVRD
210 C_Close = 0
220 KSKlemm = 1   'Klemme KS

230 JOVRD 100
240 MOV PKmD8    'Hole Kolben mit Dichtung

```

```
250 C_Close = 1
260 JOVRD 10
270 MOV PKmD8, -100
280 OVRD 3
290 MVS PMontKmD,-50
300 OVRD M_NOVRD
310 JOVRD 2
320 MOV PMontKmD 'Montiere KmD
330 JOVRD 100
340 C_Close = 0

350 MOV PI6kt 'Hole Schraube
360 C_Close = 1
370 JOVRD 10
380 MOV P4
390 MOV P2
400 MOV P5
410 JOVRD 1
420 MOV PMontI6k 'Montiere Schraube
430 C_Close = 0
440 JOVRD 100

450 MOV PHolKS 'Lege KS mit Kolben ins ZWL
460 KSKlemm = 0
470 C_Close = 1
480 JOVRD 10
490 OVRD 2
500 MVS P6
510 MVS P7
520 OVRD M_NOVRD
530 JOVRD 1000
540 C_Close = 0
550 MOV P8 'Umgreifen
560 C_Close = 1
570 JOVRD 10
580 MOV P3
590 MOV PKSmK 'Lagerpos KSmK
600 C_Close = 0

610 JOVRD 100
620 MOV PStart

630 END
```

SchraubenLD110

```
10 REM * Montieren der Schrauben *
20 REM *   am Lagerdeckel   *
30 REM *   bei Zylinderrohr 110 *

40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11

60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL 'Montiere Schraube links
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1   'Neue Schraube
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR 'Montiere Schraube rechts
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100

270 NeuerI6kt = 1   'Neue Schraube
280 MOV PI6kt
290 C_Close = 1
300 JOVRD 10
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV 'Montiere Schraube vorne
350 C_Close = 0
360 JOVRD 100

370 NeuerI6kt = 0
380 NeuerI6kt = 1   'Neue Schraube
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10
```

```
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH 'Montiere Schraube hinten
460 NeuerI6kt = 0
470 C_Close = 0

480 NeuerI6kt = 1 'Neue Schraube
490 NeuerI6kt = 0

500 END
```

SchraubenLD130

```
10 REM * Montieren der Schrauben *
20 REM *   am Lagerdeckel   *
30 REM *   bei Zylinderrohr 130 *

40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11

60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL 'Montiere Schraube links
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1 'Neue Schraube
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR 'Montiere Schraube rechts
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100

270 NeuerI6kt = 1 'Neue Schraube
280 MOV PI6kt
290 C_Close = 1
300 JOVRD 10
```

```
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV 'Montiere Schraube vorne
350 C_Close = 0
360 JOVRD 100

370 NeuerI6kt = 0
380 NeuerI6kt = 1 'Neue Schraube
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH 'Montiere Schraube hinten
460 NeuerI6kt = 0
470 C_Close = 0

480 NeuerI6kt = 1 'Neue Schraube
490 NeuerI6kt = 0

500 END
```

SchraubenLD150

```
10 REM * Montieren der Schrauben *
20 REM *   am Lagerdeckel   *
30 REM *   bei Zylinderrohr 150 *

40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11

60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL 'Montiere Schraube links
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1 'Neue Schraube
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
```

```
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR 'Montiere Schraube rechts
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100
```

```
270 NeuerI6kt = 1 'Neue Schraube
280 MOV PI6kt
290 C_Close = 1
300 JOVRD 10
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV 'Montiere Schraube vorne
350 C_Close = 0
360 JOVRD 100
```

```
370 NeuerI6kt = 0
380 NeuerI6kt = 1 'Neue Schraube
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH 'Montiere Schraube hinten
460 NeuerI6kt = 0
470 C_Close = 0
```

```
480 NeuerI6kt = 1 'Neue Schraube
490 NeuerI6kt = 0
```

```
500 END
```

SchraubenAD110

```
10 REM * Montieren der Schrauben *
20 REM * am Abschlussdeckel *
30 REM * bei Zylinderrohr 110 *
```

```
40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11
```

```
60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
```

90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL *'Montiere Schraube links*
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1 *'Neue Schraube*
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR *'Montiere Schraube rechts*
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100

270 NeuerI6kt = 1 *'Neue Schraube*
280 MOV PI6kt
290 C_Close = 1
300 JOVRD 10
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV *'Montiere Schraube vorne*
350 C_Close = 0
360 JOVRD 100

370 NeuerI6kt = 0
380 NeuerI6kt = 1 *'Neue Schraube*
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH *'Montiere Schraube hinten*
460 NeuerI6kt = 0
470 C_Close = 0

480 NeuerI6kt = 1 *'Neue Schraube*
490 NeuerI6kt = 0

500 END

SchraubenAD130

10 REM * Montieren der Schrauben *
20 REM * am Abschlussdeckel *
30 REM * bei Zylinderrohr 130 *

40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11

60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL *'Montiere Schraube links*
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1 *'Neue Schraube*
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR *'Montiere Schraube rechts*
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100

270 NeuerI6kt = 1 *'Neue Schraube*
280 MOV PI6kt
290 C_Close = 1
300 JOVRD 10
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV *'Montiere Schraube vorne*
350 C_Close = 0
360 JOVRD 100

370 NeuerI6kt = 0
380 NeuerI6kt = 1 *'Neue Schraube*
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10

```
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH 'Montiere Schraube hinten
460 JOVRD 10
470 NeuerI6kt = 0
480 C_Close = 0

490 NeuerI6kt = 1 'Neue Schraube
500 NeuerI6kt = 0

510 END
```

SchraubenAD150

```
10 REM * Montieren der Schrauben *
20 REM * am Abschlussdeckel *
30 REM * bei Zylinderrohr 150 *

40 DEF IO C_Close = Bit,5
50 DEF IO NeuerI6kt = Bit,11

60 JOVRD 100
70 MOV PI6kt
80 C_Close = 1
90 JOVRD 10
100 MOV P3
110 MOV P4
120 JOVRD 1
130 MOV PMontSRL 'Montiere Schraube links
140 C_Close = 0
150 JOVRD 100

160 NeuerI6kt = 1 'Neue Schraube
170 MOV PI6kt
180 C_Close = 1
190 JOVRD 10
200 MOV P3
210 MOV P5
220 JOVRD 1
230 MOV PMontSRR 'Montiere Schraube rechts
240 NeuerI6kt = 0
250 C_Close = 0
260 JOVRD 100

270 NeuerI6kt = 1 'Neue Schraube
280 MOV PI6kt
290 C_Close = 1
```

```
300 JOVRD 10
310 MOV P3
320 MOV P6
330 JOVRD 1
340 MOV PMontSRV 'Montiere Schraube vorne
350 C_Close = 0
360 JOVRD 100
```

```
370 NeuerI6kt = 0
380 NeuerI6kt = 1 'Neue Schraube
390 MOV PI6kt
400 C_Close = 1
410 JOVRD 10
420 MOV P3
430 MOV P7
440 JOVRD 1
450 MOV PMontSRH 'Montiere Schraube hinten
460 JOVRD 10
470 NeuerI6kt = 0
480 C_Close = 0
```

```
490 NeuerI6kt = 1 'Neue Schraube
500 NeuerI6kt = 0
```

```
510 END
```

ZYL6x80

```
10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM * 6x80 *
```

```
40 DEF IO C_Close = Bit,5
50 DEF IO ZylPos = Bit,19
60 DEF IO ZylKlemm = Bit,20
70 DEF IO NeuerI6kt = Bit,11
80 DEF IO ZylMit = Bit,21
```

```
90 ZylPos = 1 'ZR-Ablage fährt hoch
100 MOV PZR110 'Hole Zylinderrohr
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR 'Montagepos ZR
180 C_Close = 0
```

```
190 JOVRD 10
200 MOV PADmD 'Hole Abschlussdeckel
210 ZylMit = 1 'Schalte Zyl-Mitnahme ein
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1

250 C_Close = 1
260 MOV P12
270 MOV P13
280 JOVRD 3
290 MOV PMontAD 'Montiere AD
300 JOVRD 10
310 C_Close = 0

320 CALLP "SchraubenAD110"

330 JOVRD 10
340 ZylKlemm = 0
350 ZylPos = 1 'Wende Zylinderrohr
360 MOV PMontZR
370 WAIT M_IN(18) = 1
380 ZylMit = 0
390 C_Close = 1
400 DLY 1
410 MOV P2
420 MOV PDreh
430 MOV PDrehM
440 C_Close = 0
450 ZylMit = 1
460 ZylPos = 0
470 WAIT M_IN(19) = 1
480 ZylKlemm = 1
490 JOVRD 100
500 MOV PKSmK140 'Hole Kolbenstange mit Kolben
510 C_Close = 1
520 JOVRD 10
530 MOV P9
540 MOV P10
550 JOVRD 2
560 MOV PFett 'Kolben in Fett tauchen
570 JOVRD 6
580 MOV P10
590 MOV P11
600 OVRD 1
610 MVS PMontKS 'Montiere KSmK
620 OVRD M_NOVRD
630 C_Close = 0
```

```
640 JOVRD 100
650 MOV PLDmD6 'Hole Lagerdeckel mit Dichtung
660 C_Close = 1
670 JOVRD 10
680 MOV P14
690 MOV P15
700 JOVRD 1
710 MOV PMontLD 'Montiere LDmD
720 JOVRD 10
730 C_Close = 0

740 CALLP "SchraubenLD110"

750 JOVRD 10
760 ZylKlemm = 0
770 ZylPos = 1
780 MOV PMontZR 'Lege fertigen Zylinder ins Lager
790 WAIT M_IN(18) = 1
800 ZylMit = 0
810 C_Close = 1
820 DLY 1
830 MOV P2
840 MOV P16
850 MOV PZyl6x80 'Lagerpos Zyl 6x80
860 C_Close = 0
870 JOVRD 100
880 MOV PStart
890 JOVRD M_NJOVRD

900 END
```

ZYL6x100

```
10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM * 6x100 *'ZR-Ablage fährt hoch
100 MOV PZR130 'Hole Zylinderrohr
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
```

```
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR 'Montagepos ZR
180 C_Close = 0
190 JOVRD 10
200 MOV PADmD 'Hole Abschlussdeckel
210 ZylMit = 1 'Schalte Zyl-Mitnahme ein
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1

250 C_Close = 1
260 MOV P12
270 MOV P13
280 JOVRD 3
290 MOV PMontAD 'Montiere AD
300 JOVRD 10
310 C_Close = 0

320 CALLP "SchraubenAD130"

330 JOVRD 10
340 ZylKlemm = 0
350 ZylPos = 1 'Wende Zylinderrohr
360 MOV PMontZR
370 WAIT M_IN(18) = 1
380 ZylMit = 0
390 C_Close = 1
400 DLY 1
410 MOV P2
420 MOV PDreh
430 MOV PDrehM
440 C_Close = 0
450 ZylMit = 1
460 ZylPos = 0
470 WAIT M_IN(19) = 1
480 ZylKlemm = 1
490 JOVRD 100
500 MOV PKSmK165 'Hole Kolbenstange mit Kolben
510 C_Close = 1
520 JOVRD 10
530 MOV P9
540 MOV P10
550 JOVRD 2
560 MOV PFett 'Kolben in Fett tauchen
570 JOVRD 6
580 MOV P10
```

```

590 MOV P11
600 OVRD 1
610 MVS PMontKS 'Montiere KSmK
620 OVRD M_NOVRD
630 C_Close = 0
640 JOVRD 100
650 MOV PLDmD6 'Hole Lagerdeckel mit Dichtung
660 C_Close = 1
670 JOVRD 10
680 MOV P14
690 MOV P15
700 JOVRD 1
710 MOV PMontLD 'Montiere LDmD
720 JOVRD 10
730 C_Close = 0

740 CALLP "SchraubenLD130"

750 JOVRD 10
760 ZylKlemm = 0
770 ZylPos = 1
780 MOV PMontZR 'Lege fertigen Zylinder ins Lager
790 WAIT M_IN(18) = 1
800 ZylMit = 0
810 C_Close = 1
820 DLY 1
830 MOV P2
840 MOV P16
850 MOV PZy6x100 'Lagerpos Zyl 6x100
860 C_Close = 0
870 JOVRD 100
880 MOV PStart
890 JOVRD M_NJOVRD

900 END

```

ZYL6x120

```

10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM * 6x120 *

40 DEF IO C_Close = Bit,5
50 DEF IO ZylPos = Bit,19
60 DEF IO ZylKlemm = Bit,20
70 DEF IO NeuerI6kt = Bit,11
80 DEF IO ZylMit = Bit,21

```

90 ZylPos = 1 *'ZR-Ablage fährt hoch*
100 MOV PZR150 *'Hole Zylinderrohr*
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR *'Montagepos ZR*
180 C_Close = 0
190 JOVRD 100
200 MOV PADmD *'Hole Abschlussdeckel*
210 ZylMit = 1 *'Schalte Zyl-Mitnahme ein*
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1
250 JOVRD 10
260 C_Close = 1
270 MOV P12
280 MOV P13
290 JOVRD 3
300 MOV PMontAD *'Montiere AD*
310 JOVRD 10
320 C_Close = 0

330 CALLP "SchraubenAD150"

340 JOVRD 10
350 ZylKlemm = 0
360 ZylPos = 1 *'Wende Zylinderrohr*
370 MOV PMontZR
380 WAIT M_IN(18) = 1
390 ZylMit = 0
400 C_Close = 1
410 DLY 1
420 MOV P2
430 MOV PDreh
440 MOV PDrehM
450 C_Close = 0
460 ZylMit = 1
470 ZylPos = 0
480 WAIT M_IN(19) = 1
490 ZylKlemm = 1
500 JOVRD 100
510 MOV PKSmK185 *'Hole Kolbenstange mit Kolben*
520 C_Close = 1
530 JOVRD 10

```

540 MOV P9
550 MOV P10
560 JOVRD 2
570 MOV PFett      'Kolben in Fett tauchen
580 JOVRD 6
590 MOV P10
600 MOV P11
610 OVRD 1
620 MVS PMontKS  'Montiere KSmK
630 OVRD M_NOVRD
640 C_Close = 0
650 JOVRD 100
660 MOV PLDmD6  'Hole Lagerdeckel mit Dichtung
670 C_Close = 1
680 JOVRD 10
690 MOV P14
700 MOV P15
710 JOVRD 1
720 MOV PMontLD  'Montiere LDmD
730 JOVRD 10
740 C_Close = 0

```

```
750 CALLP "SchraubenLD150"
```

```

760 JOVRD 10
770 ZylKlemm = 0
780 ZylPos = 1
790 MOV PMontZR  'Lege fertigen Zyl ins Lager
800 WAIT M_IN(18) = 1
810 ZylMit = 0
820 C_Close = 1
830 DLY 1
840 MOV P2
850 MOV P16
860 MOV PZy6x120 'Lagerpos Zyl 6x120
870 C_Close = 0
880 JOVRD 100
890 MOV PStart
900 JOVRD M_NJOVRD
910 END

```

ZYL8x80

```

10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM *      8x80      *

```

```
40 DEF IO C_Close = Bit,5
```

```
50 DEF IO ZylPos = Bit,19
60 DEF IO ZylKlemm = Bit,20
70 DEF IO NeuerI6kt = Bit,11
80 DEF IO ZylMit = Bit,21

90 ZylPos = 1      'ZR-Ablage fährt hoch
100 MOV PZR110   'Hole Zylinderrohr
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR 'Montage ZR
180 C_Close = 0
190 JOVRD 10
200 MOV PADmD   'Hole Abschlussdeckel
210 ZylMit = 1   'Schalte Zyl-Mitnahme ein
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1
250 C_Close = 1
260 MOV P12
270 MOV P13
280 JOVRD 3
290 MOV PMontAD 'Montiere AD
300 JOVRD 10
310 C_Close = 0

320 CALLP "SchraubenAD110"

330 JOVRD 10
340 ZylKlemm = 0
350 ZylPos = 1   'Wende Zylinderrohr
360 MOV PMontZR
370 WAIT M_IN(18) = 1
380 ZylMit = 0
390 C_Close = 1
400 DLY 1
410 MOV P2
420 MOV PDreh
430 MOV PDrehM
440 C_Close = 0
450 ZylMit = 1
460 ZylPos = 0
470 WAIT M_IN(19) = 1
480 ZylKlemm = 1
490 JOVRD 100
```

```
500 MOV PKSmK140 'Hole Kolbenstange mit Kolben
510 C_Close = 1
520 JOVRD 10
530 MOV P9
540 MOV P10
550 JOVRD 2
560 MOV PFett 'Kolben in Fett tauchen
570 JOVRD 6
580 MOV P10
590 MOV P11
600 OVRD 1
610 MVS PMontKS 'Montiere KSmK
620 OVRD M_NOVRD
630 C_Close = 0
640 JOVRD 100
650 MOV PLDmD8 'Hole Lagerdeckel mit Dichtung
660 C_Close = 1
670 JOVRD 10
680 MOV P14
690 MOV P15
700 JOVRD 1
710 MOV PMontLD 'Montiere LDmD
720 JOVRD 10
730 C_Close = 0

740 CALLP "SchraubenLD110"

750 JOVRD 10
760 ZylKlemm = 0
770 ZylPos = 1
780 MOV PMontZR 'Lege fertigen Zylinder ins Lager
790 WAIT M_IN(18) = 1
800 ZylMit = 0
810 C_Close = 1
820 DLY 1
830 MOV P2
840 MOV P16
850 MOV PZyl8x80 'Lagerpos Zyl 8x80
860 C_Close = 0
870 JOVRD 100
880 MOV PStart
890 JOVRD M_NJOVRD
900 END
```

ZYL8x100

```

10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM *      8x100      *

40 DEF IO C_Close = Bit,5
50 DEF IO ZylPos = Bit,19
60 DEF IO ZylKlemm = Bit,20
70 DEF IO NeuerI6kt = Bit,11
80 DEF IO ZylMit = Bit,21

90 ZylPos = 1      'ZR-Ablage fährt hoch
100 MOV PZR130   'Hole Zylinderrohr
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR 'Montagepos ZR
180 C_Close = 0
190 JOVRD 10
200 MOV PADmD   'Hole Abschlussdeckel
210 ZylMit = 1   'Schalte Zyl-Mitnahme ein
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1

250 C_Close = 1
260 MOV P12
270 MOV P13
280 JOVRD 3
290 MOV PMontAD 'Montiere AD
300 JOVRD 10
310 C_Close = 0

320 CALLP "SchraubenAD130"

330 JOVRD 10
340 ZylKlemm = 0
350 ZylPos = 1   'Wende Zylinderrohr
360 MOV PMontZR
370 WAIT M_IN(18) = 1
380 ZylMit = 0
390 C_Close = 1
400 DLY 1
410 MOV P2

```

```
420 MOV PDreh
430 MOV PDrehM
440 C_Close = 0
450 ZylMit = 1
460 ZylPos = 0
470 WAIT M_IN(19) = 1
480 ZylKlemm = 1
490 JOVRD 100
500 MOV PKSmK165 'Hole Kolbenstange mit Kolben
510 C_Close = 1
520 JOVRD 10
530 MOV P9
540 MOV P10
550 JOVRD 2
560 MOV PFett 'Kolben in Fett tauchen
570 JOVRD 6
580 MOV P10
590 MOV P11
600 OVRD 1
610 MVS PMontKS 'Montiere KSmK
620 OVRD M_NOVRD
630 C_Close = 0
640 JOVRD 100
650 MOV PLDmD8 'Hole Lagerdeckel mit Dichtung
660 C_Close = 1
670 JOVRD 10
680 MOV P14
690 MOV P15
700 JOVRD 1
710 MOV PMontLD 'Montiere LDmD
720 JOVRD 10
730 C_Close = 0

740 CALLP "SchraubenLD130"

750 JOVRD 10
760 ZylKlemm = 0
770 ZylPos = 1
780 MOV PMontZR 'Lege fertigen Zylinder ins Lager
790 WAIT M_IN(18) = 1
800 ZylMit = 0
810 C_Close = 1
820 DLY 1
830 MOV P2
840 MOV P16
850 MOV PZy8x100 'Lagerpos Zyl 8x100
860 C_Close = 0
870 JOVRD 100
```

```
880 MOV PStart
890 JOVRD M_NJOVRD
```

```
900 END
```

ZYL8x120

```
10 REM * Endmontage des *
20 REM * Pneumatikzylinders *
30 REM *      8x120      *
```

```
40 DEF IO C_Close = Bit,5
50 DEF IO ZylPos = Bit,19
60 DEF IO ZylKlemm = Bit,20
70 DEF IO NeuerI6kt = Bit,11
80 DEF IO ZylMit = Bit,21
```

```
90 ZylPos = 1      'ZR-Ablage fährt hoch
100 MOV PZR150   'Hole Zylinderrohr
110 C_Close = 1
120 WAIT M_IN(18) = 1
130 JOVRD 10
140 MOV P1
150 MOV P2
160 JOVRD 2
170 MOV PMontZR 'Montage ZR
180 C_Close = 0
190 JOVRD 100
200 MOV PADmD   'Hole Abschlussdeckel
210 ZylMit = 1   'Schalte Zyl-Mitnahme ein
220 ZylPos = 0
230 WAIT M_IN(19) = 1
240 ZylKlemm = 1
```

```
250 C_Close = 1
260 JOVRD 10
270 MOV P12
280 MOV P13
290 JOVRD 3
300 MOV PMontAD 'Montiere AD
310 JOVRD 10
320 C_Close = 0
```

```
330 CALLP "SchraubenAD150"
```

```
340 JOVRD 10
350 ZylKlemm = 0
360 ZylPos = 1      'Wende Zylinderrohr
```

```
370 MOV PMontZR
380 WAIT M_IN(18) = 1
390 ZylMit = 0
400 C_Close = 1
410 DLY 1
420 MOV P2
430 MOV PDreh
440 MOV PDrehM
450 C_Close = 0
460 ZylMit = 1
470 ZylPos = 0
480 WAIT M_IN(19) = 1
490 ZylKlemm = 1
500 JOVRD 100
510 MOV PKSmK185 'Hole Kolbenstange mit Kolben
520 C_Close = 1
530 JOVRD 10
540 MOV P9
550 MOV P10
560 JOVRD 2
570 MOV PFett 'Kolben in Fett tauchen
580 JOVRD 6
590 MOV P10
600 MOV P11
610 OVRD 1
620 MVS PMontKS 'Montiere KSmK
630 OVRD M_NOVRD
640 C_Close = 0
650 JOVRD 100
660 MOV PLDmD8 'Hole Lagerdeckel mit Dichtung
670 C_Close = 1
680 JOVRD 10
690 MOV P14
700 MOV P15
710 JOVRD 1
720 MOV PMontLD 'Montiere LDmD
730 JOVRD 10
740 C_Close = 0

750 CALLP "SchraubenLD150"

760 JOVRD 10
770 ZylKlemm = 0
780 ZylPos = 1
790 MOV PMontZR 'Lege fertigen Zylinder ins Lager
800 WAIT M_IN(18) = 1
810 ZylMit = 0
820 C_Close = 1
```

```
830 DLY 1
840 MOV P2
850 MOV P16
860 MOV PZy8x120 'Lagerpos Zyl 8x120'
870 C_Close = 0
880 JOVRD 100
890 MOV PStart
900 JOVRD M_NJOVRD

910 END
```

10 Literaturverzeichnis

Backhauf, L.: Systematische Betriebsvorbereitung zur Verkürzung der Einlaufkurven von Chemieanlagen, Magdeburg, 1982, zit. nach: Weber, Klaus H.: Inbetriebnahme verfahrenstechnischer Anlagen. Praxishandbuch mit Checklisten und Beispielen, 2. Auflage, Berlin/Heidelberg, 2002

Bayer, Johann / Collisi, Thomas / Wenzel, Sigrid: Einleitung, in: Bayer, Johann u.a. (Hrsg.): Simulation in der Automobilproduktion, Berlin/Heidelberg, 2003, S. 1-6

Bernecker, G.: Planung und Bau verfahrenstechnischer Anlagen. Projektmanagement und Fachplanungsfunktionen, 3. Auflage, Düsseldorf, 1984, zit. nach: Weber, Klaus H.: Inbetriebnahme verfahrenstechnischer Anlagen. Praxishandbuch mit Checklisten und Beispielen, 2. Auflage, Berlin / Heidelberg, 2002

Breitenbach, Frank / Schwab, Joachim: Virtuelle Methoden und Prozesse für das Anlaufmanagement, in: Zäh, Michael F. u.a. (Hrsg.): Virtuelle Produktionssystemplanung. Virtuelle Inbetriebnahme und Digitale Fabrik, Garching, 2004, Kap. 2, S.1-19

Brychta, Peter / Müller, Kai: Technische Simulation, Würzburg, 2004

Griesbach, Bernd / Herzog, Frank / Ehrenstraßer, Michael: Virtuelle Betriebsmittelerstellung im digitalen Werkzeugbau, in: Zäh, Michael F. u.a. (Hrsg.): Virtuelle Produktionssystemplanung. Virtuelle Inbetriebnahme und Digitale Fabrik, Garching, 2004, Kap. 4, S. 1-27

Kremer, Axel / Liewald, Jörg: Perspektiven für die Simulation, in: Bayer, Johann u.a. (Hrsg.): Simulation in der Automobilproduktion, Berlin/Heidelberg, 2003, S. 151-209

Müller, Willi / Hrdliczka, Veronika: Statements und Trendaussagen, in: Bayer, Johann u.a. (Hrsg.): Simulation in der Automobilproduktion, Berlin/Heidelberg, 2003, S. 213-214

Spieckermann, Sven / Wortmann, Dirk: Die Simulation von Lackieranlagen in der Automobilindustrie, in: Bayer, Johann u.a. (Hrsg.): Simulation in der Automobilproduktion, Berlin/Heidelberg, 2003, S. 47-59

VDI: Simulation von Logistik-, Materialfluss- und Produktionssystemen. In VDI-Richtlinien 3633, VDI (Hrsg.), Berlin

Blatt 1, Grundlagen, 2000 (Entwurf)

Blatt 5, Integration der Simulation in die betrieblichen Abläufe, 2000

Blatt 11, Simulation und Visualisierung, 2003 (Entwurf)

Digitale Fabrik. Grundlagen. In VDI-Richtlinie 4499, VDI (Hrsg.), Berlin

Blatt 1, Grundlagen, 2008

Zäh, Michael F. u.a.: Virtuelle Inbetriebnahme im Regelkreis des Fabriklebenszyklus', in: Zäh, Michael F. u.a. (Hrsg.): Virtuelle Produktionssystemplanung. Virtuelle Inbetriebnahme und Digitale Fabrik, Garching, 2004, Kap. 1, S. 1-21