Zukunftsfabrik Bodensee: Bin Picking für das Produktionsnetzwerk

STUDIENARBEIT T3200

des Studienganges Elektrotechnik (Automation)

an der Dualen Hochschule Baden-Württemberg Campus Friedrichshafen

von

Alexander Deutsch, 9301310

Tarek Zwick, 1857672

17.07.2023

Bearbeitungszeitraum	14 Wochen
Kurs	TEA20
Gutachter der Dualen Hochschule	M. Sc. Wenzheng Cai

Erklärung

Gemäß Ziffer 1.1.13 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden-Württemberg vom 29.09.2017 in der Fassung vom 25.07.2018:

Wir versichern hiermit, dass wir unsere Studienarbeit T3200 mit dem Thema:

Zukunftsfabrik Bodensee: Bin Picking für das Produktionsnetzwerk

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben. Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Friedrichshafen, den 17.07.2023

Alexander Deutsch
Unterschrift

Tarek Zwick
Unterschrift

Inhaltsverzeichnis

1. Einleitung 1
   1.1. Vorstellung des Produktionsnetzwerkes 1
   1.2. Zusammenfassung der bisherigen Ergebnisse 2
2. Aufgabenstellung 5
3. Stand der Technik 6
   3.1. Kollaborative Roboter 6
   3.2. Bin Picking 8
      3.2.1. Roboter 9
      3.2.2. Greifer 9
      3.2.3. Kamera 10
      3.2.4. Software 11
   3.3. Aufbau der Bin-Picking-Station 12
   3.4. Mikado Softwarepaket 13
      3.4.1. Mikado Assistant 13
      3.4.2. Mikado 15
      3.4.3. Mikado Simulator 16
   3.5. Der Bin-Picking-Prozess in Mikado 16
   3.6. Grundlagen der PROFINET-Kommunikation durch Mikado 18
   3.7. Digitaler Zwilling 19
4. Projektplanung 21
5. Digitaler Zwilling 22
6. Integration in den Produktionsprozess 24
   6.1. Die Wahl des Greifers 24
   6.2. Anforderungen 28
   6.3. Erstellung einer Projektkonfiguration im Mikado Assistant 29
      6.3.1. Aufbau eines Mikado Projekts 29
      6.3.2. Duplizieren einer Mikado Assistant Projektkonfiguration 30
   6.4. Erstellung des Mikado-Programms 31
   6.5. Probleme und deren Lösung 36
   6.6. Anbindung an die SPS 38
7. Machine Learning 40
8. Fazit und Ausblick 41
9. Literaturverzeichnis 43

Abkürzungsverzeichnis

AutoML ..... automatisiertes Machine Learning
CAD ..... Computer-Aided-Design
DHBW ..... Duale Hochschule Baden-Württemberg
EN ..... Europäische Norm
GSDML ..... Generic Station Description Markup Language
IO ..... Input/Output
IWT ..... Institut für Weiterbildung, Wissens- und Technologietransfer
MCD ..... Mechatronic Concept Design
PROFINET ..... Process Field Network
PTZ ..... Produktionstechnisches Zentrum
QR-Code ..... Quick-Response-Code
ROI ..... Region of Interest
SPS ..... speicherprogrammierbare Steuerung
TCP ..... Tool Center Point
TCP/IP ..... Transmission Control Protocol/Internet Protocol

Abbildungsverzeichnis

Abbildung 1: Prozessablauf im PTZ 2
Abbildung 2: Bin-Picking-Station nach dem Umbau 4
Abbildung 3: Verlauf und Prognose des weltweiten Absatzes von Industrierobotern 6
Abbildung 4: Prognose des weltweiten Umsatzes von Cobots in Mio. USD 7
Abbildung 5: Bin Picking Symbolbild 8
Abbildung 6: Mechanische Greifer 9
Abbildung 7: Zu greifender Bolzen 12
Abbildung 8: Universal Robots UR3 12
Abbildung 9: Signalverdrahtung der Bin-Picking-Station 13
Abbildung 10: Belegung der PROFINET-Wörter im Mikado Assistant 18
Abbildung 11: Prognose des weltweiten Marktvolumens an digitalen Zwillingen 20
Abbildung 12: Gantt-Diagramm zur Projektüberwachung 21
Abbildung 13: Digitales Abbild der Bin-Picking-Station im MTpro Online Designer 22
Abbildung 14: Digitaler Zwilling der Bin-Picking-Station 23
Abbildung 15: Digitaler Zwilling des Produktionsnetzwerks im PTZ 23
Abbildung 16: Pneumatischer Vakuum-Endeffektor 24
Abbildung 17: Magnetventil 25
Abbildung 18: Konfiguration des Magnetventils mit Druckregler (markiert) 25
Abbildung 19: 3D-Modell der neuen Greifbacken 27
Abbildung 20: 2-Backen-Greifer, Rillen markiert 27
Abbildung 21: Konfiguration des Mikado-Produkts "Bolzen" 31
Abbildung 22: Griff von oben 32
Abbildung 23: Griff von der Seite 32
Abbildung 24: Altes Programmflussdiagramm 33
Abbildung 25: Neues Programmflussdiagramm 34
Abbildung 26: Flussdiagramm des Programmelements „Bolzen platzieren“ 35
Abbildung 27: Erfolgreiches Platzieren der Bolzen 39

1. Einleitung

1.1. Vorstellung des Produktionsnetzwerkes

Das Produktionsnetzwerk Fallenbrunnen entsteht in Zusammenarbeit zwischen der DHBW Ravensburg am Campus Friedrichshafen und dem Institut für Weiterbildung, Wissens- und Technologietransfer (IWT), das ebenfalls am Standort Fallenbrunnen vertreten ist. [1]

Das produktionstechnische Zentrum (PTZ) an der DHBW stellt ein Labor für Industrielle Messtechnik und für Automation dar. Hier finden diverse Laboraktivitäten zu Datenaufnahme und -verarbeitung sowie kollaborativer Robotik statt. Die Lernfabrik des IWT Bodensee verfügt über eine vielfältige Ausrüstung in Bezug auf Roboter, Bandsysteme und 3D-Drucker. Gemeinsam wird ein Anwendungsfall geschaffen, in dem spannende Themen der Industrie 4.0-Technologie gelehrt, geforscht und umgesetzt werden können. [1]

Ziel des Produktionsnetzwerkes ist die Abbildung einer realen Wertschöpfungskette mit vernetzten, adaptiven und selbstlernenden Systemen. Im Produktionsnetzwerk soll die Produktion eines Produktes in hoher Variantenvielfalt stattfinden, anschließend soll zur Abbildung einer Kreislaufwirtschaft ebenfalls eine vollständige Demontage ermöglicht werden. [1]

Das Projektteam umfasst 23 Studierende, vier Professoren, vier Laboringenieure, drei wissenschaftliche Mitarbeiter und drei Doktoranden, die das Produktionsnetzwerk in zwölf Teilprojekten entwickeln. [1]

Das Großprojekt startet mit dem Beginn der Studienarbeiten im Oktober 2022 und soll im Juli 2023 vollständig produktionsfähig sein.

Im PTZ werden fünf Stationen aufgebaut und über ein Transfersystem verbunden. Station 1 realisiert eine Pick-and-Place-Station, die die Rohlinge auf das Förderband ablegt. Station 2 platziert je nach Anforderung keinen, einen oder zwei Bolzen in den Bauteilträger. Station 3 stellt einen Handarbeitsplatz dar, an dem die Montage der eingetroffenen Bauteile stattfindet. Station 4 stellt eine kamerabasierte Qualitätsprüfung dar. Nach bestandener Qualitätsprüfung werden die gefertigten Werkstücke über

Station 5 auf ein Transportsystem positioniert, das die Vorprodukte über das Campusgelände zur Lernfabrik transportiert, wo die Endmontage über weitere manuelle und automatisierte Arbeitsschritte erfolgt. Nach wiederholter bestandener Endprüfung wird das Produkt eingelagert, im selben Kreislauf kann dann die Demontage erfolgen.

Der Prozessablauf der im PTZ stattfindenden Produktion ist in folgender Abbildung dargestellt:

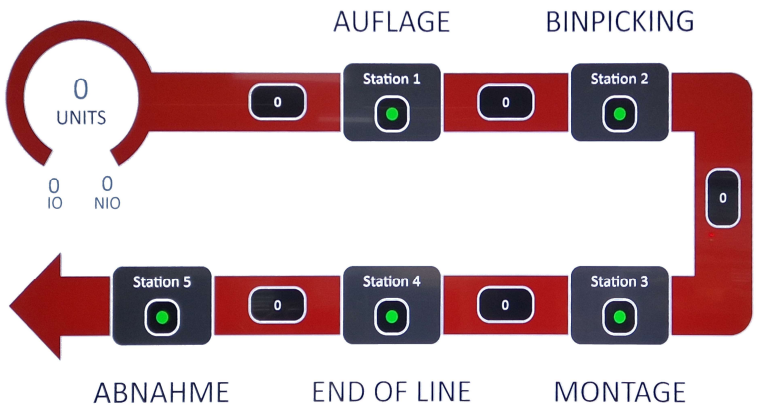


Abbildung 1: Prozessablauf im PTZ

Hierbei stellt Station 2 „Bin-Picking“ den primären Fokus dieser Studienarbeit dar.

1.2. Zusammenfassung der bisherigen Ergebnisse

Die Entwicklung des Produktionsnetzwerkes sowie die zugehörigen Studienarbeiten finden in zwei Blöcken statt. Der erste Block beginnt Anfang Oktober 2022 und endet Ende Dezember 2022 aufgrund der folgenden Praxisphasen der Studierenden. Anfang April 2023 wird die Entwicklung fortgesetzt und Ende Juni 2023 abgeschlossen.

Der erste Block wird in der Studienarbeit T3100 ausgearbeitet. Hier gilt es, sich mit der bisher für Laborversuche und Ausstellungen verwendeten Bin-Picking-Station vertraut zu machen und diese für die Integration in das Produktionsnetzwerk vorzubereiten.

Kern der Bin-Picking-Station ist ein kollaborativer Roboter des Typs UR3 des Herstellers Universal Robots. Mithilfe einer tiefenwahrnehmenden Kamera und einem Endeffektor mit zwei Greifbacken, kann dieser den Bin-Picking-Prozess ausführen. Hierbei

greift der Roboter Werkstücke aus einer Kiste platziert sie in andere Behälter. Aufgrund des senkrecht verlaufenden Strebenprofils, das als Kamerahalterung fungiert, kann die Bin-Picking-Station nicht sinnvoll in das Produktionsnetzwerk eingebunden werden, da der kollaborative Roboter hierdurch stark in seiner Bewegungsfreiheit eingeschränkt wird. Daher muss die Kamerahalterung in einer anderen Ausführung realisiert werden, allerdings ist darauf zu achten, dass die Bin-Picking-Station nach wie vor kompatibel mit dem Laborversuch bleibt.

Hierfür wird die Kamerahalterung an eine der anderen Ecken der Arbeitsplatte montiert und so angepasst, dass die Kamera selbst die gleiche Position über der Arbeitsplatte wie vor dem Umbau aufweist.

Aufgrund von auf der Hinterseite befestigten Kabelkanälen kann die Bin-Picking-Station nicht auf Anschlag an das Transfersystem geschoben werden, sondern lediglich in angeschrägter Ausführung. Um dem entgegenzuwirken, werden zwei Abstandshalter auf der Rückseite befestigt, die weiter als die besagten Kabelkanäle herausragen. Diese stoßen an dem Gerüst des Transfersystems an und garantieren so eine parallele Anbindung. Allerdings muss die Position entlang des Transfersystems fixiert bleiben und reproduzierbar sein, damit die Bin-Picking-Station hohe Erfolgsquoten erreichen kann. Hierfür werden am Gerüst des Transfersystems zwei herausragende Profilstreben angebracht, die als Führungsschienen dienen. Die Bin-Picking-Station wird zwischen diese Schienen bis auf den Anschlag an das Transfersystem eingeschoben und verfügt so lediglich über wenige Millimeter Toleranz bezüglich der absoluten Position. Um die Station vor Bewegung zu schützen, werden an die hinzugefügten Abstandshalter und Führungsschienen Spannverschlüsse montiert, um die Station über vier Punkte in ihrer Position zu fixieren. Durch das Festspannen bewegt sich die Bin-Picking-Station in die Mitte des möglichen Toleranzbereiches entlang des Transfersystems, sodass die hier noch mögliche Toleranz auf unter einen Millimeter fällt, was für eine prozesssichere Ausführung des Bin Pickings ausschlaggebend ist.

Nach einem Austausch der Rollen für eine optimierte Handhabung ist die Bin-Picking-Station aus mechanischer Sicht vollständig für die Integration in das Produktionsnetzwerk umgerüstet.

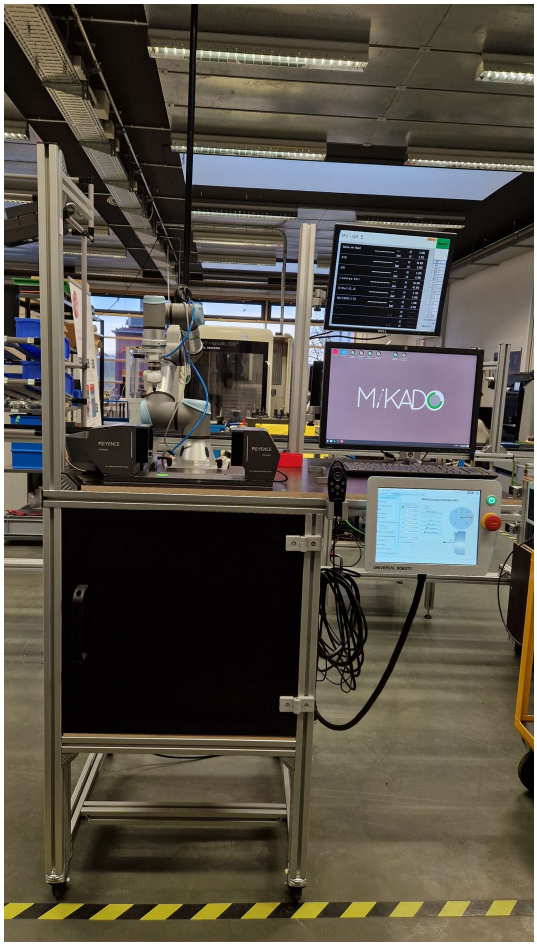


Abbildung 2: Bin-Picking-Station nach dem Umbau

Anschließend wird sich mit der Robotersteuerungssoftware vertraut gemacht und ein Konzept ausgearbeitet, das den Programmablauf für das Produktionsnetzwerk schildert. Hierbei werden erste Abhängigkeiten von anderen Projektgruppen deutlich, zum Beispiel in den Fragen, wie die Bin-Picking-Station erfährt, welche der vielen Varianten des Produkts benötigt wird oder wie erkannt wird, dass ein Bauteilträger die Station erreicht hat, sodass diese ihren Prozessablauf starten kann. Diese Fragen gilt es, bis zur nächsten Projektphase zu klären.

Zusätzlich wird sich mit Machine-Learning Verfahren beschäftigt, da das Erkennen und Greifen unbekannter Teile realisiert werden soll. Der Unterschied zur Integration in das Produktionsnetzwerk besteht darin, dass hierfür jederzeit baugleiche Bolzen gegriffen werden. Die unbekannten Teile weisen allerdings unterschiedliche Dimensionierungen auf, für deren Erkennung ein zusätzlicher, komplexerer Algorithmus benötigt wird. Hierfür wird bezüglich des Machine-Learnings sowie dessen verschiedenen Unterarten recherchiert und sich mit möglichen Ausarbeitungen und Problemstellungen vertraut gemacht wird.

2. Aufgabenstellung

Die Bin-Picking-Station soll in den Gesamtprozess des PTZs integriert werden und wird dafür an einem Förderband als Transfersystem platziert. Auf diesem Förderband laufen in regelmäßigen Abständen Werkstücke entlang und halten für eine kurze Zeit bei der Station an. Jedes dieser Werkstücke ist durch einen Quick-Response-Code (QR-Code) gekennzeichnet, der nach Platzieren durch die vorangehende Pick-and-Place-Station auf dem Weg zur Bin-Picking-Station ausgelesen wird. Die Aufgabe des kollaborativen Roboters ist es, aus einem Behälter Bolzen zu greifen und diesen im Werkstückträger abzulegen. Die Anzahl und die Position der zu platzierenden Bolzen wird über den QR-Code bestimmt, hierbei muss der Roboter keinen, einen oder zwei Bolzen einsetzen.

Für die Integration der Bin-Picking-Station in den Prozessablauf des Produktionsnetzwerkes muss für die Steuerung der Station ein Programm entworfen, das den geforderten Produktionsablauf schnell und prozesssicher durchlaufen kann. Hierfür soll anstelle der bisher verwendeten Greifbacken ein elektrisch angesteuerter Vakuumgreifer verwendet werden, der die Bolzen aus dem Behälter ansaugt. Um dies zu ermöglichen, muss die Bin-Picking Station in das Kommunikationssystem des Produktionsnetzwerks eingebunden und mit dessen speicherprogrammierbaren Steuerung (SPS) verbunden werden.

Parallel zur Entwicklung des Produktionsnetzwerkes soll ein digitaler Zwilling entstehen, der die Prozessabläufe auf digitaler Ebene festhält. Hiermit kann das Produktionsnetzwerk bereits vor seiner abgeschlossenen Entwicklung getestet werden und mögliche Probleme an mechanischen Schnittstellen wie bspw. Bewegungsgrenzen der kollaborativen Roboter vorzeitig erkannt werden. Hierfür muss die gesamte Station in dem Programm „Mechatronic Concept Designer“ nachgebildet werden.

Darüber hinaus soll ein Machine Learning Algorithmus entwickelt werden, mit dessen Hilfe nicht nur bekannte Bolzen, sondern auch unbekannte Bauteile erkannt sowie gegriffen werden können. Der Algorithmus wird von der 3D-Kamera mit einem Bild des Bin-Picking-Behälters, in der sich eine unbekannte Menge unbekannter Bolzen befinden, gefüttert. Der Algorithmus soll dann eine Liste erstellen, in der die Maße aller zu sehenden Bolzen aufgelistet sind sowie einen Punkt finden, an dem der Bolzen gegriffen werden kann.

3. Stand der Technik

3.1. Kollaborative Roboter

Die europäische Norm EN 775 definiert einen Industrieroboter als ein automatisch gesteuertes, wiederprogrammierbares, vielfach einsetzbares Handhabungsgerät mit mehreren Freiheitsgraden, das entweder ortsfest oder beweglich in automatisierten Fertigungssystemen eingesetzt wird. [2]

Der Einsatz von Robotern in der Produktion stieg in der Vergangenheit deutlich an und hat sich in vielen Bereichen der Industrie etabliert. Die nachfolgendende Statistik zeigt den jährlichen Absatz von Industrierobotern ab 2004 und prognostiziert den weiteren Verlauf bis 2025. Bis auf wenige Ausnahmen ist ein kontinuierliches Wachstum zu sehen.

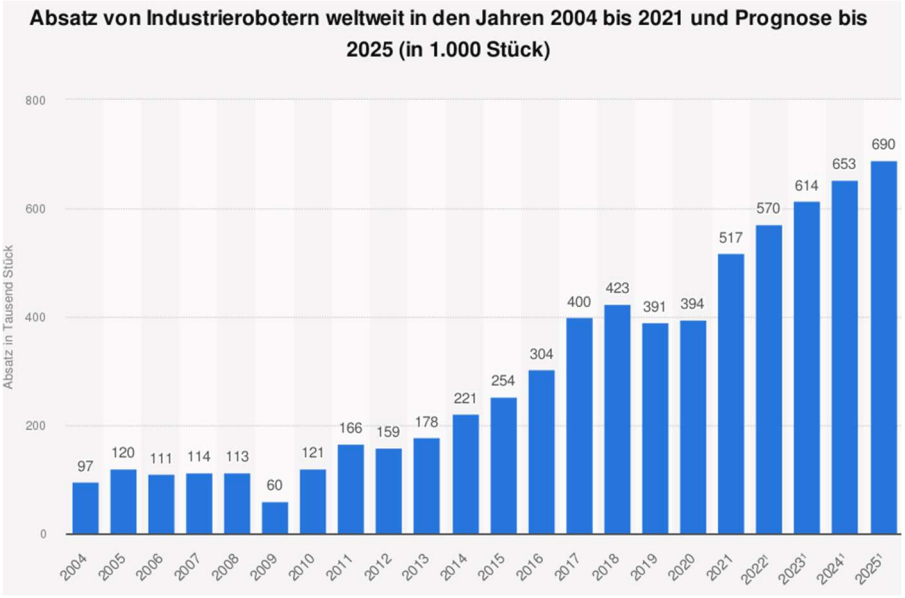


Abbildung 3: Verlauf und Prognose des weltweiten Absatzes von Industrierobotern [3]

Dieses Wachstum geht auf die vielen Vorteile der Verwendung von Industrierobotern zurück. Industrieroboter garantieren eine gleichbleibende Produktionsqualität sowie eine effiziente Produktion, da diese im Vergleich zum menschlichen Körper nicht ermüden, keine Pausen benötigen und eine deutlich geringe Fehleranfälligkeit besitzen. Abzüglich von Wartungs- und Störungszeiten kann so 24h pro Tag produziert werden. Langfristig gesehen werden die Produktionskosten gesenkt, wodurch eine Investition in Industrieroboter den Gewinn steigern kann. [4]

Einen besonderen Typ der Industrieroboter stellen die sogenannten „Cobots“ dar. Das Kofferwort Cobot setzt sich aus den englischen Begriffen „Collaboration“ und „Robot“ zusammen und bezeichnet Roboter, die gemeinsam mit Menschen in einer Arbeitsumgebung Produktionsschritte ausführen. [2]

Kollaborative Roboter gewinnen immer mehr an Aufmerksamkeit und Bedeutung. Durch zunehmend ausgereifte Technologien und steigendem Wettbewerb wächst das Marktvolumen immer weiter. Einer Prognose zufolge (siehe Abbildung 4) wird das jährliche Wachstum des weltweiten Umsatzes von kollaborativen Robotern für die nächsten Jahre über 20% liegen. Somit wird sich dieser innerhalb von acht Jahren zwischen 2018 und 2026 fast vervierfachen und zwei Milliarden USD übersteigen. Dieses Wachstum übertrifft das relative Wachstum der Industrieroboter (vgl. Abbildung 3) deutlich, wodurch der Marktanteil signifikant zunehmen wird.

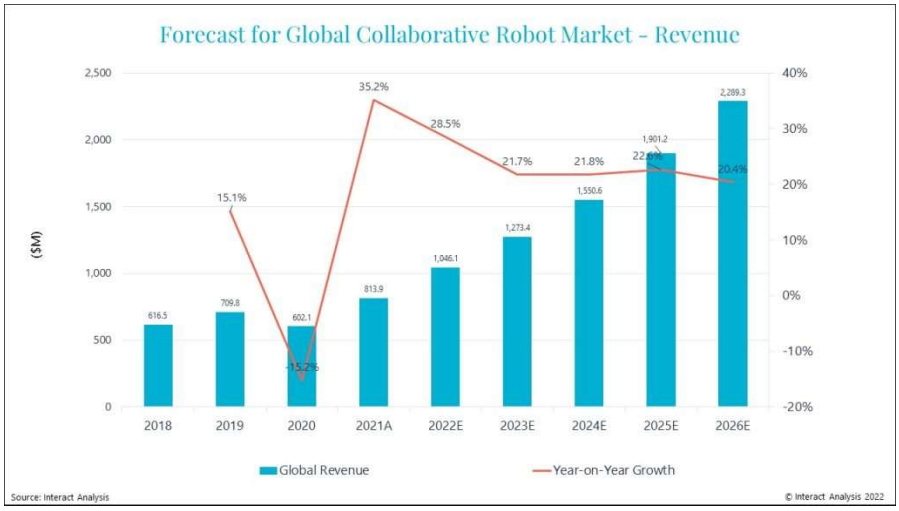


Abbildung 4: Prognose des weltweiten Umsatzes von Cobots in Mio. USD [5]

Cobots unterscheiden sich von herkömmlichen Industrierobotern in mehreren Aspekten. Während herkömmliche Industrieroboter immer mit Schutzeinrichtungen wie Zäunen und Lichtschranken arbeiten, um die Sicherheit von Mitarbeiter zu gewährleisten, arbeiten Cobots ohne trennende Schutzeinrichtung. So können sie gleichzeitig mit einem Mitarbeiter am gleichen Produkt arbeiten, mit der Umwelt interagieren und sich an diese anpassen, während ein herkömmlicher Industrieroboter festgegebene Prozeduren durchführt. Die Anwendungsfälle bei den herkömmlichen Industrierobotern sind immer festvorgegeben, während Cobots diesbezüglich flexibler sind und können leicht angepasst werden, während bei herkömmlichen Industrierobotern eine anwendungsspezifische Software benötigt wird. Zusätzlich dazu fällt der Preis für Cobots geringer aus, was zu einer erhöhten Rentabilität der Investition führt. [6]

3.2. Bin Picking

Der englische Begriff „Bin Picking“ kann zu dem „Griff in die Kiste“ übersetzt werden. Bin Picking ist eine Art der automatisierten Teile-Handhabung, bei der ein Behälter („Bin“) mit gleichen oder verschiedenen ungeordneten Einzelteilen, die daher oftmals auch als Schüttgut bezeichnet werden, bereitgestellt wird. Über eine installierte Kamera erkennt das System Roboter bestimmte, vordefinierte Teile in dem Behälter, aus dem die völlig zufällig angeordneten Werkstücke einzeln herausgegriffen („Picking“) werden und dann in vordefinierte Lagen abgelegt oder einem weiteren Verarbeitungsprozess zugeführt werden. [8]



Abbildung 5: Bin Picking Symbolbild [7]

Der Bin-Picking-Prozess durchläuft dabei die folgenden Arbeitsschritte:

Objekterkennung: Zunächst muss anhand bauteilspezifischer Merkmale erkannt werden, welche Einzelteile sich in der Kiste befinden. Hierfür scannt das Kamerasystem den Behälter und gibt mithilfe mathematischer Verfahren Typ, Position und Lage der sich im Behälter befindenden Einzelteile aus. [8] [9]

Entnahme: Anschließend werden die zuvor gesammelten Daten an die Steuerung des Roboters übertragen, die ausgehend von den übermittelten Daten das zu greifende Einzelteil bestimmt, eine optimale Bahnplanung durchführt sowie eine Greif- und Entnahmestrategie bestimmt. Hier spielt die Kollisionserkennung eine große Rolle, da bei den optimalsten Bahnen oft Hindernisse wie beispielsweise der Behälter selbst vorhanden sind. [8]

Platzierung: Nach erfolgreichem Griff eines Einzelteils gilt es, dieses an einem vordefinierten Punkt abzulegen, das kann beispielsweise ein Förderband, eine Palette oder eine weitere Werkzeugmaschine sein. Hierfür findet erneut eine kollisionsbetrachtete Bahnplanung statt. [8]

Zusätzlich kann eine Objektinspektion in den Gesamtprozess integriert werden. Über externe optische Messmittel ist es möglich, eine Vermessung der gegriffenen Einzelteile zu implementieren. Diese können dann mit Sollmerkmalen und -toleranzen verglichen werden und entsprechend den Kriterien auch aussortiert werden. [9]

Für eine erfolgreiche Anwendung des Bin-Pickings müssen Roboter, Greifer, Kamerasystem und die Software reibungslos harmonieren. Daher bieten viele Hersteller auch ideal abgestimmte Komplettsysteme an. Im Folgenden werden die einzelnen Komponenten sowie deren anwendungsbezogenen Anforderungen genauer untersucht.

3.2.1. Roboter

Es ist wichtig, den prozessdurchführenden Roboter anhand seiner technischen Spezifikationen auszuwählen. Wichtig sind hierbei die maximale Tragfähigkeit sowie der Arbeitsraum des Roboters. Die maximale Tragfähigkeit gibt das maximale Gewicht der Einzelteile, die gegriffen und bewegt werden sollen, an. Der Arbeitsraum des Roboters gibt den Bereich um den Roboter an, der von diesem angefahren werden kann. Die Auswahl des Roboters richtet sich demnach nach der Größe und dem Gewicht des Spektrums an zu greifenden Werkstücken sowie nach der Entfernung, die diese bewegt werden müssen.

3.2.2. Greifer

Der Greifer liegt am Werkzeugflansch des Roboters an und ermöglicht die Bewegung der Einzelteile von einer Position zu einer anderen. Das Design des Greifers spielt bei Bin-Picking-Anwendungen eine große Rolle und muss genau auf den Anwendungsfall spezialisiert werden. [8] Zunächst werden gängige Varianten von Greifern und ihre unterschiedlichen Wirkprinzipien betrachtet.

Bei dem mechanischen Wirkprinzip wird das Werkstück zwischen mehreren Greifbacken eingeklemmt. Hierbei wird zwischen den Winkel- und Parallelgreifern unterschieden. Die Unterscheidung liegt hierbei in der Bewegungsart der Greifbacken. Beim Winkelgreifer schließen diese in einer Rotationsbewegung, während die Greifbacken des Parallelgreifers in einer linearen Bewegung schließen. Die Bewegung wird meist über elektrische Antriebssysteme oder pneumatisch angesteuerte Kolben realisiert. Eine Realisierung ist auch mit mehr als zwei Backen möglich. [10] [11]

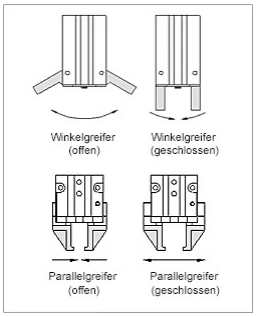


Abbildung 6: Mechanische Greifer [10]

Neben dem mechanischen Wirkprinzip stellt auch das pneumatische Wirkprinzip eine Möglichkeit zum Vereinzeln der Werkstücke dar. Hier wird das Bauteil nicht zwischen Greifbacken eingespannt, sondern mithilfe eines Unterdrucks aus dem Behälter angesaugt. Die Ansteuerung findet pneumatisch statt, hierfür wird ein Überdruck durch den Greifer geleitet und mithilfe des Venturi-Effekts ein Unterdruck erzeugt, der zum Ansaugen der Einzelteile genutzt wird. [12] Voraussetzung hierfür sind allerdings Bauteile mit glatten Oberflächen, da für die Haftung des Bauteils an dem Sauger ein luftdichter Raum zwischen diesen benötigt wird. [11]

Solche Vakuumgreifer gibt es in verschiedenen Ausführungen, dies können Flachsauggreifer, Balgsauggreifer oder auch Flächensauggreifer, die aus der Kombination mehrerer Sauggreifer entstehen, sein. [13]

Ein weiteres, seltener angewandtes Wirkprinzip stellt der Magnetgreifer dar. Hierbei wird der Greifer über das Werkstück gefahren und durch einen Magneten angezogen. Dieses Wirkprinzip ist selbstverständlich nur für ferromagnetische Werkstücke durchführbar. Auf dem Markt gibt es Ausführungen mit Permanent- und Elektromagnetgreifern. Ein Nachteil der Permanentmagnetgreifer besteht darin, dass das Werkstück nicht selbstständig von dem Greifer losgelassen werden kann. Elektromagnetgreifer sind in den meisten Anwendungsfällen praktischer, da der Elektromagnet angesteuert werden kann und das Anziehen und Loslassen des Werkstückes gesteuert werden kann. [11] Gerade bei kleinen Bauteilen ist dieses Wirkprinzip allerdings nicht verwendbar, da die durch den Magnetismus hervorgerufenen Kräfte sich auch auf die umliegenden Bauteile auswirken und so mehrere Einzelteile angezogen werden. Für nicht-ferromagnetische Einzelteile entfällt dieses Wirkprinzip.

3.2.3. Kamera

Die Kamera erfüllt die Aufgabe, ein Bild des Behälters, in dem sich die Einzelteile befinden, aufzunehmen. Für den Anwendungsfall des Bin Pickings ist eine tiefenwahrnehmende Kamera („3D-Kamera“) essentiell. Die Tiefenwahrnehmung wird hierbei über die Verwendung mehrerer Linsen ermöglicht. Hierbei wird eine digitale dreidimensionale Darstellung mithilfe von Millionen von Messpunkten erzeugt, die „Punktwolke“ genannt wird. [8] [14]

Die Kamera kann mobil oder stationär angebracht werden. Bei der mobilen Ausführung wird die Kamera an dem Roboter befestigt, sodass die Punktwolke direkt über dem Behälter aufgenommen werden kann. Eine weitere Möglichkeit ist die stationäre Befestigung der Kamera über dem Behälter. Hier muss beachtet werden, dass der Roboter nicht durch die Kamerahalterung in seiner Bewegungsfreiheit eingeschränkt wird. [15]

Vorteil der stationär angebrachten Kamera ist, dass die Punktewolke aufgenommen werden kann, während der Roboter fährt. So kann der nächste durchzuführende Griff berechnet werden, während der Roboter noch den vorherigen ausführt. Dies spart im Vergleich zur mobilen Anbringung einige Sekunden Zykluszeit ein, was den Durchsatz erhöht. [15]

Vorteil der mobil angebrachten Kamera ist der deutlich geringere Abstand des Sensors zum Behälter, so kann eine höhere Auflösung der Punktwolke erreicht werden, was das Risiko für nicht gefundene Griffstrategien senkt. Ein Nachteil dieser Umsetzung ist allerdings, dass die Kamera jederzeit mitbewegt wird, was die maximal bewegliche Nutzlast senkt und die Bewegungsfreiheit des Roboters einschränken könnte. [15]

3.2.4. Software

Der Aufgabenbereich der Software stellt ein breites Spektrum dar. Die Software erkennt aus der eintreffenden Punktewolke Einzelteile sowie deren Position und Lage, sucht sich in eventueller Abhängigkeit externer Parameter eines davon aus und berechnet eine Griffstrategie, mit der ein zuverlässiger Griff möglich ist. Daraufhin führt die Software eine Bahnplanung durch, um einen schnellen und kollisionsfreien Weg zur Griffposition und anschließend zum Ablageort zu finden. Auch der Ablageort kann von externen Parametern vorgegeben werden. [8]

Die Anforderungen an die Software sind hoch, für die Entwicklung wird somit sehr viel Fachwissen benötigt. Nicht grundlos wird das Bin Picking oftmals als „Königsdisziplin der Robotik“ bezeichnet. [8]

3.3. Aufbau der Bin-Picking-Station

Die Bin-Picking-Station ist im Rahmen vorangehender Studienarbeiten entstanden und wurde für Laborversuche und Messeausstellungen verwendet. Die Bin-Picking-Station baut auf einem Komplettsystem der Optonic GmbH auf, das aus einer tiefenwahrnehmenden Kamera und einer umfassenden Steuerung namens Mikado Control besteht. Bei Mikado Control handelt es sich um eine Lösung zur Roboteransteuerung, die auf Bin-Picking Anwendungen spezialisiert ist. Es besteht aus einer Hardware und drei zusammenhängenden Softwares. Diese sind kompatibel mit einem breiten Spektrum an Cobots, hier wurde sich für den UR3 des Herstellers Universal Robots entschieden. In bisherigen Laborversuchen wurden mit der Drehmaschine gefertigte Werkstücke gegriffen und in ein optisches 2D-Mikrometer geführt, um die Drehteile auf eingehaltene Fertigungstoleranzen zu prüfen. Abhängig davon wurden diese dann in Gut- und Schlechtteile sortiert.

Der UR3 verfügt über insgesamt sechs Achsen, die jeweils um 360° in beide möglichen Drehrichtungen gedreht werden können. Für die drei Hauptgelenke sind Winkelgeschwindigkeiten von 180°/s möglich, für die Werkzeuggelenke beträgt die maximale Winkelgeschwindigkeit 360°/s. Die maximale Nutzlast beträgt 3kg. Mit dem Werkzeug können Punkte in 500mm Entfernung zur Basis erreicht werden. Damit eignet sich der UR3 ideal für Bin-Picking-Anwendungen mit kleinen Bauteilen. Das geringe Gewicht von 9,4 eignet sich ebenfalls für die Integration in einer mobilen Station. [16]

Für die Prozessintegration in das Produktionsnetzwerk, soll der UR3 mithilfe eines Vakuumgreifers den in Abbildung 7 dargestellten, ungefähr zwei Zentimeter langen Bolzen greifen.



Abbildung 8: Universal Robots UR3 [16]

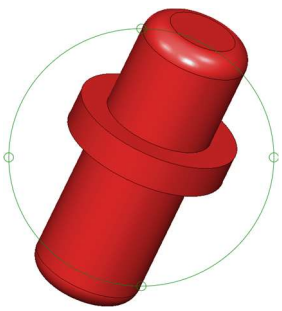


Abbildung 7: Zu greifender Bolzen

Erst durch eine Interaktion der einzelnen Systemkomponenten kann das Gesamtsystem in Betrieb genommen werden. Die einzelnen Komponenten des Bin-Picking-System werden dabei wie in folgender Abbildung 9 schematisch dargestellt, verdrahtet.

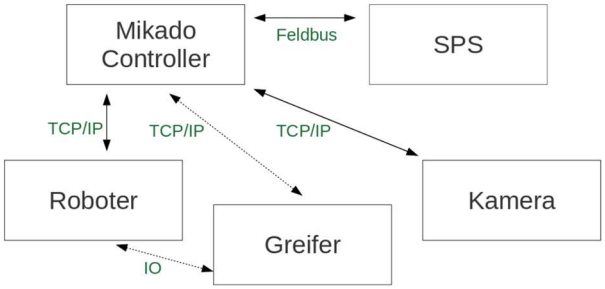


Abbildung 9: Signalverdrahtung der Bin-Picking-Station [17]

Der Mikado Controller stellt die Steuerung hardware- und softwareseitig dar. Die Kommunikation der Steuerung mit dem Roboter und der Kamera erfolgt über das Transmission Control Protocol/Internet Protocol (TCP/IP). Diese werden über ein Ethernet-Kabel mit dem Mikado Controller verbunden. Der Greifer als Werkzeug des Roboters kann über TCP/IP mit dem Mikado Controller angebunden werden oder per Input/Output-Link (IO-Link) mit den IO-Schnittstellen des Roboters verbunden werden.

Es besteht die Möglichkeit, den Mikado Controller über einen Feldbus wie zum Beispiel dem Process Field Network (PROFINET) mit einer SPS zu verbinden. Dies ermöglicht zusätzlichen Datenaustausch, mit dem in den Prozessablauf des Bin Pickings über externe Parameter eingegriffen werden kann.

3.4. Mikado Softwarepaket

3.4.1. Mikado Assistant

Eines der drei Programme, die Mikado Control zur Verfügung stellt, ist der Mikado Assistant. Hier wird die Konfiguration des Aufbaus vorgenommen. Die im Mikado Assistant vorgenommenen Einstellungen bilden die Grundlage der Bahnplanung und Ansteuerung des Roboters im eigentlichen Mikado-Programm. [17]

Zunächst wird hier ein neues Projekt erstellt. Man wird von einem Assistenten durch mehrere grundlegenden Einstellungen geführt. Die Sprache, die 3D-Kamera, der Robotertyp, die Art des Greifers und seine Ansteuerung werden hier definiert. Danach ist das Projekt erstellt. Die Arbeit im Mikado Assistant ist damit aber noch nicht abgeschlossen. [17]

Als nächstes geht es ans Erstellen eines digitalen Abbilds der Station im Reiter „Modell“. Hierzu können simple Geometrien wie Kugeln, Zylinder oder Quader erstellt werden. Es ist aber auch möglich, 3D Modelle einzubinden. Die verschiedenen Objekte können beliebig skaliert und platziert werden. Zur Verfügung stehen hier grundsätzlich zwei Koordinatensysteme: Das ortsfeste Weltkoordinatensystem der Station, und das Tool Flansch Koordinatensystem. Letzteres ist am Ende des Roboterarms (an der sogenannten Tool Flansch) fixiert, und bewegt sich mit dem Roboterarm mit. Es ist auch möglich, Objekte im Koordinatensystem anderer Objekte zu fixieren. Es wird nun also der physische Aufbau der Station erstellt, wobei auch die Definition einer Zellbegrenzung sinnvoll ist. Hierbei muss es sich nicht um ein physisches Objekt handeln, es werden lediglich Wände um die Station herum definiert, innerhalb derer sich der Arm bewegen darf. Auch der Endeffektor (der Greifer) muss eingebunden werden. Wichtig ist auch die Platzierung des Tool Center Points (TCP), da dieser später gezielt positioniert werden kann und besonders beim Greifprozess essenziell ist. Die Position eines gegriffenen Objekts wird nämlich abhängig vom TCP angegeben, und auch Griff-Rotationen erfolgen um den TCP. [17]

Das hier erstellte Abbild der Station ist ausschlaggebend für die Bahnplanung, da Mikado selbstständig Kollisionen verschiedener Objekte vermeidet. Das ist nur möglich, wenn Mikado auch über die exakte Position der Objekte Bescheid weiß. Im Assistant besteht auch die Möglichkeit, Kollisionen einzelner Objekte explizit zu erlauben. [17]

Ist das digitale Abbild erstellt, müssen im Reiter „Parameter“ noch einige Einstellungen getätigt werden. Roboterparameter wie Gelenkgrenzwerte oder die Ein- und Ausgangskonfiguration können hier eingestellt werden. Es ist nötig, einzustellen, über welche Ausgänge der Greifer angesteuert werden kann. Auch besteht die Möglichkeit, eine standard-Kalibrierung für die 3D Kamera zu definieren. Einige Objekterkennungs- und Griffparameter können hier ebenfalls angepasst werden. Beispielsweise können Überlappungskollisionen beim Griff erlaubt werden. Überschneiden sich zwei Objekte zum Zeitpunkt des Griffes, so handelt es sich um eine Überlappungskollision. Da zwei physische Objekte nicht denselben Platz beanspruchen können, sind Überlappungskollisionen meist Messfehler und somit irrelevant. Falls PROFINET genutzt wird, so kann im „Parameter“-Reiter auch eingestellt werden, welches Wort der PROFINET-Nachricht auf welche Mikado-Variable gemappt wird. Wurde die Konfiguration im Mikado Assistant beendet, so finden alle weiteren Einstellungen in Mikado statt. [17]

3.4.2. Mikado

Das Programm, in dem die eigentliche Programmierung stattfindet, heißt einfach Mikado. In der Oberfläche existieren verschiedene Reiter. Unter dem Reiter „Run“ ist die Ausführung und das Debuggen des Programms möglich. [17]

Unter dem Reiter „Edit“ findet die eigentliche Programmierarbeit statt. Im Tab „Programm“ wird der Programmcode erstellt. Ein Mikado-Programm besteht aus einer oder mehreren Sequenzen, wobei immer genau eine Sequenz aktiv ist. Die aktive Sequenz wird in Dauerschleife abgearbeitet. Eine Sequenz wiederum besteht aus einem oder mehreren Tasks. Tasks sind die Codebausteine der Sprache. Sie werden per drag and drop in den Sequenzen platziert. Tasks können Variablen manipulieren, mathematische und logische Operationen ausführen und Roboterbewegungen umsetzen. Logische Vergleiche funktionieren so, dass je nach Ergebnis in eine andere Sequenz gesprungen wird. Bei einem Vergleich muss also die Sequenz enden, und zwei neue Sequenzen für je eines der Ergebnisse erstellt werden. Auch Programmflussoperationen wie Warten und Sequenzwechsel sind Tasks. Die Aufnahmen von Kamerabildern, die Griffsuche und der tatsächliche Griff werden durch Tasks ausgelöst. [17]

Im Tab „Variablen“ können die Programmvariablen verwaltet werden. Hierfür stehen sowohl klassische Datentypen also auch Datentypen für Roboter- und Teileposen zur Verfügung. Hier kann bei laufendem Programm auch der Momentanwert der Variablen angezeigt werden. Auch konstante Werte müssen in Mikado über Variablen realisiert werden. [17]

Unter dem „Edit“-Reiter findet sich auch der Tab „Produkte“. Hier muss das zu greifende Objekt, das sogenannte Produkt, definiert werden. Jedes Produkt benötigt einen Namen und ein dreidimensionales Modell, das per Computer-Aided-Design (CAD) erstellt wird. Des Weiteren müssen einige Parameter definiert werden, um die Erkennung zu vereinfachen. Hierzu zählt beispielsweise die zu erwartende Produktzahl oder der Mindestabstand zwischen zwei Produkten. Auch die verschiedenen Griffe müssen definiert werden. Zu jedem Griff können zusätzlich verschiedene Transformationen erstellt werden, also Anpassungen des betreffenden Griffs. So kann dem System kommuniziert werden, dass derselbe Griff beispielsweise von mehreren Seiten möglich ist. Den einzelnen Griffen kann auch eine Gewichtung angefügt werden, so dass im Zweifelsfall immer ein bestimmter Griff bevorzugt wird. Es gibt immer exakt ein aktives

Produkt. Wird ein produktbezogener Task, wie beispielsweise die Griffsuche, durchgeführt, so wird hierfür immer das aktive Produkt genutzt, sofern nicht anders spezifiziert. Das aktive Produkt kann mittels Task geändert werden. [17]

Der letzte Tab im „Edit“-Reiter ist der „Szene“-Tab. Hier wird zum einen die Kiste, aus der gegriffen werden soll, definiert. Es wird ein 3D-Modell der Kiste benötigt, zusätzlich dazu müssen Position und Größe der Kiste angegeben werden. Jede Kiste braucht auch eine Übergangspose. Dies ist die Pose, die der Roboter einnimmt, bevor er ein Teil aus der Kiste entnimmt. Im „Szene“-Tab werden auch verschiedene Regions of Interest (ROI) definiert. Eine ROI ist beispielsweise der für die Kameraaufnahme relevante Bereich, oder der Bereich in dem nach Objekten gesucht werden soll. [17]

Unter dem Reiter „Tune“ kann in Mikado der momentane Zustand der Objekterkennung und Griffsuche eingesehen werden. Zu guter Letzt gibt es den Reiter „Tools“, unter dem beispielsweise das Kalibrieren der Kamera oder das manuelle Bewegen des Roboters möglich ist. [17]

3.4.3. Mikado Simulator

Das dritte Programm des Mikado-Systems ist der Simulator. In ihm kann der Betrieb des Roboters und das Abarbeiten des Programms simuliert werden. Der Simulator hat in dieser Studienarbeit keine Relevanz und wird nur der Vollständigkeit halber erwähnt. [17]

3.5. Der Bin-Picking-Prozess in Mikado

Um das Bin-Picking mit Mikado zu ermöglichen, sind mehrere Schritte notwendig. Zunächst einmal muss eine Punktwolke aufgenommen werden. Dies kann entweder synchron stattfinden, also während der Roboter in seiner Pose verharrt, oder asynchron. Im asynchronen Fall wartet der Controller, bis sich der Roboter außerhalb der ROI für die Kameraaufnahme befindet, und nimmt dann die Punktwolke auf. Der Roboter kann sich während der Aufnahme weiterhin bewegen. Wurde die Punktwolke aufgenommen muss im Anschluss per Task die Griffsuche gestartet werden. Hierbei findet zunächst die Objekterkennung statt.

Diese läuft in zwei Schritten ab:

1. Näherungsweises Matching:
   Kamerabild und CAD-Modell des Produkts werden auf wenige Punkte reduziert. Dann werden diese verglichen. Sind zwei Punkte aus Kamerabild und CAD-Modell näher aneinander als der definierte Scoring-Abstand, steigt der Erkennungs-Score. Erreicht der Erkennungs-Score in einem Bereich einen gewissen Wert, so wird ein Objekt erkannt.
2. Verfeinerung
   Die zuvor erkannten Objekte werden nun erneut untersucht. Es werden nun mehr Punkte genutzt, um das Objekt zu bestätigen und die genaue Orientierung und Position zu erkennen.

Wenn die entsprechende Option gesetzt wurde, werden danach noch die Matches gespeichert, um die nächste Objektsuche zu beschleunigen. [17]
Wurde ein Objekt gefunden, so läuft der zweite Teil der Griffsuche an. Hierbei werden für die erkannten Produkte die definierten Griffe überprüft. Ein Griff besteht aus 4 Schritten:

1. Anfahren: Der Roboter startet in der Übergangspose und fährt in die Anfahrtspose. Diese liegt eine definierte Distanz in Stoßrichtung von der Greifpose entfernt, wobei der Endeffektor die gleiche Orientierung wie in der Greifpose hat.
2. Greifen: Der Arm fährt von der Anfahrtspose linear in die Greifpose.
3. Entnahme: Der Roboter fährt von der Greifpose aus linear in die Entnahmepose, die eine definierte Distanz von der Greifpose entfernt liegt, wobei der Endeffektor die gleiche Orientierung wie in der Greifpose hat.
4. Zurückfahren: Der Roboter fährt von der Entnahmepose in die Übergangspose.

Diese Trajektorien werden für mehrere Griffe berechnet und auf Kollisionen überprüft. Es ist möglich einzustellen, dass das Überprüfen der Griffe abgebrochen wird, sobald eine gewisse Anzahl an erfolgreichen Griffen gefunden wurde. Aus den gefundenen Griffen wird dann ein einziger ausgewählt. Diese Auswahl berücksichtig die händisch gesetzte Gewichtung des Griffs, aber auch Kriterien wie die Dauer und Fahrtstrecke des Griffs. Auch der Erkennungs-Score kann in die Gewichtung einfließen. [17]

Um nun auch tatsächlich einen Griff durchzuführen, muss nur noch der Task „Greife Objekt“ aufgerufen werden. Der Roboter fährt dann in die Übergangspose und führt den zuvor isolierten Griff aus. [17]

3.6. Grundlagen der PROFINET-Kommunikation durch Mikado

Der Mikado-Controller stellt über PROFINET einen 64 Byte Eingang und einen 64 Byte Ausgang zur Verfügung. Diese Schnittstellen werden in Wörter à 2 Byte unterteilt. [18]

Um die Bus-Kommunikation auszulesen, müssen den Wörtern im Mikado Assistant entsprechende Variablen zugewiesen werden. Die entsprechende Oberfläche ist in Abbildung 10 zu sehen. Die linke Tabelle ist hierbei die Kommunikation von der SPS zu Mikado, die rechte die Kommunikation von Mikado zur SPS. Die Variablen können im Mikado-Programm importiert und wie alle anderen Variablen genutzt werden. Einige Variablen werden automatisch zwischen den Systemen synchronisiert, andere nur auf Anfrage. Die automatisch synchronisierten werden im Mikado Assistant durch ein blaues Quadrat in der Spalte „Live“ gekennzeichnet. [17]

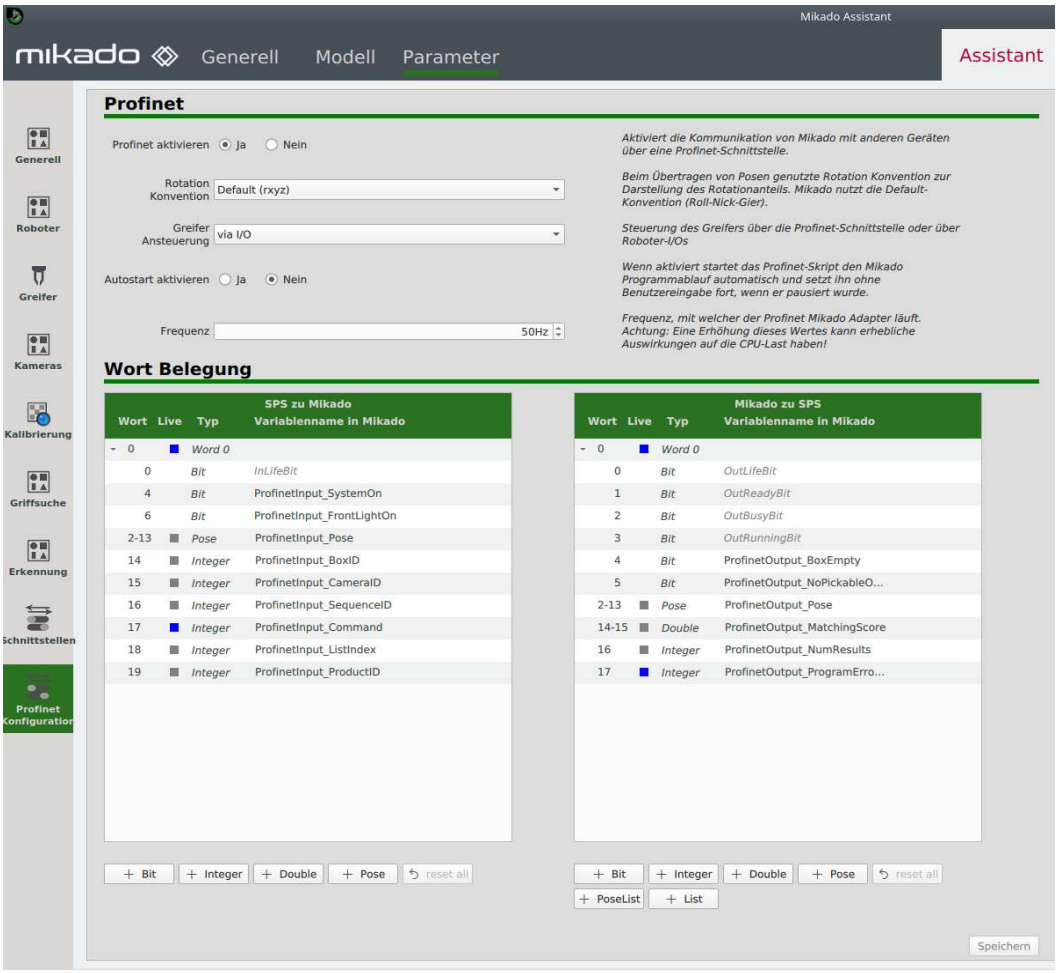


Abbildung 10: Belegung der PROFINET-Wörter im Mikado Assistant

Standardmäßig sind bereits bestimmte Variablen implementiert. Einige dieser Signale sind besonders wichtig für die Bin-Picking-Anwendung:

1. Das InLifeBit und das OutLifeBit (Wort 0 Bit 0 in beiden Richtungen): Diese Bits dienen zur Überprüfung, ob eine Verbindung zwischen SPS und Mikado-System existiert. Das gesamte Wort 0 wird automatisch zwischen SPS und Mikado synchronisiert. Mikado invertiert in regelmäßigen Abständen das OutLifeBit. Die SPS muss das OutLifeBit auf das InLifeBit spiegeln. Passiert das nicht, schließt Mikado daraus, dass die Verbindung unterbrochen wurde, und beendet die Kommunikation.
2. Der Integer ProfinetInput_Command (Wort 17 von SPS zu Mikado): Auch dieses Wort wird automatisch synchronisiert. Hierüber kann die SPS verschiedene Befehle an Mikado senden. So kann die SPS beispielsweise das Mikado-Programm starten und stoppen, das Programm in bestimmte Sequenzen springen lassen, oder, was für die Bin-Picking-Station besonders wichtig ist, das Synchronisieren aller Wörter zwischen SPS und Mikado auslösen. Dies geschieht, indem der Integer auf den Wert 6 gesetzt wird.
3. Der Integer ProfinetOuptut_ProgrammErrorCode (Wort 17 von Mikado zu SPS): Dieses Wort wird automatisch zwischen den Systemen synchronisiert. Tritt ein Fehler im Mikado-Programmablauf auf, wird dieser der SPS über dieses Wort mitgeteilt. Hierfür sind die Werte -1 bis -18, sowie -100 reserviert. Der positive Bereich kann frei genutzt werden.

Um die Kommunikation auf SPS-Seite zu ermöglichen, muss eine Mikado-Datei in der Generic Station Description Markup Language (GSDML) im SPS-Programm eingebunden werden. Dies erzeugt ein Netzwerkelement, in welches dann beliebig Werte für die einzelnen Wörter geschoben werden können. [18]

3.7. Digitaler Zwilling

Ein digitaler Zwilling ist ein virtuelles Modell eines physischen Objektes oder Systems, das präzise auf digitaler Ebene widergespiegelt werden soll. Hierfür werden konzeptionelle Modelle wie CAD-Dateien importiert und kombiniert. Je präziser das reale Produkt digital abgebildet wird, desto besser lassen sich Simulationen durchführen und mit deren Ergebnissen qualitativ hochwertigere Schlüsse ziehen. [19]

Die Simulation stellt neben der reinen Visualisierung den größten Vorteil des digitalen Zwillings dar. So können in einem virtuellen Versuchsaufbau beispielsweise Bewegungen von Robotern simuliert werden und auf Konformität mit deren Gelenkgrenzen überprüft werden. Gerade während der Entwicklung und des Aufbaus des Systems stellt dies eine wichtige Grundlage dar, da bereits im Voraus Fehler gefunden und verhindert werden können. Dies stellt gerade in der Entwicklung einen immensen Vorteil dar, da hier auftretende Fehler sehr hohe Kosten verursachen und unbedingt vermieden werden sollten. [19] [20]

Als Addition zur Visualisierung des geplanten Konzeptes gibt es die Möglichkeit, diese nicht nur auf einem Bildschirm darzustellen, sondern auf Geräte mit Unterstützung von Virtual-Reality (VR) oder Augmented-Reality (AR) zurückzugreifen, um sich ein deutlich besseres, realer wirkendes Bild des Konzeptes zu machen. [19]

Viele Entwickler erkennen das Potential, das digitale Zwillinge bieten und setzen bereits auf diese Technologie. Marktanalysen zufolge betrug der weltweite Umsatz des Marktes an digitalen Zwilling im Jahre 2022 insgesamt 6,8 Milliarden USD. Für 2023 wird ein Anstieg auf 10,1 Milliarden USD erwartet. Die Technologie stellt auch für kleinere und mittelständische Unternehmen eine gute Investition dar, da gerade hier teure Entwicklungsfehler gravierende Folgen tragen können. Gerade bei diesen Unternehmen wird für die kommenden Jahre ein besonders hohes Umsatzwachstum prognostiziert. Für das Jahr 2028 wird mit einem weltweiten Umsatz von 110,1 Milliarden USD gerechnet, was mehr als einer Verzehnfachung des Umsatzes von 2023 entsprechen soll. Auch in Deutschland gewinnt dieser Markt immer mehr an Bedeutung, mit Siemens und Bosch haben sich zwei große Unternehmen in diesem Markt positioniert. [21] [22]

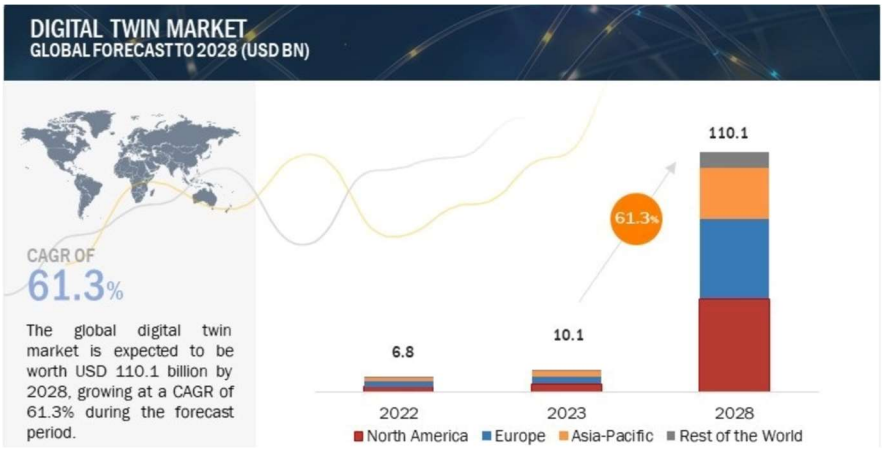


Abbildung 11: Prognose des weltweiten Marktvolumens an digitalen Zwillingen in Mrd. USD [22]

4. Projektplanung

Zur Projektplanung wird ein Gantt-Diagramm erstellt, anhand dessen wichtige Termine sowie die benötigte Zeitdauer der einzelnen Arbeitsschritte im Laufe der Bearbeitung übersichtlich wahrgenommen werden können. So können Abweichungen vom Zeitplan erkannt werden und frühzeitig gegensteuernde Maßnahmen getroffen werden. Das Gantt-Diagramm ist in Abbildung 12 dargestellt.

Die Praxisphasen enden zum 14.04, daher wird am Montag, dem 17.04 mit der weiteren Bearbeitung begonnen. Gerade zu Anfang gibt es viele Aufgabenbereiche abzudecken. Es gilt, den Sauggreifer auszutesten und anschließend mittels Magnetventil in Betrieb zu nehmen. Parallel muss der digitale Zwilling der Bin-Picking-Station erstellt werden und an Gruppe 9 zur Zusammenstellung des gesamten Aufbaus im PTZ gesendet werden. Von Beginn an wird sich der Programmierung von Mikado gewidmet. Sobald das abgeschlossen ist, wird mit der Einrichtung der PROFINET-Schnittstelle zur SPS sowie einer Optimierung des Mikado-Programms begonnen. Ebenfalls werden Recherchen zum Machine Learning angestoßen. Sobald die PROFINET-Schnittstelle eingerichtet ist, werden finale Tests durchlaufen und falls nötig weitere Optimierungen getroffen. Eine Woche vor der Klausurwoche soll die Bearbeitung abgeschlossen sein, um über eine freie Woche zur Klausurvorbereitung zu verfügen. Am 28.06, mitten in der Klausurwoche, findet eine Abschlussveranstaltung statt, in der das Produktionsnetzwerk vorgestellt wird. Sollte es Abweichungen im Zeitplan geben, so muss spätestens zu diesem Termin die vollständige Funktionalität gewährleistet sein. Im Anschluss an die Klausurwoche wird die Studienarbeit verfasst.

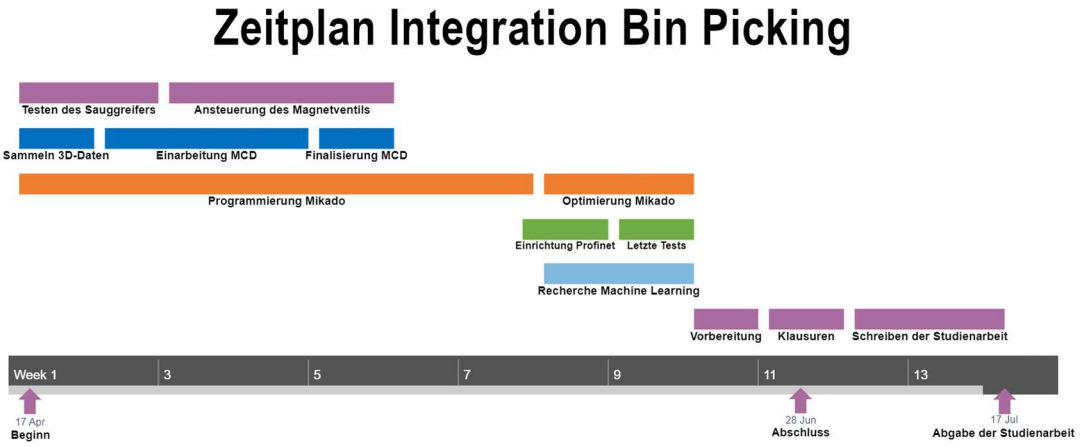


Abbildung 12: Gantt-Diagramm zur Projektüberwachung

5. Digitaler Zwilling

Im Rahmen der Studienarbeiten verschiedener am Produktionsnetzwerk beteiligter Gruppen soll ein digitaler Zwilling der gesamten Fertigungsstätte innerhalb des PTZ erstellt werden. Hierbei gilt es, den konstruktiven Aufbau der einzelnen Stationen sowie eine Simulation des gesamten Produktionsprozesses zu veranschaulichen. Dies eignet sich sowohl zu Demonstrationszwecken als auch zu Simulationszwecken vor oder während der Entwicklung des Produktionsnetzwerkes, um mögliche Problemstellen im Voraus ausfindig zu machen oder Optimierungsmöglichkeiten zu finden und zu testen.

Erste Erfahrungen mit digitalen Zwillingen wurden während des ersten Blocks der Studienarbeit gemacht, als der Umbau der Bin-Picking-Station zur Ermöglichung der Eingliederung dieser in das Produktionsnetzwerk zu ermöglichen. Durch Replikation der Bin-Picking-Station sowie des relevanten Teils des Transfersystems zwischen den einzelnen Stationen auf digitaler Ebene im „MTpro Online Designer“ von Bosch [23] konnten einige Fehler in der bisherigen Planung des Umbaus gefunden werden, die aufgrund von Fehlern in Skizzen oder fehlender Visualisierung aufgetreten sind.



Abbildung 13: Digitales Abbild der Bin-Picking-Station im MTpro Online Designer

Dieses digitale Modell dient allerdings nur der Planung des Umbaus. Die Online-Software ermöglicht keine Simulation, da hier kein Roboter eingefügt werden kann. Daher muss auf eine andere Software ausgewichen werden, hierfür wird das Programm Mechatronic Concept Designer“ (MCD) [24] von Siemens verwendet.

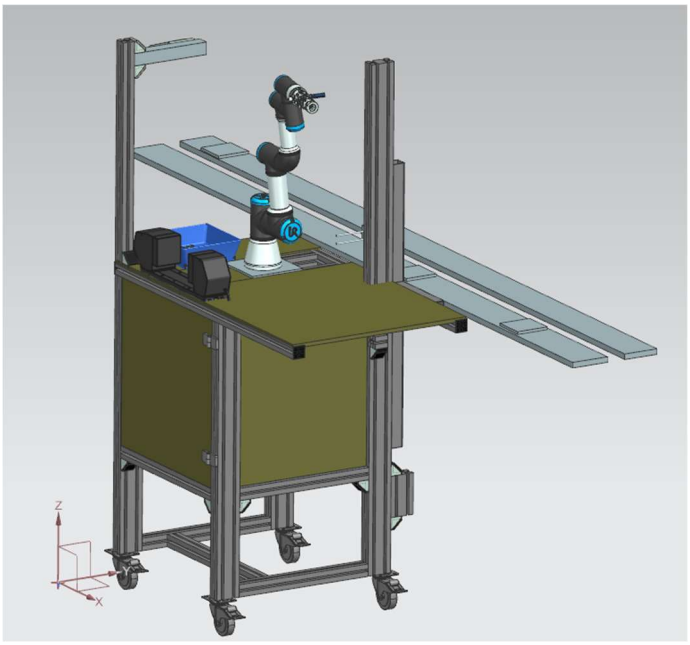


Abbildung 14: Digitaler Zwilling der Bin-Picking-Station

Die digitalen Zwillinge aller Stationen werden gesammelt und zusammengefügt, sodass ein digitales Abbild des gesamten Produktionsnetzwerkes im PTZ entsteht. In der Simulation kann das Förderband eingeschaltet werden und der gesamte Produktionsprozess durchlaufen werden.

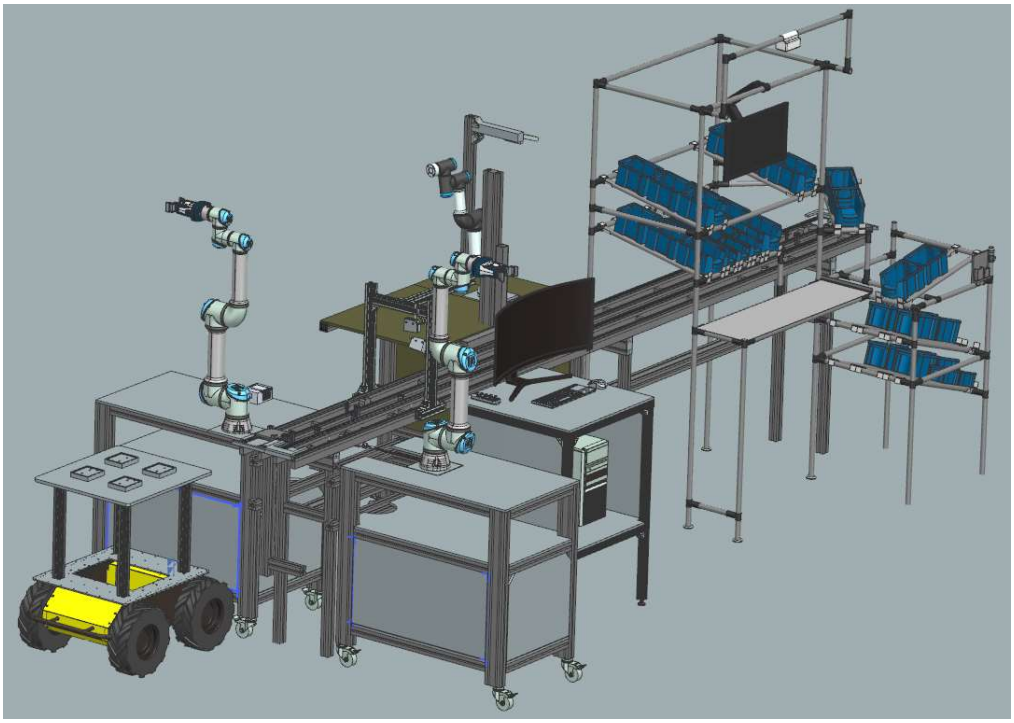


Abbildung 15: Digitaler Zwilling des Produktionsnetzwerks im PTZ

6. Integration in den Produktionsprozess

Im Folgenden sollen die Schritte detailliert werden, die notwendig waren, um den anforderungsgemäßen Ablauf des Produktionsprozesses an der Bin-Picking-Station zu gewährleisten.

6.1. Die Wahl des Greifers

Ursprünglich war es geplant, einen Vakuumgreifer zu nutzen (siehe Abbildung 16). Es handelt sich hierbei um einen pneumatischen Vakuumerzeuger. Im Inneren des Vakuumgreifers befindet sich ein Hohlraum, sowie ein seitlich geöffnetes Rohr. An das Rohr wird Druckluft angeschlossen, welche einen Luftstrom durch es fließen lässt, der an der anderen Seite aus dem Greifer austritt. Dieser Luftstrom erzeugt unter Ausnutzung des Bernoulli-Effekts (vgl. [25]) einen Unterdruck im Hohlraum. Der Hohlraum endet in einer Gummidüse, die an das zu greifende Objekt bewegt wird. Da der Luftdruck im inneren des Greifers geringer ist als außen, drückt der äußere Luftdruck das Objekt gegen die Düse und fixiert es dort.



Abbildung 16: Pneumatischer Vakuum-Endeffektor

Es wird klar, dass zum Schalten des Vakuumgreifers ein Ventil nötig ist, dass den Greifer mit der Druckluft verbindet oder trennt. Hierfür sollte das Elektromagnetventil EMVP 5 24V-DC 3/2 NC des Herstellers J. Schmalz GmbH (siehe Abbildung 17) genutzt werden.

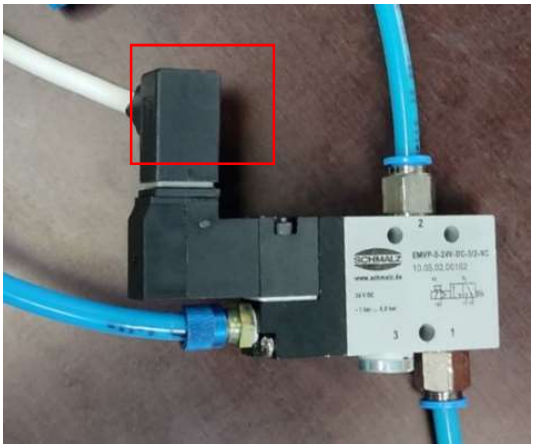


Abbildung 17: Magnetventil

Die Ansteuerung dieses Ventils nahm einige Zeit in Anspruch. Zunächst kam es zu Verwirrungen bezüglich des elektrischen Anschlusses des Ventils. Es stellte sich heraus, dass bereits ein Stecker am Ventil angeschlossen war (Markierung in siehe Abbildung 17). Dieser musste auseinandergebaut werden und mit einem 3-adrigen Kabel verbunden werden. Im Anschluss wurde klar, dass das Magnetventil auch einen Steuerdruck benötigt, um schalten zu können. Dieser muss nach Rückfrage bei Schmalz im Bereich 3-6 bar liegen. Da das Druckluftsystem des PTZ einen Druck von 8 bar führt, sollte das Druckregelventil MS2-LR-M5-D6-AR-BAR-B des Herstellers Festo GmbH (siehe Abbildung 18) genutzt werden, um diesen Steuerdruck anzulegen.

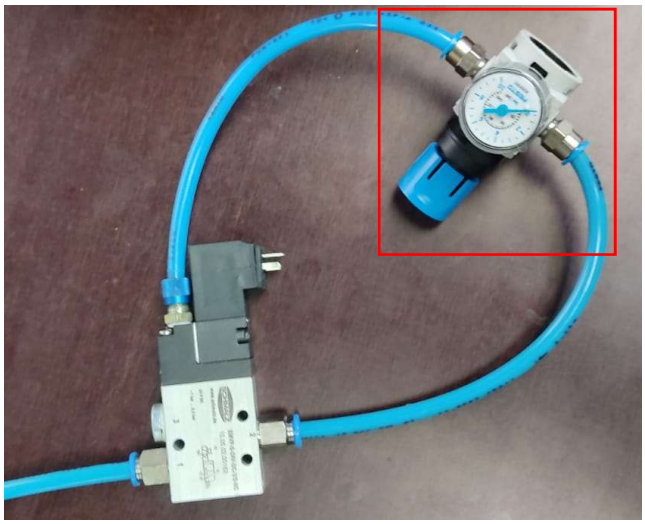


Abbildung 18: Konfiguration des Magnetventils mit Druckregler (markiert)

Bei der Inbetriebnahme kam es jedoch zu mehreren Problemen:
Wenn das Magnetventil geschlossen ist, so fließt kein Luftstrom durch den Steuereingang. Ist das Magnetventil offen, so fließt ein Luftstrom durch den Steuereingang und damit auch durch den Druckregulator. Der Druckregulator wiederum stellte sich als nicht sehr stabil heraus. Fließt ein Luftstrom durch den Druckregulator, so steigt der Druck an. Fließt keine Luft durch den Druckregulator, so sinkt der Druck. Dies führte dazu, dass je nach Einstellung des Druckregulators eines der folgenden Ereignisse auftritt:

1. Das Ventil ist geschlossen. Steuert man das Ventil an, um es zu öffnen, funktioniert das problemlos. Nun steigt aber der Steuerdruck, was es unmöglich macht, dass Ventil wieder zu schließen.
2. Das Ventil ist geschlossen. Der eingestellte Steuerdruck ist zu niedrig, um das Ventil zu öffnen.

Es gab keinen einstellbaren Druck, bei dem das Ventil zuverlässig geöffnet und geschlossen werden konnte. Dies war jedoch nicht das einzige Problem. Auf dem Datenblatt und dem Magnetventil selbst stand, dass das Ventil einen Druck von -1 bar bis 8 bar schalten kann. Dies war insofern der Fall, als es (mit dem richtigen Steuerdruck, der für diese Tests händisch angepasst wurde) möglich war, das Ventil zu schließen und zu öffnen, während ein Druck von 8 bar am Ventileingang anlag. Jedoch leitete das Ventil nicht den Druck von 8 bar weiter. Je höher der Steuerdruck, desto höher wurde auch der durchgeschaltete Druck. Mit einem Steuerdruck von 5 bar war der am Vakuumgreifer anliegende Druck nicht hoch genug, um den Bolzen zuverlässig zu greifen. Bei einem Druck von 8 bar war das kein Problem.
Es stellte sich also heraus, dass der Vakuumgreifer mit den gegebenen Gerätschaften nicht zuverlässig geschaltet werden kann und auch nicht in der Lage ist, den Bolzen zuverlässig aufzuheben.
Eine alternative zum Vakuumgreifer ist der klassische Zwei-Backen-Greifer, welcher in der vorangehenden Studienarbeit für die Laborversuche und Messeausstellungen entwickelt wurde. Erste Versuche zeigen, dass die zwei Parallelbackengreifer zu weit voneinander entfernt sind, um einen prozesssicheren Griff gewährleisten zu können. Deren 3D-Modell wird etwas angepasst, sodass die Greifbacken insgesamt zwei Millimeter näher aneinander liegen (siehe ***Abbildung 19*** und ***Abbildung 20***).

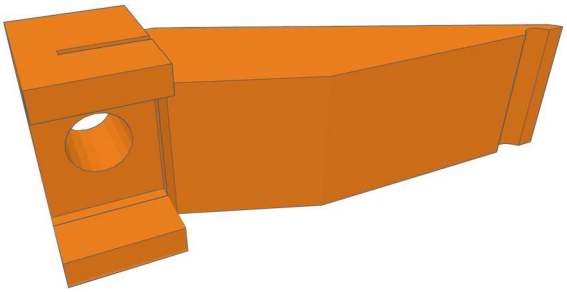


Abbildung 19: 3D-Modell der neuen Greifbacken

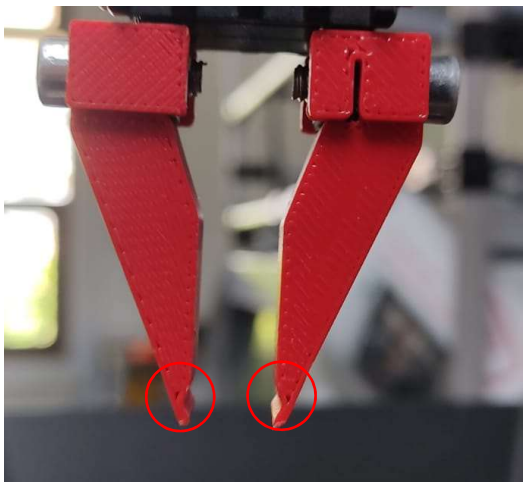


Abbildung 20: 2-Backen-Greifer, Rillen markiert

Dieser hat gegenüber dem Vakuumgreifer einige entscheidende Vorteile:

1. Er kann zuverlässig angesteuert werden
2. Er kann das Bauteil sicher greifen
3. Er kann das Bauteil besser reproduzierbar greifen. Beim Vakuumgreifer hängt der Bolzen nach dem Griff am Ende einer Gummidüse. Diese ist beweglich. Sie wird beim Greifen durch den Unterdruck im inneren des Greifers verkürzt, und außerdem durch das Gewicht des Bolzens verbogen. Dies lässt sich zum einen schwer berechnen, zum anderen lässt es sich nicht in Mikado modellieren. Mikado geht beim erfolgten Griff davon aus, dass das Bauteil relativ zum TCP fixiert ist, und zwar in der Pose, in der es sich zum Zeitpunkt des Griffs befand. Dieses Problem existiert beim 2-Backen Greifer nicht. Er greift das Teil, welches dann fixiert im Greifer gehalten wird und seine Pose relativ zum TCP nicht mehr verändert, bis der Greifer geöffnet wird.

Aus diesen Gründen wurde sich schlussendlich dazu entschieden, den 2-Backen-Greifer zu nutzen. Zusätzlich zeigte sich, dass durch die Rille in den Greifbacken (Markierung in ***Abbildung 20***) ein ungenauer Griff korrigiert werden kann, da sich der Bolzen parallel zur Rille ausrichtet.

6.2. Anforderungen

Der erste Schritt, um einen anforderungsgerechten Ablauf zu garantieren, ist das Definieren von Anforderungen. Dies soll hier erfolgen.

1. Geforderter Ablauf:
   1.1. Die Station muss erkennen, wenn ein neuer Bauteilträger an der Station ankommt. Dann wird der Prozess gestartet
   1.2. Die Station muss erkennen, welche Version des Produkts produziert wird, und dementsprechend ihren Prozess anpassen.
   1.3. Es muss ein Bolzen gegriffen werden, und in den Bauteilträger eingesetzt werden. Wie viele Bolzen wo eingesetzt werden ist versionsabhängig.
   1.4. Nach Abschluss des Einsetzvorgangs muss dem Gesamtsystem mitgeteilt werden, dass das momentane Bauteil fertiggestellt ist. Dann wird das Bauteil weitergeschickt und ein neues trifft ein.
2. Um mit der SPS und den anderen Anlagen zu kommunizieren, muss die Station an das PROFINET-Netz angeschlossen werden.
   2.1. Die Station muss von der SPS erfahren, wenn ein neuer Bauteilträger eintrifft, und was die Version des Bauteilträgers ist.
   2.2. Die Station muss der SPS mitteilen, wenn ein Arbeitszyklus abgeschlossen wurde.
3. Um einen schnellen Ablauf zu garantieren, soll die Station vorbereitend einen Bolzen greifen, auch wenn kein neuer Bauteilträger da ist. So kann beim Eintreffen des neuen Bauteilträgers direkt mit dem Einsetzen gestartet werden, und das Bauteil ist schneller fertig bearbeitet.
4. Um einen fehlerfreien Ablauf zu ermöglichen, muss eine Routine eingefügt werden, welche aufgerufen wird, wenn kein Bauteil gegriffen werden kann. Diese Routine durchmischt die Bauteile, woraufhin erneut versucht wird ein Teil zu greifen. Sinnvollerweise bricht diese Routine nach einigen aufeinander folgenden Aufrufen das Programm ab, da entweder keine Bolzen mehr vorhanden sind oder die Bolzen so liegen, dass das Durchmischen sie nicht betrifft.

6.3. Erstellung einer Projektkonfiguration im Mikado Assistant

Erster Schritt beim Erstellen eines Mikado-Programms ist das Implementieren eines digitalen Abbilds der Station im Mikado Assistant. Die Station wurde in einer vorangehenden Studienarbeit für Laborversuche bereits eingerichtet, allerdings wurden in der ersten Studienarbeitsphase im Rahmen der Studienarbeit T3100 einige Umbauten vorgenommen. Interessanterweise ist es jedoch im Mikado Assistant nicht ohne weiteres möglich, den Aufbau der Station eines alten Projekts in ein neues Projekt zu übernehmen. Um zu erklären, wie der Aufbau trotzdem übernommen werden kann, soll an dieser Stelle ein kleiner Exkurs in die Dateistruktur eines Mikado-Projekts unternommen werden.

6.3.1. Aufbau eines Mikado Projekts

Auf dem Mikado Rechner existiert ein Ordner „mikado“ mit dem Unterordner „cells“. Hierin befinden sich die Projektordner der einzelnen Mikado-Projekte.
In einem Projektordner befinden sich drei .bash Dateien, welche den Simulator, Mikado und den Assistant starten. Auch befindet sich hier ein Ordner namens „urdf“. In diesem befinden sich die sogenannten urdf-Files, wobei URDF für Unified Robot Description Format steht. Das Dateiformat der urdf-Files ist .xacro, es handelt sich also um eine Form einer XML (Extensible Markup Language). [17] Es gibt drei solcher Dateien:

1. world.xacro

   In diesem Dokument wird die Welt definiert. Das bedeutet konkret: Das Welt-Koordinatensystem wird definiert, die Pose der Roboterbasis und der 3D-Kamera wird festgesetzt. Außerdem werden die urdf-Files von Roboterarm, Greifer, Kamera und der Zelle importiert.
2. workcell.xacro

   Hier wird die Roboterzelle definiert. Die Zellbegrenzungen werden dimensioniert und platziert, und auch alle vom Nutzer im Assistant platzierte und mit dem Weltkoordinatensystem verknüpfte Objekte werden hier definiert. Zunächst wird Position und Orientierung der Objekt-Koordinatensysteme festgelegt. Dann wird bei einfachen Formen, die nicht über CAD-Dateien definiert wurden, ein entsprechendes Objekt erstellt, bei den anderen wird die entsprechende CAD-Datei eingebunden. Das Objekt wird skaliert und mit dem Objekt-Koordinatensystem verknüpft.

3. gripper.xacro
   Hier wird der Greifer, sowie sonstige mit dem Tool Flansch-Koordinatensystem verknüpfte Objekte definiert. Auch die möglichen Bewegungen des Greifers werden hier definiert.

Im Projektordner befindet sich außerdem der Ordner „config“. Hier befinden sich verschiedene .yaml-Dateien. In ihnen werden die im Assistant definierten Parameter gespeichert. Da währen beispielsweise die Datei gripper.yaml, in der festgehalten wird, wie der Greifer angesteuert wird, oder status.yaml, in der die in Mikado angezeigten Statusanzeigen festgelegt werden. Außerdem befindet sich in diesem Ordner die world.srdf, in der zusätzliche Informationen zum Zellenmodell festgehalten werden. Hierbei handelt es sich beispielsweise um die erlaubten Kollisionen oder die Unsichtbarkeit der Zellbegrenzung. Zu guter Letzt befinden sich im „config“-Ordner noch Kamerakalibrierungen und die assistant.settings Datei mit den Assistant-Einstellungen.

Im Ordner „launch“ befinden sich .launch-Files, welche notwendige Informationen zum Öffnen des Projekts in Mikado enthalten. Sie verbinden die einzelnen Dateien im Projektordner miteinander.

Im „meshes“-Ordner werden die benötigten CAD-Dateien abgelegt.

Im „products“- Ordner sind die .product-Files für die definierten Produkte, inklusive Griffe, sowie eine .productlist-Datei, welche den Zugriff auf die Produkte ermöglicht.

Im programs Ordner wird das Mikado-Programm inklusive Variablen in einer .progam-Datei gespeichert.

Im scenes Ordner zuletzt werden die angelegten Szene-Objekte in einer gemeinsamen .scene-Datei gespeichert.

Die restlichen Ordner sind für die weitere Bearbeitung nicht interessant.

6.3.2. Duplizieren einer Mikado Assistant Projektkonfiguration

Mit dem nun gewonnenen Wissen ist es vergleichsweise einfach, ein neues Projekt zu erstellen und dabei die alte Konfiguration von Zelle und Roboter beizubehalten. Im Assistant wird ein neues Projekt angelegt. Dann werden die Ordner „config“, „launch“, „meshes“ und „urdf“ aus dem alten Projekt ins neue kopiert, wobei die automatisch

angelegten Ordner des neuen Projekts überschrieben werden. Alle kopierten Dateien müssen geöffnet werden, und der Name des alten Projekts an jeder Stelle durch den des neuen ersetzt werden. Nun enthält das neue Projekt die Konfigurationen des alten.

Danach müssen die zuvor vorgenommenen Umbauten an der Station noch im Assistant ergänzt werden. Da die gleiche Kiste wie in der vergangenen Studienarbeit genutzt wird, bietet es sich auch an, den Inhalt des „scenes“-Ordners zu übernehmen. Auch das wurde gemacht.

An dieser Stelle mag der geneigte Leser sich vielleicht fragen: „warum all der Aufwand, man kann in einem Projekt doch bestimmt auch mehrere Programme erstellen?“ Mit dieser Frage hätte der Leser recht. Es ist jedoch sinnvoll, ein separates Projekt zu erstellen, falls Greifer oder sonstige Zellparameter zwischen den beiden Projekten variieren sollen. Da der Plan lange Zeit war, einen Sauggreifer zu nutzen, sowie den Roboterarm für dieses Projekt von seiner Ursprungsposition zu verschieben, musste wie hier beschrieben vorgegangen werden. Am Ende wäre dies nicht nötig gewesen, ist aber trotzdem zugunsten zukünftiger Änderungen sinnvoll.

6.4. Erstellung des Mikado-Programms

Um das Bin-Picking zu ermöglichen, muss zunächst ein Produkt erstellt werden. Das betreffende Produkt wurde „Bolzen“ genannt, und die zur Verfügung gestellte CAD-Datei eingebunden (siehe Abbildung 21).

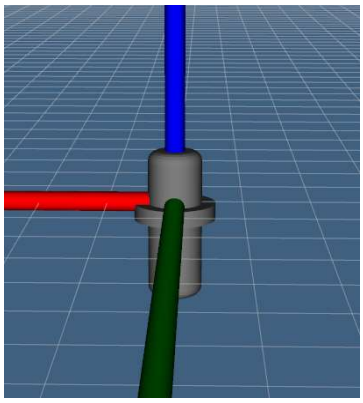


Abbildung 21: Konfiguration des Mikado-Produkts "Bolzen", x-Achse rot, y-Achse grün, z-Achse blau

Bei den Griffen wurden zwei Möglichkeiten in Betracht gezogen:

1. Ein Griff von oben, siehe Abbildung 22:

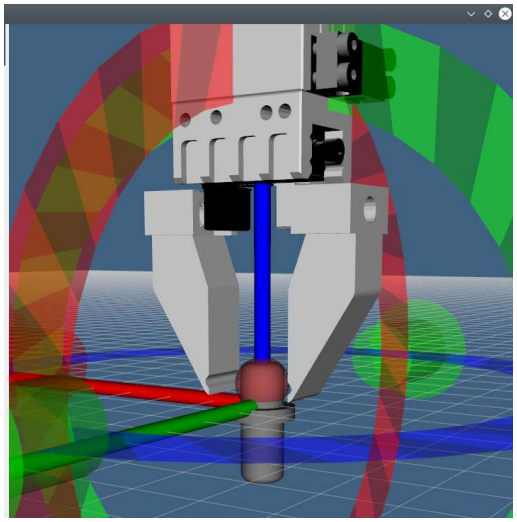


Abbildung 22: Griff von oben

Ein solcher Griff hat den Vorteil, dass die Einsetzbewegung des Bolzens in Stoßrichtung des Endeffektors verläuft. Das bedeutet, dass hier die Kollision mit dem Bauteilträger keine so große Gefahr darstellt. Der Griff hat jedoch auch zwei entscheidende Nachteile:

- Er ist nur ausführbar, wenn der Bolzen auf der Spitze steht.
- Der Bolzen verrutscht und verdreht sich schnell im Griff, da der Schwerpunkt des Bolzens weit vom Griffpunkt entfernt ist und auch nicht die Rille in den Greifbacken (siehe Abbildung 20) genutzt werden kann, um den Bolzen in Position zu halten.

2. Ein Griff von der Seite, siehe Abbildung 23:

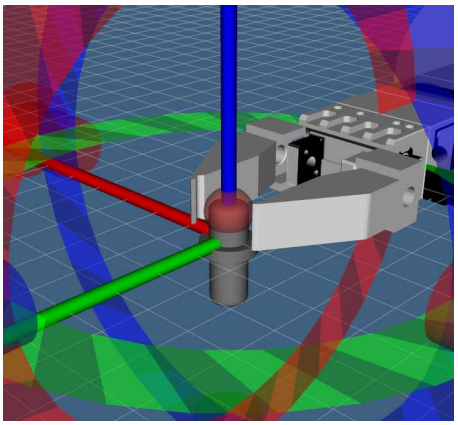


Abbildung 23: Griff von der Seite

Ein solcher Griff ist deutlich öfter machbar als Griff 1. In der Regel werden die Bolzen in der Kiste auf der Seite liegen. Auch kann hier, im Gegensatz zu Griff 1, die Rille in den Greifbacken genutzt werden (siehe Abbildung 20), um den Bolzen

in seiner Ausrichtung im Greifer zu fixieren, und diese bei leichten Fehlgriffen sogar zu korrigieren.

Aufgrund der Gefahr des Verrutschens des Bolzens bei Griff 1 wurde sich gegen diesen Griff entschieden. Griff 2 wurde umgesetzt. Hierbei wurde auch die Symmetrie des Bolzens zu seiner z-Achse genutzt (siehe Abbildung 21). Der Griff wurde also um den Bolzen rotiert, so dass er immer gegriffen werden kann, egal welche Seite des Bolzens als positive x-Richtung identifiziert wird.

Wie zuvor erwähnt, wurde die Szene aus der vorherigen Studienarbeit übernommen. Hier musste also nichts mehr gemacht werden.

Anschließend ging es an die Erstellung des Programms. In der vorherigen Studienarbeit wurde das in Abbildung 24 zu sehende Programmflussdiagramm erstellt:

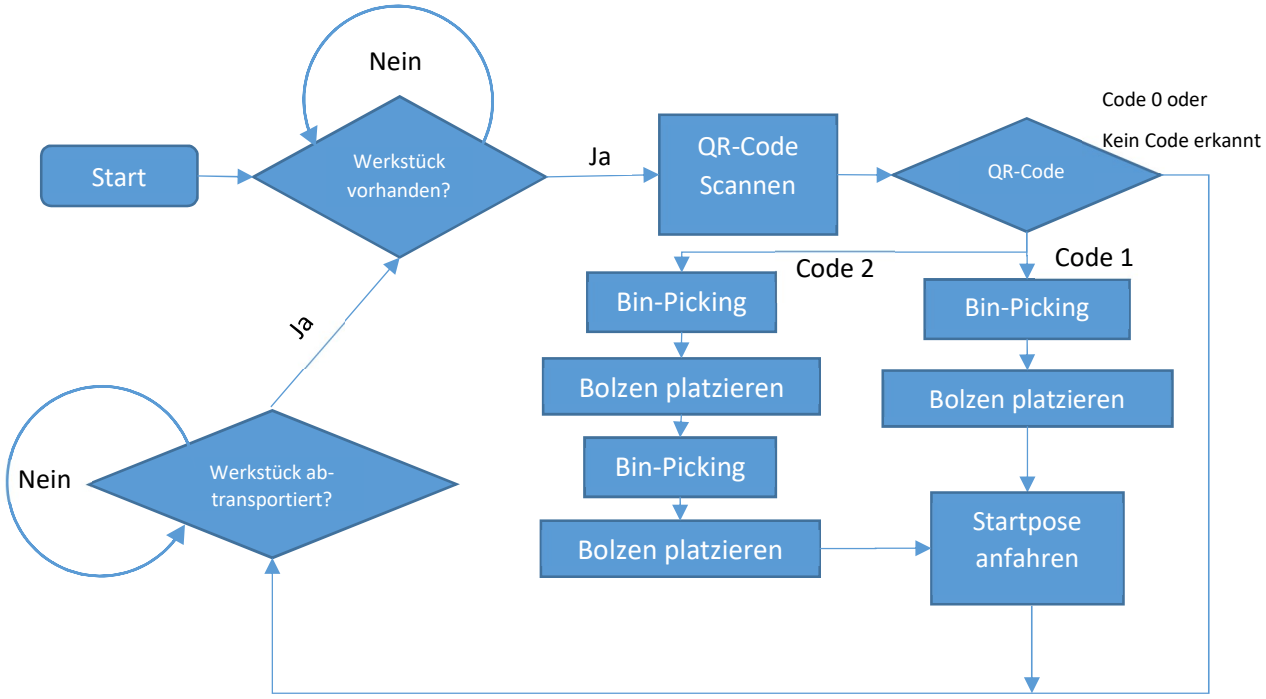


Abbildung 24: Altes Programmflussdiagramm

Hieran wurden einige Veränderungen vorgenommen. Zunächst wurde entschieden, dass die Bin-Picking-Station keinen separaten QR-Code Scanner benötigt. Stattdessen soll sie die Variante des Werkstücks von der SPS erhalten. Dies geschieht zum selben Zeitpunkt, zu dem auch das Eintreffen eines Bauteilträgers gemeldet wird. Des Weiteren wartet der Roboter nicht, bis das Werkstück abtransportiert wird, er löst den Abtransport selbst aus. Zu guter Letzt wurde festgelegt, dass der Roboter bereits einen neuen Bolzen greifen soll, bevor ein neuer Bauteilträger ankommt, um so die Zykluszeit zu reduzieren.

Das neue Flussdiagramm sieht wie folgt aus (Abbildung 25):

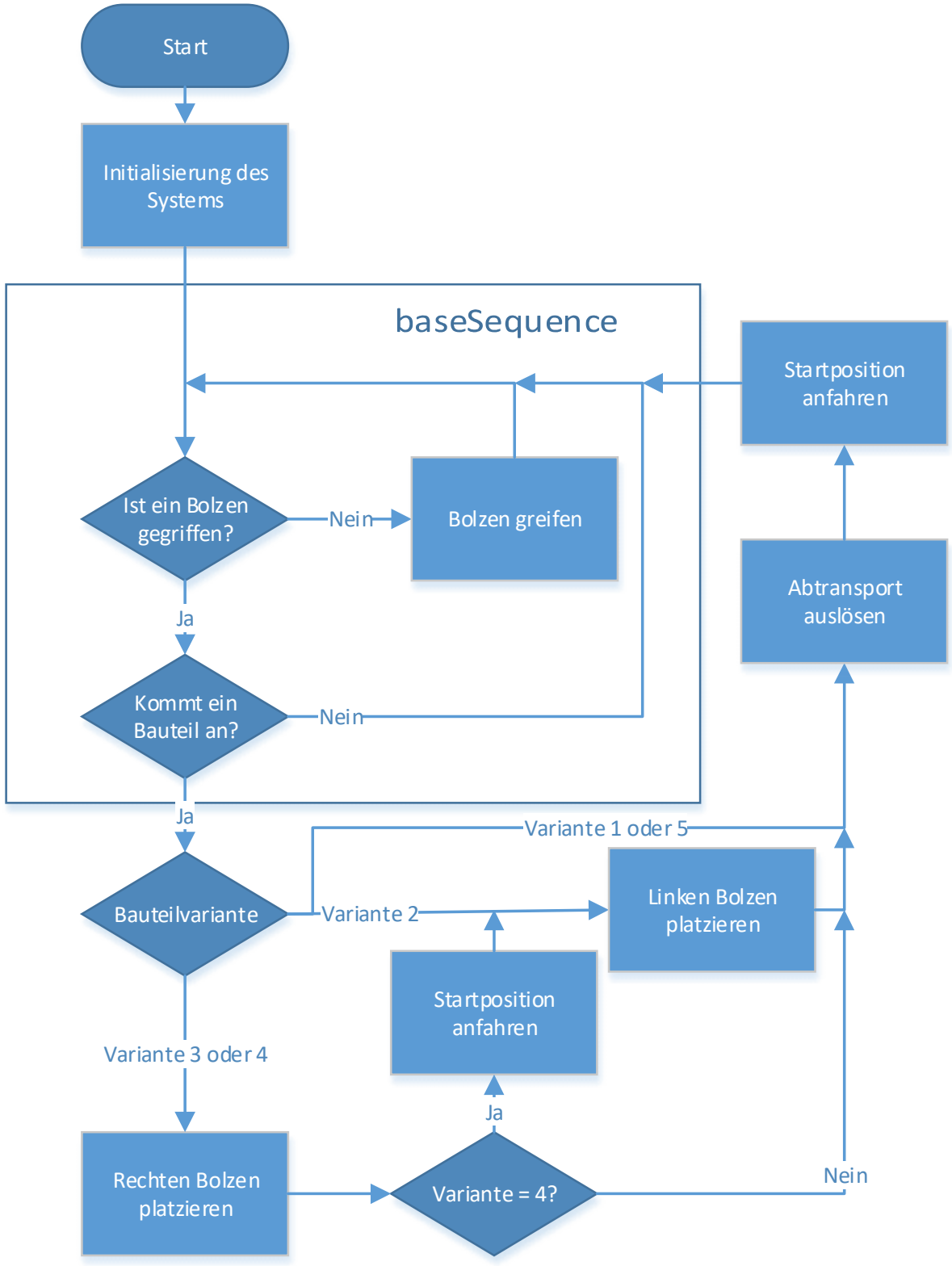


Abbildung 25: Neues Programmflussdiagramm

Wobei die „Bolzen platzieren“ Funktionen wie folgt aufgebaut sind (Abbildung 26):

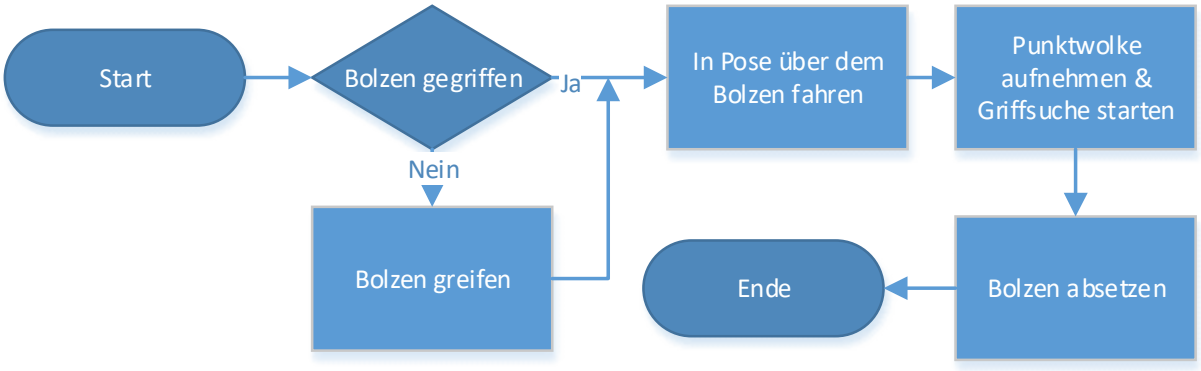


Abbildung 26: Flussdiagramm des Programmelements „Bolzen platzieren“

Nun soll detailliert werden, wie dieser Vorgang im Mikado-Programm umgesetzt wurde.

Die Initialisierung des Systems (Abbildung 25) findet in der Sequenz „Initialize“ statt. Hier wird die richtige Kamerakalibrierung geladen, der Bolzen als aktives Produkt gesetzt und die Kiste als aktive Szene. Der Roboter wird in eine Pose bewegt, in der er das Sichtfeld der 3D-Kamera nicht blockiert, danach wird eine Punktwolke aufgenommen und die Griffsuche gestartet. Abschließend wird der Roboter in eine Pose über der Kiste bewegt, so dass er griffbereit ist. Dann wird in die „baseSequence“ gesprungen.

Die „baseSequence“ besteht aus zwei Sequenzen. Die erste Sequenz überprüft, ob die boolean Variable „picked“ gesetzt ist. Wenn dem nicht so ist, wird in die „pick“ Sequenz gesprungen, andernfalls wird in der zweiten Sequenz überprüft, ob ein neues Bauteil eingetroffen ist. Trifft ein neues Bauteil ein, so wird „readVariation“ aufgerufen. Andernfalls wird kurz gewartet, bevor der Ablauf erneut gestartet wird. Das kurze Warten ist notwendig, damit das Programm nicht sofort unendlich viele Wiederholungen von „baseSequence“ in die Warteschlange einreiht, was zu Blockade und Absturz des Programms führt.

„readVariation“ überprüft die Variante des eingetroffenen Bauteils, und springt dementsprechend in die Routine zum Setzen des linken oder des rechten Bolzens (Abbildung 25).

Die Routinen zum Setzen eines Bolzens bestehen aus zwei Sequenzen. Die erste Sequenz überprüft, ob die „picked“ Variable gesetzt wurde. Ist dem nicht so, so wird in die „pick“ Sequenz gesprungen. Falls „picked“ gesetzt ist, fährt der Roboter mit dem

Bolzen über den Bauteilträger. Dort nimmt er eine Punktwolke auf und startet die Griffsuche. Anschließend setzt er den Bolzen ab. Die Routine zum Setzen des rechten Bolzens überprüft abschließend noch die Variante. Ist die Variante 4, so wird in die Sequenz zum Setzen des linken Bolzens gesprungen. Wenn dem nicht so ist wird der Bauteilabtransport ausgelöst und der Roboter fährt wieder in seine Grundpose. Danach wird in die „baseSequence“ gesprungen. Die Routine des linken Bolzens geht immer nach dem zweiten Schema vor, schickt also das Werkstück ab und fährt in die Grundpose zurück.

Die „pick“ Routine wird immer genutzt, wenn ein Teil gegriffen werden muss. Bevor ein Task diese Routine startet, setzt er immer die Variable lastTriedBolt. Ruft die „baseSequence“ „pick“ auf, so wird lastTriedBolt auf 0 gesetzt, beim rechten Bolzen auf 1 und beim linken auf 2. „pick“ versucht dann, einen Bolzen zu greifen, und nutzt anschließend lastTriedBolt, um wieder in die aufrufende Sequenz zu springen („baseSequence“, „rightBolt“ oder „leftBolt“). Wenn es beim Griffversuch zu Fehlern kommt, wird in eine entsprechende Routine gesprungen. Falls kein Objekt oder kein ausführbarer Griff gefunden werden konnte, fährt der Roboterarm auf einer festgelegten Route durch die Kiste, mit dem Ziel die Bolzen neu zu mischen. Danach wird in die „pickFailed“ Sequenz gesprungen. Bei anderen Fehlern beim Griff wird direkt in diese Sequenz gesprungen. Hier wird der Roboter aus dem Kamerasichtfeld bewegt, eine Punktwolke aufgenommen und die Griffsuche gestartet. Danach wird eine Zählvariable erhöht. Wenn diese Variable zu hoch wird, wird das Programm unterbrochen und eine Meldung ausgegeben, andernfalls wird wieder in „pick“ gesprungen. Findet ein erfolgreicher Griff statt, so wird die Zählvariable zurückgesetzt. Dies soll verhindern, dass das Programm in Dauerschleife durchmischt, obwohl keine Bolzen mehr in der Kiste sind oder die Bolzen so liegen, dass die Mischbewegung nicht mehr durchmischt.

6.5. Probleme und deren Lösung

Im Folgenden sollen einige bei der Implementierung auftretende Probleme und der gewählte Lösungsweg erklärt werden.

Zu einem fortgeschrittenen Zeitpunkt wurde beschlossen, vom ursprünglich geplanten Sauggreifer auf einen Zwei-Backen Greifer zu wechseln. Der Greifer lässt sich im Mikado Assistant zwar beim Erstellen eines Projekts einstellen, jedoch nicht mehr im

Nachhinein ändern. Im gripper.yaml und gripper.xacro File lässt sich das jedoch bewerkstelligen. Es wird dafür ein neues temporäres Projekt mit Zwei-Backen-Greifer erstellt, und mithilfe der beiden so erstellten gripper.yaml und gripper.xacro Dateien die entsprechenden Dateien in dem eigentlichen Projekt umgeschrieben. Mit Erfolg.

Bei den ersten Tests des Bin Pickings kam es zu unerwartetem Verhalten und Kollisionen. Nach langem Suchen der Ursache stellte sich heraus, dass die Bemaßung des Modells in der an unsere Tätigkeiten vorangehende Studienarbeit komplett inkorrekt vorgenommen wurde. Nicht nur haben die Maße nicht gestimmt, auch das Seitenverhältnis der Tischoberfläche sowie die Positionierung der einzelnen Objekte war inkorrekt. Nach einem langen Tag des Messens konnte jedoch sowohl die vorangehende Studienarbeit als auch das Projekt dieser Studienarbeit mit den korrekten Bemaßungen zum Laufen gebracht werden. Dies ermöglicht nun auch einen komplett kollisionsfreien Betrieb der alten Studienarbeit. Dies war davor nicht garantiert, da die Bemaßung des Roboters von Universal Robots gestellt, und damit korrekt, war, die der Station jedoch nicht.

Bei verschiedenen Versionen des Programms war es nötig, den Roboter auf der linken Seite auch außerhalb der Zellbegrenzung zu Bewegen. Auch zum Einsetzen des Bolzens muss der Roboter außerhalb der hinteren Zellbegrenzung operieren. Es ist im Assistant möglich, die Kollision mit den Zellbegrenzungen zu erlauben. Dabei stößt man jedoch auf zwei Probleme:

1. Es ist nicht möglich, Kollisionen des gegriffenen Objekts mit anderen Objekten zu erlauben.
2. Die Zellbegrenzungen würden damit obsolet. Sie existieren, um zu verhindern, dass der Roboter gegen Dinge außerhalb seines Zellbereichs stößt.

Aus diesen Gründen wurde sich dafür entschieden, die linke Zellbegrenzung 20 cm weiter außen zu platzieren als das physische Tischende, und auch die Rückwand 30cm weiter nach hinten zu bewegen, so dass der Roboterarm uneingeschränkt auf der der Station zugewandten Seite des Förderbands agieren kann. Es wurden auch Förderband und Bauteilträger in der Zellkonfiguration ergänzt.

Um die Bolzen abzusetzen, muss sich der Roboterarm im Bereich über dem Förderband bewegen. Wenn mehrere Bauteilträger hintereinander an der Station ankommen, wie ursprünglich vorgesehen, stößt der Roboterarm gegen diese und kann den

Bolzen nicht einsetzen. Darum wurde mit Gruppe 4 (Pick-and-Place-Station), die den Bauteilträger vor der Bin-Bicking-Station bearbeitet, gesprochen und gemeinsam eine Lösung ausgearbeitet. Die Bin-Bicking-Station teilt der SPS mit, wenn ein Bauteil fertiggestellt wurde, woraufhin dieses auf dem Förderband wegfährt. Dasselbe Signal setzt nun in der SPS ein RS-Flipflop. Dieses Flipflop wird zurückgesetzt, wenn die Pick-and-Place-Station einen neuen Bauteilträger zur Bin-Picking-Station schickt. Die Pick-and-Place-Station schickt nur einen Bauteilträger ab, wenn das Flipflop gesetzt ist.

6.6. Anbindung an die SPS

Die Anbindung an die SPS soll per PROFINET erfolgen. Die PROFINET-Schnittstelle ist zwar nicht im Standard-Lieferumfang von Mikado enthalten, allerdings wurde bei der Erstinbetriebnahme der Station tatsächlich eine solche Schnittstelle verbaut. Um die Nutzung der Schnittstelle zu ermöglichen, wurde zunächst Kontakt zum Mikado-Support aufgenommen. Es stellt sich heraus, dass die Station Mikado 2 benutzt, PROFINET jedoch erst mit Mikado 3 möglich ist. Zunächst wird also mithilfe des Mikado-Supports das System auf Mikado 3 aktualisiert.

Als nächstes muss ein Konzept erarbeitet werden, nach dem die Kommunikation stattfindet. Da nur wenig Kommunikation stattfinden muss, gestaltet sich dies als relativ einfach.

1. Mikado muss erfahren, dass ein neues Bauteil eintrifft, und welche Variante dieses hat. Es bietet sich an, für die Variante das standardmäßig definierte Wort ProfinetInput_ProductID zu nutzen. Dieses wird über das Setzen von ProfinetInput_Command = 6 mit Mikado synchronisiert.
2. Ursprünglich war geplant, das Eintreffen eines neuen Bauteils auch über die ProfinetInput_ProductID zu erkennen. Diese sollte, nachdem sie ausgelesen wurde, auf 0 gesetzt werden. Bei einer Synchronisation zwischen den Anlagen würde die ProductID neu gesetzt werden und das Programm könnte daran, dass die ProductID nicht mehr 0 ist, erkennen, dass ein neues Produkt eintrifft. Es stellt sich jedoch heraus, dass durch den Synchronisationsbefehl (ProfinetInput_Command = 6) nur Variablen synchronisiert werden, die in der Quelle

seit der letzten Synchronisation verändert wurden. Um dieses Problem zu umgehen, wurde das Programm so angepasst, dass das System anhand des Setzens von ProfinetInput_Command = 6 erkennt, dass ein neues Bauteil ankommt, und die ProfinetInput_ProductID nicht verändert.

3. Mikado muss der SPS mitteilen, dass das Bauteil fertig bearbeitet wurde. Hierfür wird die Variable ProfinetOutput_ProgrammErrorCode genutzt. Sie wird automatisch zwischen den Systemen synchronisiert, und die positiven Werte sind frei nutzbar. Wenn die Bearbeitung des Bauteils abgeschlossen ist, wird die Variable kurzzeitig auf 1 gesetzt, danach wieder auf 0.

Dieses Konzept muss nun zunächst auf Mikado-Seite umgesetzt werden, was keinen großen Aufwand darstellt.

Auf der SPS-Seite stellt sich das Ganze als etwas komplexer heraus. Anfangs gibt es Probleme, was die Einbindung der GDSML-Datei angeht. Mit Hilfe von Gruppe 2 (Steuerung und Vernetzung von Anlagen und Management-Systemen) und Prof. Kever kann dies aber erledigt werden. Das Senden der Variablen wird freundlicherweise von Jeffrey Rose von Gruppe 4 realisiert. Im Prozess stellt sich heraus, dass das Update auf Mikado 3 nicht vollständig ist, auch dieser Fehler kann aber mithilfe des Mikado-Supports behoben werden.

Damit ist die Integration der Bin-Picking-Station in das Produktionsnetzwerk sowohl auf der Hardware- als auch auf der Softwareebene abgeschlossen.

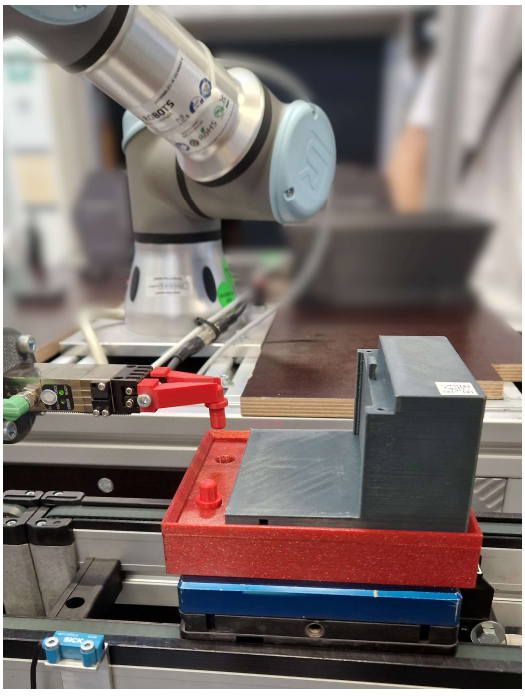


Abbildung 27: Erfolgreiches Platzieren der Bolzen

7. Machine Learning

In der vorangehenden Studienarbeit werden die verschiedenen Umsetzungen des Machine Learnings für die Implementation des Bin Pickings für das Greifen unbekannter Einzelteile aus dem Behälter diskutiert. Die meisten Machine Learning Algorithmen lassen sich in die Kategorien Supervised Learning, Unsupervised Learning, Semi-supervised Learning und Reinforcement Learning einteilen.

Für den Anwendungsfall der Bilderkennung beim Bin Picking ergeben sich Deep Learning und automatisiertes Machine Learning (AutoML) als die besten Ansätze.

Deep Learning basiert auf einem mehrlagigen neuronalen Netzwerk, das über verschiedene der kategorisierten Trainingsmethoden trainiert werden kann. Das Training ist sehr zeit- und ressourcenaufwändig, auch der Betrieb erfordert hohe Rechenleistungen, hat sich allerdings als Standard in der Bilderkennung durchgesetzt.

Die Nutzung von AutoML automatisiert den Erstellungsprozess eines Machine Learning Algorithmus in Hinsicht auf Datenvorbereitung, Modellauswahl und Optimierung. Das Verwenden von AutoML reduziert den benötigten Zeitaufwand erheblich.

Dabei werden die folgenden Anforderungen an den Algorithmus gestellt.

1. Der Algorithmus soll mit Machine Learning Methoden realisiert werden.
2. Er soll in der Lage sein, verschiedene Bolzen zu erkennen. Das bedeutet:
   2.1. Der Input des Algorithmus ist ein zweidimensionales Kamerabild.
   2.2. Der Output des Algorithmus ist eine Liste der erkannten Bolzen.
      2.2.1. Diese Liste soll in eine .txt Datei geschrieben werden.
      2.2.2. Inhalt dieser Liste sind die Namen der erkannten Bolzen in der Form M<Durchmesser in mm>x<Länge in mm>, also beispielsweise M5x40. Auch die Anzahl der jeweiligen erkannten Bolzen wird benötigt.
      2.2.3. Die Liste sollte übersichtlich formatiert sein.
   2.3. Die Größe der Bolzen, die erkannt werden müssen, ist auf einen sinnvollen Bereich begrenzt. Es werden keine Bolzen schwerer als 1 kg benötigt, und keine die länger als 15 cm sind.
3. Es gelten generelle Qualitätsanforderungen:
   3.1. Der Algorithmus soll so einfach wie möglich sein.
   3.2. Der Algorithmus soll so schnell wie möglich arbeiten.

8. Fazit und Ausblick

Der erste Teil der Bearbeitung der Studienarbeit war stark von theoretischer Natur. Es fand umfangreiche Recherche zu den Fachgebieten und Anwendungsfällen des Bin-Picking-Prozesses und den vielen Unterkategorien des Machine Learnings statt. Zudem wurde sich mit der Bin-Picking-Steuerung über Mikado Control sowie der damit verbundenen Software vertraut gemacht. Zusätzlich wurde die Bin-Picking-Station umgebaut und auf die Integration in das Produktionsnetzwerk vorbereitet.

Im zweiten Teil der Bearbeitung wurde ein Programm entwickelt, das die Steuerung des Cobots übernimmt und die Anforderungen des Produktionsnetzwerks umsetzt. Die Anbindung an die SPS wurde umgesetzt, wodurch die Prozessintegration in das Produktionsnetzwerk vollständig umgesetzt wurde.

Zusätzlich wurde ein digitaler Zwilling der Bin-Picking-Station erstellt und nach Eingliederung in den digitalen Zwilling des gesamten Produktionsnetzwerks im PTZ Simulationen des Produktionsnetzwerks ermöglicht.

Abschließend wurden Anforderungen an einen Machine Learning Algorithmus formuliert, um das Greifen unbekannter Einzelteile aus dem Behälter zu ermöglichen. Die Implementation eines Machine Learning Algorithmus konnte aus zeitlichen Gründen leider nicht fertiggestellt werden.

Die Entwicklung des Produktionsnetzwerks wurde erfolgreich abgeschlossen, zur Abschlussveranstaltung am 28.06.2023 fand ein erfolgreicher einwandfreier Durchlauf der Produktion statt.

Die Fertigung des „Smart Product“ zeigt die Grundlage für eine zielgerichtete, praxisorientierte Forschung und bietet weiterhin einen hochinteressanten Anwendungsfall, in dem Technologiethemen der Industrie 4.0 erforscht und umgesetzt werden können.

Auch die Bin-Picking-Station selbst bietet nach Abschluss dieser Studienarbeit weiteres Potential für kommende studentische Tätigkeiten und Forschungen. Weiterhin kann die Bin-Picking-Station für den Laborversuch verwendet werden. Während der gesamten Integration in das Produktionsnetzwerk wurde die Unversehrtheit des Laborversuchs gewährleistet. So können die technischen Möglichkeiten der zukunftsweisenden Technologie Bin Pickings Studenten nähergebracht werden und mit den aus den Vorlesungen erworbenen Kenntnissen verknüpft werden.

Eine mögliche Anknüpfung an diese Studienarbeit stellt die Entwicklung eines Machine Learning Algorithmus, der das Greifen unbekannter Bauteile ermöglicht, dar. Erste Anforderungen und Grundlagen wurden in dieser sowie in der vorangegangenen Studienarbeit formuliert.

Eine weitere Anknüpfungsstelle bietet das optische 2D-Mikrometer TM-3000 des Herstellers Keyence, das in den erläuterten Laborversuchen für die Qualitätsprüfung von Drehteilen verwendet wird. Dieses kann auch in Bin-Picking-Prozesse ohne Qualitätsprüfung eingebunden werden. Ein Beispiel stellt die Prüfung, ob das Einzelteil richtig gegriffen wurde, dar. Eine Griffsicherheit von 100% kann bisher nicht gewährleistet werden, durch die Einbindung des 2D-Mikrometers kann die Prozesssicherheit somit mit geringen Abstrichen bezüglich der Zykluszeit auf das Maximum gesteigert werden.

Abschließend lässt sich sagen, dass die Technologie des Bin Pickings interessante Forschungen ermöglicht und als zukunftsweisende Technologie eine Vielzahl von Anwendungsmöglichkeiten für die DHBW aber auch für moderne Produktionsanlagen bietet.

9. Literaturverzeichnis

[1] DHBW Ravensburg: Produktionsnetzwerk Fallenbrunnen
Online verfügbar unter https://www.ravensburg.dhbw.de/forschung-transfer/kompetenzzentren/zentrum-fuer-digitalisierung-in-produktion-und-produktentwicklung#zukunftsfabrik-bodensee, zuletzt geprüft am 09.07.2023.

[2] Konradin-Verlag Robert Kohlhammer GmbH: Industrieroboter - Grundlagen, Fakten, Märkte und Hersteller von Industrierobotik
Online verfügbar unter https://automationspraxis.industrie.de/industrierobotik/industrieroboter-grundlagen-fakten-und-hersteller-von-industrierobotik, zuletzt geprüft am 09.07.2023

[3] Statista Research Department: Absatz von Industrierobotern weltweit in den Jahren 2004 bis 2021 und Prognose bis 2025
Online verfügbar unter https://de.statista.com/statistik/daten/studie/29386/umfrage/weltweiter-absatz-fuer-industrie-roboter-seit-2004, zuletzt geprüft am 10.07.2023

[4] Industrie Wegweiser: Industrieroboter – weltweit 1,7 Millionen neue Roboter in der Industrie
Online verfügbar unter https://industrie-wegweiser.de/industrieroboter, zuletzt geprüft am 10.07.2023

[5] TeDo Verlag GmbH: Cobot-Markt 2021 um 45% gewachsen
Online verfügbar unter https://robotik-produktion.de/allgemein/cobot-markt-in-2021-um-45-gewachsen, zuletzt geprüft am 10.07.2023

[6] Statista Research Department: Robotik - Marktdatenanalyse und Prognosen
Online verfügbar unter https://de.statista.com/statistik/studie/id/116785/dokument/robotics-report, zuletzt geprüft am 10.07.2023

[7] TeDo Verlag GmbH: Bin-Picking-System mit KI-Vision
Online verfügbar unter https://robotik-produktion.de/allgemein/bin-picking-system-mit-ki-vision-2/, zuletzt geprüft am 10.07.2023.

[8] Konradin-Verlag Robert Kohlhammer GmbH: Bin Picking - Roboter beim Griff in die Kiste
Online verfügbar unter https://automationspraxis.industrie.de/robotik/bin-picking-roboter-fuer-den-griff-in-die-kiste/, zuletzt geprüft am 11.07.2023

[9] B+M Blumenbecker GmbH: Bin Picking. Automatische Objekterkennung und -positionierung.
Online verfügbar unter https://www.blumenbecker.com/de/industrie-automation/industrierobotik/bin-picking, zuletzt geprüft am 11.07.2023.

[10] Omega Engineering GmbH: Greifer - Einführung zum Thema Greifer
Online verfügbar unter https://www.omega.de/prodinfo/greifer.html,
zuletzt geprüft am 12.07.2023.

[11] Konradin-Verlag Robert Kohlhammer GmbH: Alles im Griff – Greifer für Roboter
Online verfügbar unter https://automationspraxis.industrie.de/handling/greifer-fuer-roboter-grundlagen-funktion-und-hersteller, zuletzt geprüft am 12.07.2023.

[12] COVAL VakuumTechnik GmbH: Verfahren zur Vakuumerzeugung
Online verfügbar unter https://www.coval-germany.com/Vakuumtechnik/Leitfaden-f%C3%BCr-das-Greifen-mit-Vakuum/Verfahren-zur-Vakuumerzeugung,
zuletzt geprüft am 11.07.2023.

[13] J. Schmalz GmbH: Vakuum-Sauggreifer:
Online verfügbar unter https://www.schmalz.com/de-de/vakuumtechnik-fuer-die-automation/vakuum-komponenten/vakuum-sauggreifer, zuletzt geprüft am 11.07.2023

[14] NavVis GmbH: Alles, was Sie über Punktwolken wissen müssen
Online verfügbar unter https://www.navvis.com/de/blog/everything-you-need-to-know-about-point-clouds-navvis, zuletzt geprüft am 13.07.2023

[15] Zivid A/S: Unlocking the Full Potential of Bin Picking with the Right 3D Camera
Online verfügbar unter https://blog.zivid.com/unlocking-the-full-potential-of-bin-picking-with-the-right-3d-camera, zuletzt geprüft am 13.07.2023

[16] Universal Robots A/S: UR3 User Manual, Odense, 2018

[17] Optonic GmbH: Handbuch mikado control Version 2.1, Freiburg, 2022.

[18] Optonic GmbH: documentation_profinet_interface_0.3.0+42-branch-0.3.6-a867f38, Freiburg, 2023.

[19] Unity Technologies: Was ist der digitale Zwilling?
Online verfügbar unter https://unity.com/de/solutions/digital-twin-definition, zuletzt geprüft am 14.07.2023

[20] IBM Deutschland GmbH: Was ist ein digitaler Zwilling?
Online verfügbar unter https://www.ibm.com/de-de/topics/what-is-a-digital-twin, zuletzt geprüft am 14.07.2023

[21] MarketsandMarkets™: Digital Twin Industry worth $110.1 billion by 2028
Online verfügbar unter https://www.marketsandmarkets.com/PressReleases/digital-twin.asp, zuletzt geprüft am 14.07.2023

[22] MarketsandMarkets™: Digital Twin Market by Application
Online verfügbar unter https://www.marketsandmarkets.com/Market-Reports/digital-twin-market-225269522.html, zuletzt geprüft am 14.07.2023

[23] Bosch-Rexroth AG: MTpro Online Designer. Online verfügbar unter https://www.boschrexroth.com/en/pl/products/product-groups/assembly-technology/topics/engineering-software-mtpro/mtpro-online-designer/, zuletzt geprüft am 13.01.2023.

[24] Siemens AG: Mechatronic Concept Design
Online verfügbar unter https://plm.sw.siemens.com/de-DE/nx/cad-online/automation/mechatronic-design, zuletzt geprüft am 15.07.2023

[25] U. Harten, Physik: Eine Einführung für Ingenieure und Naturwissenschaftler, Berlin: Springer Berlin, 2017.