

Diplomarbeit

# **Abstraktion verteilter Produktions- maschinen in cyber-physischen Pro- duktionssystemen**

10. September 2016

Peter Heisig

Matr.-Nr.: 3521226

**Betreuer**

Dipl.-Medieninf. Gordon Lemme,  
Dr.-Ing. Sebastian Götz

**Verantwortl. Hochschullehrer**

Prof. Dr. Uwe Aßmann

## **Erklärung**

Hiermit erkläre ich, dass ich die vorliegende Arbeit mit dem Titel

### **Abstraktion verteilter Produktionsmaschinen in cyber-physischen Produktionssystemen**

unter Angabe aller Zitate und nur unter Verwendung der angegebenen Literatur und Hilfsmittel selbstständig angefertigt habe.

Dresden, den 10. September 2016

*Peter Heisig*



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Zielsetzung . . . . .	2
1.3	Methode und Aufbau . . . . .	2
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Produktion und deren Automatisierung . . . . .	3
2.1.1	Kommunikation . . . . .	3
2.2	Informationsmodelle in der Fertigungsindustrie . . . . .	5
2.2.1	OPC Unified Architecture . . . . .	5
2.2.2	MTConnect . . . . .	6
2.3	Kontrolle & Überwachung von Produktionsmaschinen . . . . .	7
2.4	Cyber-physische Produktionssysteme . . . . .	7
2.5	Cloud Manufacturing & Fog-Computing . . . . .	8
2.6	Zusammenfassung . . . . .	8
<b>3</b>	<b>Anforderungen</b>	<b>9</b>
<b>4</b>	<b>Forschungsstand</b>	<b>11</b>
4.1	Maschinendatenerfassung & -analyse . . . . .	11
4.1.1	Legacy Machine Monitoring Using Power Signal Analysis [DP11]	11
4.1.2	Diss: In-process tool condition monitoring systems in CNC turning operations [Lee06] . . . . .	12
4.1.3	Machine-readable data carriers – a brief introduction to au- tomatic identification and data capture [Fur00] . . . . .	13
4.1.4	Zusammenfassung . . . . .	14
4.2	Rechnergestützte numerische Steuerung . . . . .	14
4.2.1	An ARM-based Multi-channel CNC Solution for Multi-tasking Turning and Milling Machines [GM16] . . . . .	14
4.2.2	Remote real-time CNC machining for web-based manufactu- ring [Wan+04] . . . . .	16

4.3	Architekturen & Methoden für CPPS . . . . .	17
4.3.1	Information Architecture for Reconfigurable production systems [Pau+13] . . . . .	18
4.3.2	Prototype OPC UA Server for Remote Control of Machine Tools [Aya+13] . . . . .	20
4.3.3	A systematic approach to OPC UA information model design [PFK16] . . . . .	21
4.4	Zusammenfassung . . . . .	23
<b>5</b>	<b>Konzeption</b>	<b>25</b>
5.1	Virtuelle Maschinenrepräsentation . . . . .	27
5.2	Kommunikations- & Informationsmodell . . . . .	27
5.3	Systemintegration . . . . .	27
5.4	Zusammenfassung . . . . .	27
<b>6</b>	<b>Implementation</b>	<b>29</b>
6.1	Zusammenfassung . . . . .	29
<b>7</b>	<b>Evaluation</b>	<b>31</b>
7.1	Zusammenfassung . . . . .	31
<b>8</b>	<b>Zusammenfassung</b>	<b>i</b>
8.1	Schlussfolgerung . . . . .	i
8.2	Ausblick . . . . .	i
<b>A</b>	<b>Anhang</b>	<b>i</b>
	<b>Abbildungen</b>	<b>iii</b>
	<b>Programmcode</b>	<b>v</b>
	<b>Tabellen</b>	<b>vii</b>
	<b>Literatur</b>	<b>ix</b>

# 1 Einleitung

## 1.1 Motivation

Heutige Produktionseinrichtungen beherbergen Maschinen jeden Alters, die zu einem gemeinsamen System verwachsen müssen. Gerade ältere Modelle besitzen häufig keine Möglichkeit der Integration in die IT-Systeme einer modernen Fertigungsstrecke. Das schlichte Ersetzen dieser Altmaschinen ist aufgrund hoher Kosten meist keine Lösung. Jedoch behindern diese vorrangig die nahtlose Machine-To-Machine (M2M) Kommunikation durch fehlende Infrastrukturanbindung, womit die Kette von Bearbeitungsschritten für ein Produkt zahlreiche manuelle Eingriffe erfordert. Als Teil des Fertigungsprozesses besitzt eine Altmaschine keine Möglichkeit externer Kommunikation und kein *Application Programming Interface* (API) [DP11]. Bei jüngeren Konstruktionen treten Integrationsschwierigkeiten an anderer Stelle auf. So sind selbst bei bestehender Netzwerkfähigkeit geschlossene Soft- und Hardwarearchitekturen und fehlende Schnittstellen verantwortlich für eingeschränkte Überwachung und Steuerung, respektive für die Verhinderung von ökonomisch sinnvoller Automatisierung [DP11]. Weiterhin erschweren die unzureichende Umsetzung von Industriestandards und -normen die Integration der Maschinen [Wan+04].

Doch technische Komponenten, wie eine Netzwerkanbindung, sind nicht die einzigen Barrieren moderner Produktion. Fehlerbehaftete Kommunikationsmechanismen, sowie die Gefahr der Veräußerung betriebsinterner Daten, sind Probleme die heute gelöst werden können [Wan+04]. Auch erfordern sinkende Losgrößen und steigende Produktvariabilität eine flexible Automatisierung von Echtzeitüberwachung und -kontrolle verteilter, rekonfigurierbarer Fertigungssysteme [Wan+04]. Produktionseinrichtungen basierten bisher auf dem manuellen Sammeln und Verteilen von Daten für Überwachung, Steuerung und Wartung der Maschinen. Doch gegenüber den hohen Kosten, menschlichen Fehlern, dem teilweise schlechten Zugang zur Anlage und Aspekten der Datensicherheit, sind Automatisierungslösungen heute günstig und damit bereits Teil der Fertigungsindustrie [DP11].

Das händische Programmieren und die Überwachung von Fertigungsparametern geschieht meist an einem Terminal und erfordert die physische Anwesenheit einer

Fachkraft.

## 1.2 Zielsetzung

Aufteilung nach Introduction von [Lee06]

- Zweck
- Voraussetzungen
- Wirkung

Durch steigende Rechenleistung sind ARM-Prozessoren auf Einplatinencomputern in der Lage Mehrachsmaschinen zu kontrollieren [GM16].

## 1.3 Methode und Aufbau

## 2 Grundlagen

### 2.1 Produktion und deren Automatisierung

- Ebenen der Automatisierungspyramide
  - MES
  - ...

#### 2.1.1 Kommunikation

Zusammenfassung von [Pau+13]

- SPS mit digital inputs/outputs (DI/DO)
- Feldbusse (EtherCAT, ProfiBUS, CAN, etc.) => Adapter
  - trotz ICE 61158 untersch. Standards
- Ethernet Varianten (TCP, RPC, OPC)

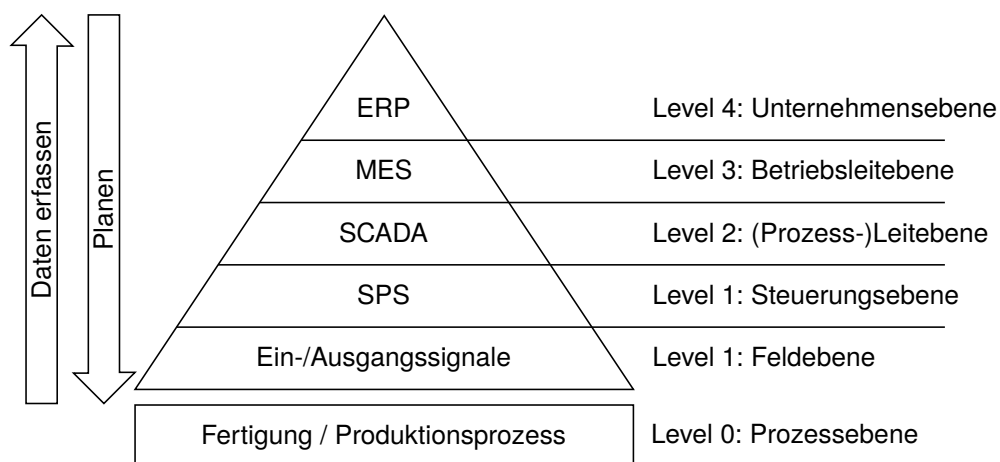
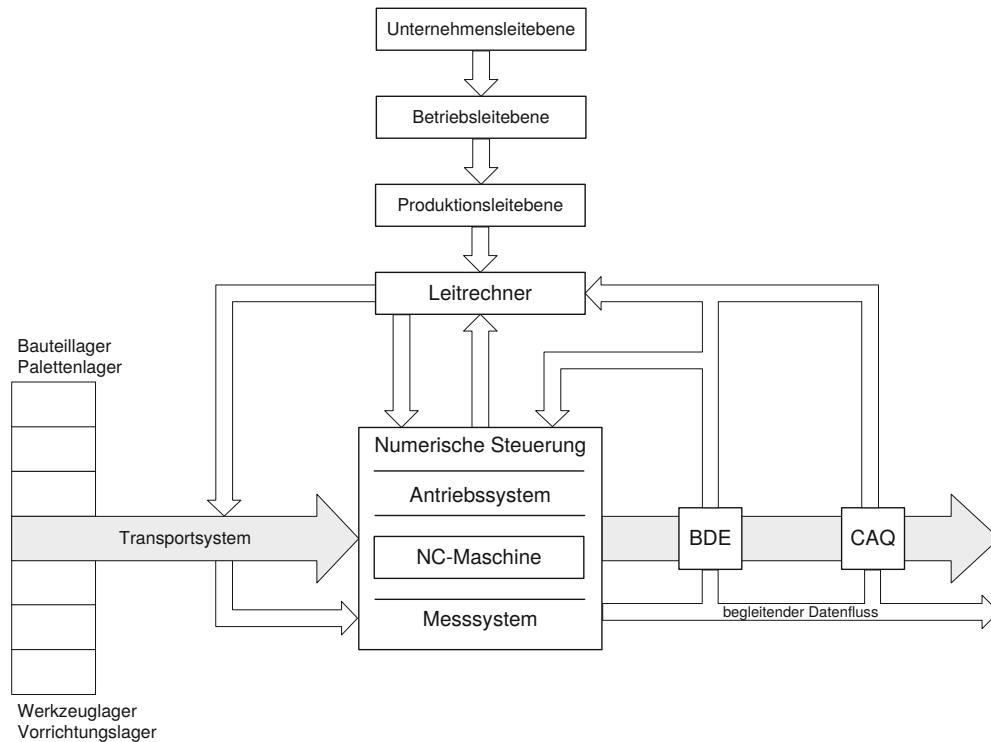


Abbildung 2.1: Beispiel einer klassischen Automatisierungspyramide<sup>2</sup>





**Abbildung 2.2:** Grundstruktur flexibler Automation [Lin15]

Value-Stream Mapping (VSM) ist eine Methode der *Lean Production*, mit der ein vollständiger Material- und Informationsfluss vom Zulieferer zum Endkunden abgebildet werden kann. Damit bietet VSM ein Maß tatsächlich benötigter Produktions- und Durchlaufzeit eines Produkts [MFT09].

G-code is considered a „dumb“ language as it only documents instructional and procedural data, leaving most of the design information behind. G-code programs are also hardware dependent, denying modern CNC machine tools desired interoperability and portability [XLY06].

In einer flexiblen Fertigungszelle (FFZ) befinden sich zwei oder mehr CNC-Maschinen, die im Verbund ein flexibles Fertigungssystem (FFS) bilden [Gro08].

Hersteller von Software für *Supervisory Control and Data Aquisition* (SCADA) verwalten eine große Anzahl an Kommunikationstreibern für unterschiedliche Automations- und Informationssysteme. Außerdem erschweren verschiedene Kommunikationsprotokolle und Nachrichtenformate die Integration zusätzlicher Systeme [Aya+13].

<sup>2</sup>Darstellung durch Wikipedia-Nutzer UlrichAAB

## 2.2 Informationsmodelle in der Fertigungsindustrie

Informationsmodelle sind Repräsentationen von Konzepten, Relationen, Beschränkungen, Regeln und Operationen zur Spezifikation der Bedeutung (Semantik) von Daten innerhalb einer bestimmten Domäne [Lee99].

### 2.2.1 OPC Unified Architecture

Die *OPC Unified Architecture* (OPC UA) ist ein semantischer Kommunikations- und Datenmodellierungsstandard für den Informationsaustausch via TCP/IP [Aya+13].

- Communication Technology that merges:
  - (1) Transport mechanism (uses internet standards XML, HTTP,... and also optimized binary TCP)
  - (2) Information modelling (using an extensible meta model)
- Extensible meta model
- Platform independent (cross-platform)
- Scalable
- But not hard real-time (not yet) => nicht geeignet für direkt Bewegungskontrolle [Pau14]

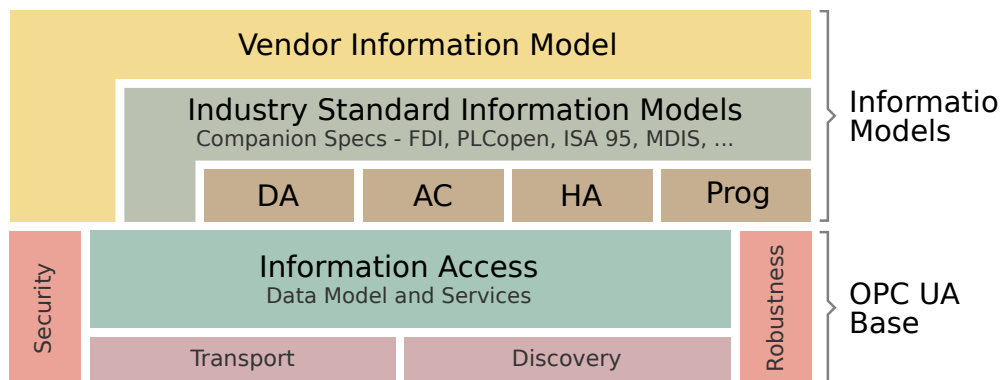


Abbildung 2.3: OPC UA Übersicht<sup>3</sup>

Die definierte Semantik des Address Space erlaubt nicht nur anspruchsvolle M2M-Kommunikation. Sie ermöglicht dem Operator einer FFZ Strukturinformationen einzusehen und die Automatisierungskomponenten zu kontrollieren [Aya+13].

Hoppe 2014 [Hop14]

<sup>3</sup>nach [opcfoundation.org/about/opc-technologies/opc-ua](http://opcfoundation.org/about/opc-technologies/opc-ua)

Obwohl bereits verschiedene wichtige Informationsmodelle, wie OPC-UA for Analyser Devices, FDI (Field Device Integration), ISA95, MT-Connect, BACnet und PLCopen existieren, oder in der Entstehung sind, gibt es hier noch Handlungsbedarf:

- Wie geben sich z. B. ein „Temperatursensor“ oder eine „Ventilsteuerung“ zu erkennen?
- Welche Objekte, Methoden, Variablen und Ereignisse definieren die

Schnittstelle für Konfiguration, Initialisierung, Diagnose und Laufzeit? OPC-UA hat das Potential, sich als De-facto-Standard für den Daten- und Informationsaustausch in der Automatisierungspyramide für nicht-echtzeitkritische Anwendungen zu etablieren. Eine sichere, horizontale und vertikale Kommunikation vom Sensor bis in die IT-Systeme ist damit bereits heute umsetzbar. Die Verbände BITCOM, VDMA und ZVEI werden durch die Industrie-4.0-Arbeitskreise keinen neuen Kommunikationsstandard definieren können; die Arbeitskreise bieten aber eine gute Grundlage zum Informationsaustausch.

Alternative: Woopsa (<http://www.woopsa.org/>)

### 2.2.2 MTConnect

- RO Standard for Process Information in CNC [Vij+08]

Wikipedia:

Data from shop floor devices is presented in XML format, and is retrieved from information providers, called Agents, using Hypertext Transfer Protocol (HTTP) as the underlying transport protocol. MTConnect provides a RESTful interface, which means the interface is stateless. No session must be established to retrieve data from an MTConnect Agent, and no logon or logoff sequence is required (unless overlying security protocols are added which do). Lightweight Directory Access Protocol (LDAP) is recommended for discovery services.

=> MTConnect-OPC UA Companion Specification

sonstige Standards...

## 2.3 Kontrolle & Überwachung von Produktionsmaschinen

- CNC
- STEP-NC [HL07; XN06]
- IEC 61499 Function Blocks

## 2.4 Cyber-physische Produktionssysteme

Die Verbindung von Überwachung und Kontrolle technischer Systeme mündet in Paradigmen, die Realität und virtuelle Umgebungen miteinander verschmelzen lassen. So wurde das Konzept cyber-physischer Systeme (CPS) 2006 durch Edward A. Lee erstmalig erläutert. Er versteht diese als Integration von Informationsverarbeitung und physischen Prozessen. Virtuelle und physische Abläufe werden durch Sensoren und Aktuatoren überwacht, beziehungsweise beeinflusst, stehen in unmittelbarer Wechselwirkung und sind durch Kontrollschleifen rückgekoppelt [Lee06]. Zeit und Nebenläufigkeit der realen Welt sind Eigenschaften die durch Infrastruktur- und Informationsabstraktionen abgedeckt werden müssen [WVG08]. Technologien wie Echtzeitbetriebssysteme, Middlewares und spezialisierte eingebettete Prozessorarchitekturen bilden den ersten Schritt zum Schließen dieser Lücke [Lee06]. Dennoch ist vor allem die inhärente Heterogenität der Komponenten eine Herausforderung für bestehende Kontrollmechanismen, Kommunikationsmuster und Softwareparadigmen [WVG08].

Im Kontext industrieller Produktionskontrolle ergeben sich neben den Herausforderungen für CPS (vgl. [Lee08]) weitere Anforderungen. Die autonome Kontrolle von Produktionsprozessen mit Hilfe von Kontrollschleifen wird durch Sensoren und Aktuatoren entlang der Produktionskette und an den individuellen Maschinen unterstützt. Dafür muss das Gesamtsystem in eigenständige Subsysteme mit gekapselten Rückkopplungsmechanismen gegliedert werden. Kommunikation geschieht auf der Basis bestehender Infrastrukturen mit kabelgebundenen und kabellosen Übertragungstechnologien. Die bisher eingesetzten Standards wurden auf der Prämisse homogener Teilnehmer entwickelt und müssen im Zusammenhang mit cyber-physischen Produktionssystemen (CPPS) überdacht werden. Ein drahtloses Sensornetzwerk ist eine Ausprägung einer solchen Infrastruktur innerhalb derer die Kommunikationspartner mit unterschiedlicher Bandbreite und Zuverlässigkeit arbeiten

müssen. Weiterhin sind auf diesen Komponenten verteilte Echtzeitoperationen für Kontrollschleifen unerlässlich, wodurch das Design bestehender Netzwerkprotokolle oft in Frage steht. So kann die Verarbeitung von Netzwerkpaketen, beziehungsweise deren Routing, Verifikation und Redundanz unvorhersehbar Zeit beanspruchen [WVG08].

## 2.5 Cloud Manufacturing & Fog-Computing

Aazam 2016 [AH16]

Im WAN problematisch [Sch+15] => OPC4Factory

## 2.6 Zusammenfassung

# 3 Anforderungen

Stakeholder?

- R1** Entfernte Kontrolle von Produktionsmaschinen
- R2** Überwachung von Zustand und Leistungsparametern
- R3** automatische Reaktion auf veränderte Bedingungen => Feedback => höherer Automatisierungsgrad



## 4 Forschungsstand

### 4.1 Maschinendatenerfassung & -analyse

#### 4.1.1 Legacy Machine Monitoring Using Power Signal Analysis [DP11]

**Problem:** Datenerfassung (=> Prozessüberwachung) bei Altmaschinen nicht vorhanden

**Lösung:** unabhängige minimalinvasive Sensorik

**Purpose.** Ziel von Deshpande et al. war eine nicht-invasive Methode der Echtzeitüberwachung von Energieverbrauch und weiteren Parametern bei Legacy-Maschinen.

**Design/Methodology/Approach.** Durch das Abgreifen des Stromverbrauchs über eine *Universal Power Cell* (UPC), dem Sensor an der Maschine, können Informationen via TCP und UDP an eine externe Komponente übergeben und ausgewertet werden. Die in Kilowatt eingehenden Verbrauchsdaten wurden durch an Bedingungen gekoppelte Schwellwerte in Status (an, aus, Leerlauf), Energieverbrauch, Werkzeugwechsel und Werkstückdurchsatz unterschieden. Für die Case Study und einen anschließenden Vergleich hatten Deshpande et al. auch moderne Maschinen mit der UPC ausgestattet. Verglichen wurde die zeitabhängige Auslastung von drei unterschiedlichen Modellen.

**Findings.** Mit einer Genauigkeit von 95% für den Status und 99% für Werkzeugwechsel und Durchsatz wurde das Konzept erfolgreich getestet.

**Research Limitations/Implications.** Das Konzept nutzt ausschließlich den Eingangsstrom der Maschine, wodurch Genauigkeit und Umfang der Daten begrenzt werden.

**Practical Implications.**

Für den Einsatz in einer realen Produktionsumgebung fehlen UI und API. Anforderungen an die Energieversorgung, sowie der Verbrauch von Teilsystemen der



Produktionskette werden messbar. Die periodische Planung der Wartung kann aufgrund tatsächlicher Nutzung und Auslastung geschehen. Außerdem werden Prinzipien wie VSM echtzeitfähig und ermöglichen eine höhere Produktivität.

**Originality/Value.** Die minimal-invasive Methode ist unabhängig von Hard- Software und erlaubt die autonome Aggregation von Informationen unzugänglicher Altmaschinen.

**Problem:** ein Sensor => zu wenig Infos

**Lösung:** verschiedene Sensoren (auch für Fusion) einsetzen

#### 4.1.2 Diss: In-process tool condition monitoring systems in CNC turning operations [Lee06]

- Purpose
  - Online Tool Condition Monitoring (TCM) bei CNC-Drehmaschinen mittels
    - \* Signal-Dekomposition
    - \* statischer Datenanalysen
    - \* neuronaler Netze
      - Untersuchung vieler versch. Netzwerkstrukturen => Optimierung für TCM
  - Schwerpunkt auf Analyse indirekter Messungen von Maschinensignalen
- Design/Methodology/Approach
  - Sensordaten durch Wavelet-Transformation in versch. Komponenten aufgeteilt, Quellen:
    - \* dreidimensionales Accelerometer
    - \* Schallemissionssensor
  - statistische Elimination maschinenspezifischer Effekte auf Sensordaten => Filter
  - statistische Signifikanzprüfung der Komponenten
  - Entwurf zweier Systeme basierend auf signifikanten Komponenten
    1. multipler linearer Regression
    2. künstlichen neuronalen Netzen
  - Fallstudie
- Findings
  - 12,6%ige Verbesserung der KNN-Vorhersagen durch Filter

- 90% Genauigkeit bei multipler Regression
  - 97% Genauigkeit bei KNN
- Research Limitations/Implications
  - ausschließliche Betrachtung von TCM
- Practical Implications
  - Reduktion von Maschinenstillstand durch verbesserte Vorhersage des Werkzeugausfalls
  - Minimierung von Ausschuss
  - Kostenreduktion
- Originality/Value
  - Umfassende Untersuchung aktueller Forschung zu MDE (Tabelle 2.1) und Entscheidungssystemen

#### 4.1.3 Machine-readable data carriers – a brief introduction to automatic identification and data capture [Fur00]

**Problem:** ausschließliche Betrachtung von TCM (Wartung) => Aber was wird wie bearbeitet?

**Lösung:** Werkstückidentifikation/-erkennung für Verknüpfung operativer Schritte

- Purpose
  -
- Design/Methodology/Approach
  -
- Findings
  -
- Research Limitations/Implications
  -
- Practical Implications
  -
- Originality/Value
  -

#### 4.1.4 Zusammenfassung

Ohne Carrier?

Data-Collection/Reasoning Stuff [Dow+16]

Anfallende Daten zu \* Prozess \* Technologie (CAD/CAM) \* Werkzeug \* Werkstück  
\* Peripherie (*loading door*)

Kommunikation der Informationen via OPC UA, MTConnect, MQTT ansprechen

- Umfangreiche Recherchen zu bestehender Forschung durch [Tet+10]
  - Werkzeugzustand
  - Spaneigenschaften
  - Prozesszustand
  - Werkstückbeschaffenheit

## 4.2 Rechnergestützte numerische Steuerung

**Problem:** Wissen über operative Schritte und Werkstück verfügbar - Umsetzung der Schritte?

**Lösung:** Kontrolle der Maschine

### 4.2.1 An ARM-based Multi-channel CNC Solution for Multi-tasking Turning and Milling Machines [GM16]

**Problem:** Kontrolle der Maschine durch SPS mit CNC auf Terminal-PC inflexibel und teuer

**Lösung:** Portierbarkeit des CNC-Kernels auf andere Systeme

- Purpose
  - Untersuchung von CNC mit ARM-Computern
    - \* Portierung eines CNC-Kernels auf Pi 2 durch Virtualisierung mit gemeinsamen Bibliotheken (Cross-Compile)
    - \* Kontrolle mehrerer paralleler Kanäle mit Schrittmotoren und Spindel

- \* Kommunikation über EtherCAT Feldbus (Echtzeit)
- Design/Methodology/Approach
  - Soft-SPS, RT-Linux auf ARM-Computer
  - ARM/PC-Anbindung
    - \* Spindel- und Motortreiber via EtherCAT
    - \* Feldbus-Koppler via EtherCAT
    - \* NC-Terminal via TCP
  - Fallstudie mit einer Dreh-/Fräsmaschine
  - Synch. der Kanäle durch High-Level-Funktionen (Load, Run, Start, Stop, Reset, Wait, Sleep)
- Findings
  - Ressourcen eines Einplatinencomputers sind ausreichend
  - parallele Portierung PC/ARM von NC-Software bei guter Kernel-Arch. mgl.
  - lediglich individuelle Konfiguration der Werkzeuge/Maschinen notwendig
  - CNC-Kernel braucht idle 20% ARM-CPU / 3% PC-CPU => steigt mit #Kanäle #Achsen
- Research Limitations/Implications
  - weitere Forschung auf Basis dessen
    - \* Verarbeitungspräzision/-Stabilität (Precision Engineering)
    - \* Adaptive Kontrolle
    - \* Diagnose/Prognose
- Practical Implications
  - EtherCAT Cycle Time<sup>1</sup> < 2ms, perspektivisch < 1ms => beeinfl. Anzahl paralleler Kontroll-Kanäle
  - künftig: Einplatinencomputer kontrollieren > 12 Achsen
- Originality/Value
  - Bewertung mit *Technology Readiness Level*<sup>2</sup> 6

---

<sup>1</sup>„[...] die Zeit, die ein Teilnehmer (slave) warten muß, bis er wieder ‚dran‘ ist.“ [Sch99].

<sup>2</sup>[www.nasa.gov/directorates/heo/scan/engineering/technology/txt\\_accordion1.html](http://www.nasa.gov/directorates/heo/scan/engineering/technology/txt_accordion1.html)

#### 4.2.2 Remote real-time CNC machining for web-based manufacturing [Wan+04]

**Problem:** Maschine ist nur am Terminal kontrollierbar; Datenerfassung und Auswertung nur vor Ort **Lösung:** Entfernter Zugriff

**Purpose.** Das Ziel von Wang et al. war die Entwicklung einer offenen Architektur für die Echtzeitüberwachung und -kontrolle von im Netzwerk befindlichen CNC-Maschinen.

**Design/Methodology/Approach.** Ein Web-basierter Thin-Client des *Wise-ShopFloor* ermöglicht die Kontrolle und Überwachung der Maschinen über ein dreidimensionales Modell der Fertigungsstrecke. Das darunterliegende Framework basiert auf einer Client/Server-Architekturstil und verwendet seitens des Servers das MVC-Entwurfsmuster. Maschinen werden über das Fabriknetzwerk mit dem Server verbunden und sind somit vom Internet getrennt. Bei der Verwendung mehrerer Clients wird für das Routing ein Publish/Subscribe Mechanismus über HTTP-Streaming eingesetzt. Mit Hilfe dessen wird das Verhalten des auf Java 3D basierenden Visualisierungsmodells durch Sensorik an den Maschinen beeinflusst. In der von Wang et al. durchgeführten Case Study wurde unter Verwendung einer CNC-Fräsmaschine die Tauglichkeit des Konzepts verifiziert. Die Schnittstelle zwischen Server und Maschine wurde durch einen *Open Architecture Controller*<sup>1</sup> bereitgestellt. Für die Kontrolle der Fräse kann zwischen einem manuellen und einem automatischen Modus gewählt werden, wobei letzterer die direkte Übertragung von G-Code ermöglicht.

##### **Findings.**

Das Internet ist ein zentraler Aspekt verteilter Produktion. Jedoch sind damit Sicherheitslücken fatal für interne Daten und vertrauliche Informationen der Organisation. Die gezielte Verbreitung dieser stellt ein erhöhtes wirtschaftliches Risiko dar und muss in besonderem Maße geschützt werden. Weiterhin sind Systemfehler auf Maschinenebene im Bezug auf Personen- und Materialschäden untragbar. Daher muss die reibungslose Kommunikation von Steuerungsbefehlen zu jeder Zeit gewährleistet sein.

**Research Limitations/Implications.** Standards für die Kommunikation von Sensor- und Steuerungsinformationen sind notwendig um Effizienz und Integration der Systeme zu vereinfachen. So müssen globale Schnittstellen definiert und durch die Komponenten des Systems implementiert werden. Durch die Verwendung eines zuverlässigen NC-Befehlsinterpreters ist die verteilte Echtzeitsteuerung von CNC-

---

<sup>1</sup>Steuerungskomponente, die Modifikationen über das API hinaus zulässt [Yon04]

Maschinen nach Wang et al. praktisch möglich. Jedoch setzt dieses System eine bestehende Anbindung an die Steuerungsebene voraus.

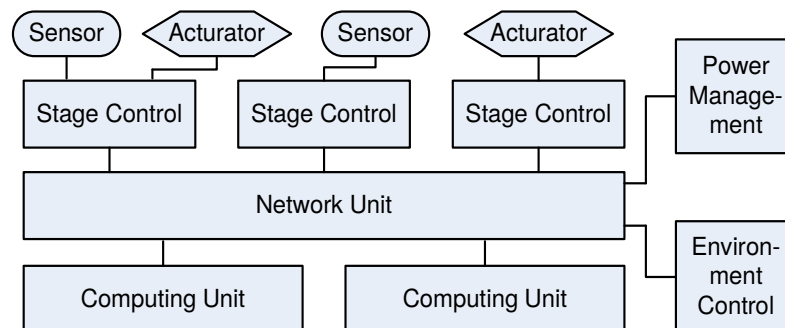
**Practical Implications.** Die direkte Verbindung des Clients zu einer Maschine ist mit der verwendeten Technologie nicht möglich. Sowohl die Java Sicherheitsinfrastruktur, als auch die Überwindung von Firewalls stellen zukünftig zu lösende Probleme dar. Für künftige Maschinen ist daher das Einbetten eines dedizierten Web-Services in die Kontrolleinheit notwendig.

**Originality/Value.** Ein wichtiger Aspekt des Konzepts von Wang et al. ist die technische Umsetzung auf der Java-Plattform. Mit dieser werden Sicherheitsinfrastrukturmerkmale wie byte-code-Verifikation und Rechtemanagement direkt unterstützt. Die Indirektion des Kontrollflusses über den Server der Architektur zu den Maschinen verhilft zur Einhaltung.

## 4.3 Architekturen & Methoden für CPPS

Grundsätzlich werden hoch skalierende gesamtheitlichen Systeme in Teilsysteme gegliedert, wodurch ein *System of Systems* (SoS) entsteht.

Abbildung 4.1 verbildlicht das Konzept der Kapselung und Verbindung einzelner Produktionskomponenten in einer Architektur für CPPS.



**Abbildung 4.1:** Komponentenbasierte CPS-Architektur nach Wang et al. [WVG08]

- SOA [Mey10; FWW16]
- MAS, Holonic [Lei09; Fal16]
- Blackboard [MVK06; Pau+13]
- RAMI4.0 [AE15]

### 4.3.1 Information Architecture for Reconfigurable production systems [Pau+13]

**Problem:** Betrachtungen des Retrofitting bzgl. Kontrolle und Überwachung einzelner Maschine, Integration?

**Lösung:** flexibel konfigurierbarer Verbund miteinander kommunizierender Maschinen => FFZ

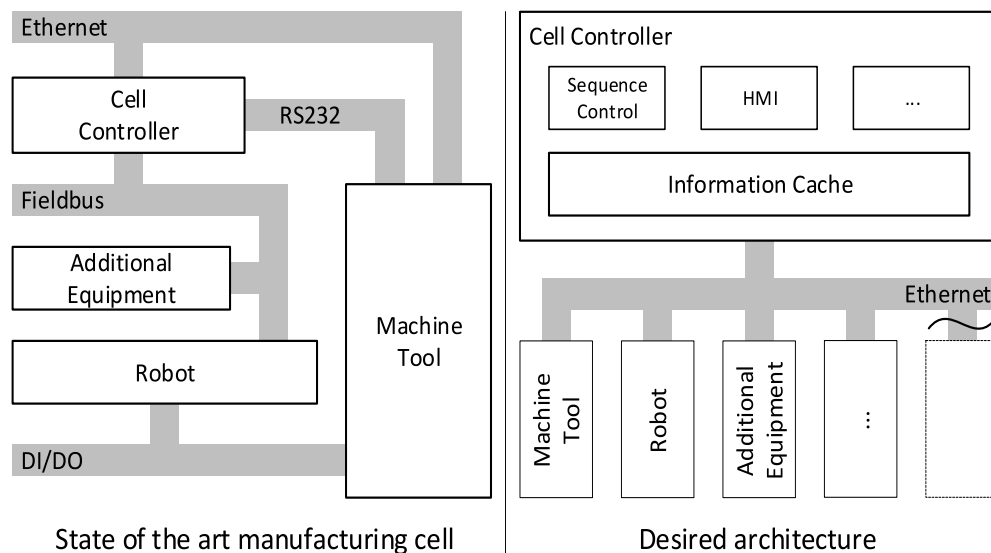


Abbildung 4.2: Vergleich der Architekturen von Fertigungszellen aus [Pau+13]

- Purpose
  - Montage und Konfiguration von FFZ bzgl. Informationsebene vereinfachen
  - Design von Informationsmodell und zentralem Daten-Cache
  - Design einer Kommunikationsarchitektur für die Komponenten einer FFZ
  - Rekonfiguration einer FFZ zur Laufzeit (Losgrößenproblematik)
  - wissensch. Experimente mit rekonfigurierbaren Fertigungssystemen
- Design/Methodology/Approach
  - VDMA 34180 als Arch.-Ref. für FFZ betrachtet
    - \* beschreibt Kommunikationsarchitekturen für *Robotized Manufacturing Cells* (RMC)
    - \* Schwerpunkt auf Werkstücklogistik mit Robotern
  - Blackboard-Architektur für Kommunikation
    - \* zentrale Datenbasis

- \* keine Kontrolleinheit (vgl. Blackboard in intelligenten Systemen)
- \* XML als Datenformat
  - Parameter für Kommunikation (IP, Port)
  - Maschinenstatus
  - Programm (Ident., Status)
  - Informationen zu angeschlossenen Geräten
- \* Konsistenz durch R/W-Locks
- \* Sequenzkontrollkomponente legt Handlungsweisungen auf Blackboard ab
- \* *Microsoft Visual Programming Language* (VPL) als Sequenzdefinitionsformat (Datenflussdiagramm)
- \* Maschinen und Sequenzkontrolle als aktive *Knowledge Sources* holen sich diese Instruktionen
- Master/Slave-Prinzip für Workflow und Job-Delegation (Master-Worker-Pattern)
- Ethernet-basierte Kommunikationsschnittstellen
- Adapter zu den Maschinen-Controllern (via Socket an Cell-Controller)
- Fallstudie durch FFZ mit zwei Fräsen und einem Roboter
- Findings
  - Reduktion von Komplexität der Topologie => geringerer Konfigurationsaufwand
  - VDMA 34180 nutzt DI/DO für Kommunikation => Konfiguration auf HW-Ebene => zu inflexibel für rekonfigurierbare RMCs
  - Fallstudie zeigt die Eignung der Architektur für FFZ
  - Blackboard geeignet bei notwendiger Austauschbarkeit der Komponenten
- Research Limitations/Implications
  - Schwerpunkt auf Blackboard-Informationscache
  - Eignung von VPL für Datenflussdefinition zu untersuchen
  - Performanz-Probleme mit XML als Blackboard-Backend (SQLite wird diskutiert)
    - \* R/W-Locks
    - \* Parsen der Datei
  - Socket-basierte Kommunikation Problematisch
    - \* proprietäre Kommunikationsprotolle => Adapter immer notwendig
    - \* OPC UA Server für beteiligte Komponenten
- Practical Implications



- Hinzufügen/löschen von HW-/SW-Komponenten ohne Überarbeiten der Informationsarch./-modelle
  - *Plug & Produce* für FFZ wird möglich
  - Einbinden einer Komponente erfordert Adapter
- Originality/Value
  - Blackboard den intelligenten Systemen entlehnt
  - Definition der Sequenz für den Cell-Controller

#### 4.3.2 Prototype OPC UA Server for Remote Control of Machine Tools [Aya+13]

**Problem:** Socket-basierte, proprietäre Maschinenkommunikation (Adapter) **Lösung:** OPC UA Server für Komponenten einer FFZ

- Purpose
  - OPC UA Server für CNC innerhalb einer FFZ
  - semantische Kommunikationsschnittstelle
- Design/Methodology/Approach
  - Implementierung eines OPC UA Informationsmodells für CNC-Maschinen
  - Methoden des Informationsmodells für Instruktionen verwendet
  - Kommunikation zw. Server und Maschine via propriät. Direct Numerical Control (DNC)
  - Case-Study an CNC-Drehmaschine & Industrieroboter
  - C++ UA Server SDK, .NET UA Client SDK (Unified Automation)
  - OPC UA Client mit Kontrollfunktionalität
- Findings
  - OPC UA ist eine geeignete Technologie für erweiterte Maschinenanbindung
  - Problem der Granularität des Modells (Ockhams Rasiermesser, KISS), 1-zu-1 Mapping ungeeignet
- Research Limitations/Implications
  - Definition eines Standard-Informationsmodells benötigt (vgl. MTConnect-OPC UA Comp.)
  - Anforderungen (horiz./vert. Integr.) für bestehende Standards (MES)
- Practical Implications

- Originality/Value
  - OPC UA Methoden als Steuerungsschnittstelle
  - dynamische (Runtime) Werkzeugrepräsentation im OPC UA Adressraum

### 4.3.3 A systematic approach to OPC UA information model design [PFK16]

**Problem:** OPC UA ist nicht alleiniger Standard für Informationsmodelle

**Lösung:** Informationsmodelle modellgetrieben entwickeln

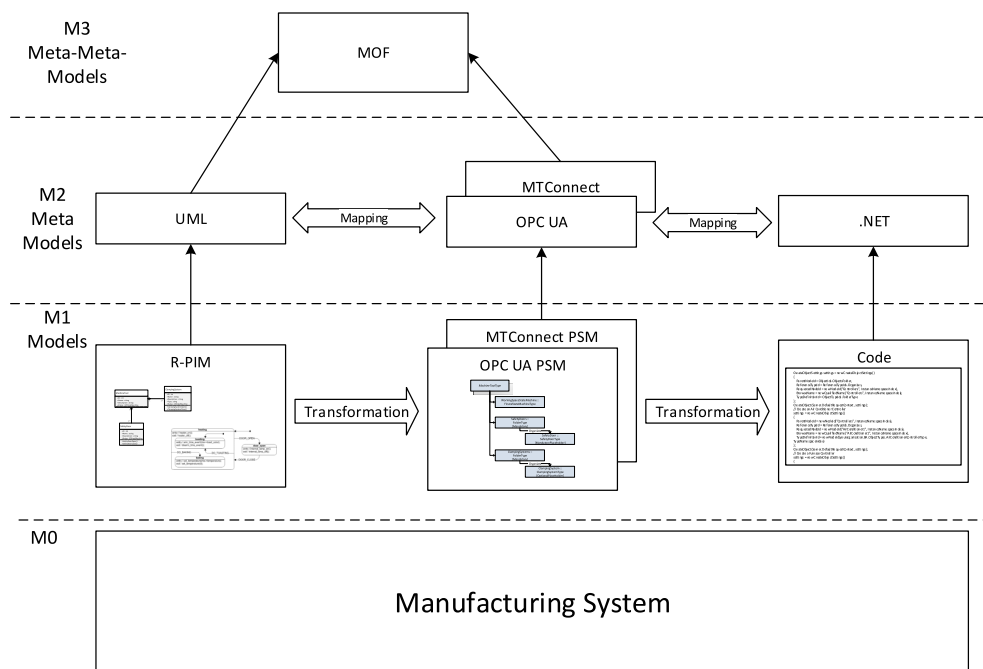


Abbildung 4.3: Transformationsprozess PIM zu PSM aus [PFK16]

- MOF?
- M0 == FFS
- FFS repräs. durch PIM, definiert in UML auf M2  
=> plattformunabh. Modellierung auf M1
- PIM => PSM: ein PIM-Elem. entspricht einem PSM-Elem. pro PSM (OPC UA, MTConnect, etc.)
- ein PSM kann dann in untersch. Sprachen (C#, C++, ...) übersetzt werden
- M3 Meta-Metamodell beschr. Sprache für spez. Metamodelle auf M2 => Interoperabilität zw. M2-Modellen

- Purpose
  - UML als einheitliche Modellierungssprache in der Softwareentwicklung, untersch. Modellierungssprachen in der Fertigungsindustrie
  - Model und Code müssen für jede Sprache separat entwickelt und gewartet werden => hoher manueller Aufwand
  - Konzept modellgetriebenen Designs (MDA) auf Fertigungsindustrie übertragen
    - \* automatisiertes Ableiten plattformspezifischer Modelle von plattformunabhängigen
  - Spezifikation eines Prozesses mit dem virtuelle Repräsentationen von I4.0-Komponenten generiert werden können
    - \* Diagramme und Tools für Modellierung und Code-Generierung
    - \* OPC UA Informationsmodelle von Systemanforderungen zu Codegenerierung und Deployment
  - RAMI4.0 Konformität
- Design/Methodology/Approach
  - erfassen statischer und dynamischer (Verhalten) Aspekte von Produktionssystemen
  - systematischer Ansatz für Modelltransformation zu OPC UA Informationsmodelle (hier: Zieltechnologie)
  - wichtigste plattformspez. Technologien untersucht => MDA Workflow definiert => Restriktionen für PIM definiert  
=> Restricted Platform Independent Model (R-PIM)
    - \* z.B.: Mehrfachvererbung in UML + nicht mgl. mit C#
    - \* z.B.: notwendige Attribute der OPC UA Knoten (NodeId, NodeClass, BrowseName, DisplayName)
    - \* untersch. zu MDA liegt in diesen Restriktionen
  - generisch bzgl. Zielmodell (OPC UA, MTConnect, etc.) <= Systemanalyse und -design mit der UML
    1. Komponenten- und Anwendungsfalldiagramme zur Beschreibung der Domäne
      - \* manuell, da Wissen von Domänenexperten/Systemnutzern benötigt
      - \* Komponentendiagramm: Systemgrenzen und Subsysteme identifizieren
      - \* Anwendungsfalldiagramme: abstrakte Systemfunktionalität

2. Domänenwissen => R-PIM
    - \* Klassendiagramme
    - \* Zustandsdiagramme
  3. R-PIM => Zieltechnologie
    - \* Adressraummodell
  4. Zieltechnologie => Code-Gerüst
  5. Implementierung der Logik in Komponenten (API)
  6. Deployment eines ausführbaren Artefakts
- Findings
    - fehlende APIs/Tools => noch viel manueller Aufwand
  - Research Limitations/Implications
    - Definition eines Meta-Metamodells für Restriktionen steht noch aus
    - manuelle R-PIM => PSM Transformation (exist. Konzept erlaubt nur statischer Aspekte des Address Space)
    - TU Wien arbeitet an Transf. dynamischer Aspekte => Guarded State Machines aus OPC UA Spec
    - vorerst nur OPC UA
    - Verbesserung der CIM-Definition
    - Ontologien als Wissensbasen statt Experten
  - Practical Implications
    - Adressraummodellierung, Logikimplementierung zum Großteil noch händisch (fehlende Tools und APIs)
  - Originality/Value
    - MDA-Prozess für generische Informationsmodelle durch UML

## 4.4 Zusammenfassung

Die Integration bestehender Hardware in die intelligente Steuerung einer Fabrik ist Thema des *RetroNet*-Projekts. Das Fraunhofer IPK entwickelt mit Industriepartnern physische und logische Adapter für die Anbindung von bestehenden Anlagen an eine Steuerungsplattform. Maschinen-, Anlagen und Produktionsdaten werden zu diesem Zweck zentral erfasst und gespeichert. Weiterhin soll eine Middleware im Client-Server-Architekturstil Dienste und zugrunde liegende Teilsysteme miteinander verbinden und eine vermittelnde Rolle im Gesamtsystem einnehmen [Fra16b].

Forschung im Bereich Cloud-basierter Industriesteuerung wird in Zusammenarbeit von Fraunhofer, der TU Berlin und Industriepartnern betrieben. Im Projekt *pI-CASSO* werden die Auslagerung von Steuerungsdiensten in die Cloud und Möglichkeiten einer Verteilung und Modularisierung herkömmlicher Kontrollsysteme auf CPS-Komponenten untersucht [Fra16a].

Im Projekt *OPC4Factory* der TU Wien, wurden generische OPC UA Informationsmodelle entwickelt. Diese verbessern die Konnektivität von NC-Maschinen, Industrierobotern und anderen Komponenten innerhalb einer flexibel automatisierten Fertigungszelle. Die Orchestrierung der Fertigungsoperationen, sowie die Konfiguration der Komponenten soll durch die Lösung der Schnittstellenproblematik vereinfacht werden<sup>1</sup>.

[Sch+15; Sch+14; Vic+15]

- TODO Projekte zusammenfassen & gegeneinander abgrenzen
- Schnittstellenproblematik immer Teil des Problems
- Entwicklung von Adaptern meist Standardlösung
- Kosten => Remote Maintenance?

---

<sup>1</sup>[www.ift.at/forschung/forschungsprojekte/opc4factory](http://www.ift.at/forschung/forschungsprojekte/opc4factory) see [Aya+13; Pau+13; Pau14]

## 5 Konzeption

- betrachtet
  - Fertigungsindustrie
  - CNC-Maschinen
- unbetrachtet
  - Adapter für proprietäre CNC-Protokolle (DNC)
- Laufzeitkonfiguration des Surrogate?
- Surrogate in bestehende Netzwerkinfrastruktur einbinden?
  1. direkte SG-Kommunikation (Wifi-Direct, BT, ...) zur Konfiguration der Netzwerkanbindung via Mobile device
  2. Auswahl des Access-Points
  3. Festlegen der OPC UA Discovery-Service Adresse
  4. SG mit Infrastr. verbunden + Registrierung bei Discovery-Service
  5. OPC UA Systemevent => neues Surrogate registriert
  6. Konfiguration des Adressraums via HTTP/WebApp
- Controlling nicht aus der Cloud @... sondern an der Maschine
  - SPS (und Motortreiber) als Teil des Surrogate
  - CNC-Kernel auf dediziertem Controller
    - \* Kernel muss nicht portiert werden (vgl. [GM16])
    - \*
  - Echtzeit kein Problem (OPC UA kann's eh nicht)
  - kein Feldbus, keine Koppler, kein DI/DO (vgl. [GM16])
  - Ethernet-basierte Kommunikation
- Konzept eines *Cell Controller* als Basis (vgl. [Aya+13; FWW16])

- kein Maschinenspez. NC-Terminal => verteiltes System => entfernte Mensch-Maschine-Schnittstelle (vgl. [GM16])
- Rekonfigurierbare FFZ [Pau+13]
- Bisher OPC UA Server als Adapter zu proprietären Maschinenprotokollen
  - Server <-> Maschine => Server <-> Adapter <-> Maschine ?
  - Smoothieboard/Embedded (soft-)SPS als Adapter zu Maschine
- Wiederverwendung [Aya+13]
  - des Informationsmodells
  - des Flow-Charts für die Server-Logik (teilweise)
- Persistenzkonzept: Blackboard? [Pau+13]
- Kontrolle der Arbeitssequenz? (PROtEUS, BPMN/Activiti)

#### Nach Rücksprache

- OPC UA zur Metamodellierung bzgl. der Maschine
- OPC UA Modell synchron mit Realität => Laufzeitmodell
- Zu erwartendes Verhalten des physischen Systems über FB-Loop (MAPE-K) kontrollierbar => Modellierung/Sprache der *Regeln*?
- Elemente eines Frameworks mit Schichtenarch. im Client/Server-Stil
- Microkernel-Ansatz (Plugins für OPC UA Typen, Sensoren und Aktuatoren)
- FB-Loop intern oder extern?

#### OPC4Factory:

OPC UA Server und ihre Informationsmodelle repräsentieren alle für die Automatisierungsaufgaben erforderlichen Komponenten der angeschlossenen Maschinen und Roboter (Ladetüren, Spannmittel, Werkzeuge, NC-Programme etc.) mit ihren Attributen, Ereignissen und Methoden. Die Kommunikation auf dieser Ebene **erfordert keine Echtzeitfähigkeit**, da Steuerungsaufgaben mit Echtzeitanforderungen ausschließlich innerhalb der Maschinen- bzw. Robotersteuerung abgewickelt werden.

- Statusreport: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0) [AE15]
- Kruchten 4+1?
- arc42?

## 5.1 Virtuelle Maschinenrepräsentation

- Service-/Protokolllayer (OPC UA)
- CPS-Layer, Hardwareanbindung
- 

## 5.2 Kommunikations- & Informationsmodell

[Pau+13] Komponentengranularität bis zur Achse (intelligente Maschinenteile)

## 5.3 Systemintegration

## 5.4 Zusammenfassung





## 6 Implementation

- Smoothieboard als Maschinen-Adapter
  - Nachteil: Beobachten des Prozessfortschritts langsam (*progress*) => kann nicht in online-FB einbezogen werden
- open62541 oder ähnliche OPC UA Stack-Implementierungen für Server auf Pi

### 6.1 Zusammenfassung



## 7 Evaluation

These/Behauptung? \* Steigerung des Automatisierungsgrads durch Feedbackloop \*  
physische Anwesenheit des Werkers technisch überwinden (Remote-Control/-Programming)  
\* Laufzeitmodell für *online*-Monitoring

Umsetzung? \* Proof of concept \* Case-Study mgl.? \* HIL-Simulation?

Blocking Factors/mögliche Kritik? \* Leistung von embedded computing devices =>  
siehe [GM16] \* Energieverbrauch

### 7.1 Zusammenfassung



# 8 Zusammenfassung

## 8.1 Schlussfolgerung

## 8.2 Ausblick

- Prozessmodell für Abstrakte Leitebene
- Wo läuft die Logik für orchestrierende Steuerung? (OPC UA Clients)
- Surrogate als reaktiver Agent => Einbindung in MAS denkbar
- Möglichkeiten des Nutzens der Daten
  - Welcher G-Code Befehl korreliert auf welche Weise mit welchen gemessenen Werten?
  - Automatische Erkennung von Gut-/Schlechtteilen

# A Anhang





# Abbildungen

2.1	Beispiel einer klassischen Automatisierungspyramide . . . . .	3
2.2	Grundstruktur flexibler Automation [Lin15] . . . . .	4
2.3	OPC UA Übersicht . . . . .	5
4.1	Komponentenbasierte CPS-Architektur nach Wang et al. [WVG08] .	17
4.2	Vergleich der Architekturen von Fertigungszellen aus [Pau+13] . . .	18
4.3	Transformationsprozess PIM zu PSM aus [PFK16] . . . . .	21



# Programmcode



# Tabellen



# Literatur

- [AE15] Peter Adolphs und Ulrich Epple. *Statusreport: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0)*. Technischer Bericht April. 2015 (siehe Seiten 17, 26).
- [AH16] Mohammad Aazam und Eui-Nam Huh. „Fog Computing: The Cloud-IoT/IoE Middleware Paradigm“. In: *IEEE Potentials* 35.3 (Mai 2016), Seiten 40–44. DOI: 10.1109/MPOT.2015.2456213 (siehe Seite 8).
- [Aya+13] I Ayatollahi, B Kittl, Florian Pauker u. a. „Prototype OPC UA Server for Remote Control of Machine Tools“. In: *International Conference on Innovative Technologies*. Band 1009. 12. 2013, Seiten 73–76 (siehe Seiten 4, 5, 20, 24–26).
- [Dow+16] Jonathan Downey, Denis O’Sullivan, Mirosław Nejmen u. a. „Real Time Monitoring of the CNC Process in a Production Environment- the Data Collection & Analysis Phase“. In: *Procedia CIRP* 41 (2016), Seiten 920–926. ISSN: 22128271. DOI: 10.1016/j.procir.2015.12.008. URL: <http://www.sciencedirect.com/science/article/pii/S2212827115010872> (siehe Seite 14).
- [DP11] Amit Deshpande und Ron Pieper. „Legacy Machine Monitoring Using Power Signal Analysis“. In: *ASME 2011 International Manufacturing Science and Engineering Conference, Volume 2*. ASME, 2011, Seiten 207–214. DOI: 10.1115/MSEC2011-50019 (siehe Seiten 1, 11).
- [Fal16] Solmaz Mansour Fallah. „Multi Agent based Control Architectures“. In: *26th DAAAM International Symposium on intelligent Manufacturing and Automation* (2016), Seiten 1166–1170. DOI: 10.2507/26th.daaam.proceedings.164 (siehe Seite 17).
- [Fra16a] Fraunhofer IPK. *Industrie 4.0 - Cloudbasierte Steuerungen*. 2016. URL: [https://www.ipk.fraunhofer.de/fileadmin/user\\_upload/IPK/publikationen/themenblaetter/aut\\_pICASSO\\_web.pdf](https://www.ipk.fraunhofer.de/fileadmin/user_upload/IPK/publikationen/themenblaetter/aut_pICASSO_web.pdf) (siehe Seite 24).
- [Fra16b] Fraunhofer IPK. „RetroNet - Praxisnahe Brücke in die Industrie 4.0“. In: *FUTUR* (2016), Seite 8. URL: [https://issuu.com/claudiaengel/docs/futur\\_1\\_2016](https://issuu.com/claudiaengel/docs/futur_1_2016) (siehe Seite 23).

- [Fur00] Anthony Furness. „Machine-readable data carriers – a brief introduction to automatic identification and data capture“. In: *Assembly Automation* 20.1 (2000), Seiten 28–34. ISSN: 0144-5154. DOI: 10.1108/01445150010311699 (siehe Seite 13).
- [FWW16] Solmaz Mansour Fallah, Sabine Wolny und Manuel Wimmer. „Towards model-integrated service-oriented manufacturing execution system“. In: *2016 1st International Workshop on Cyber-Physical Production Systems (CPPS)*. IEEE, Apr. 2016, Seiten 1–5. DOI: 10.1109/CPPS.2016.7483917 (siehe Seiten 17, 25).
- [GM16] Sergej N. Grigoriev und Georgi M. Martinov. „An ARM-based Multi-channel CNC Solution for Multi-tasking Turning and Milling Machines“. In: *Procedia CIRP* 46 (2016), Seiten 525–528. ISSN: 22128271. DOI: 10.1016/j.procir.2016.04.036 (siehe Seiten 2, 14, 25, 26, 31).
- [Gro08] Mikell P. Groover. *Automation, production systems, and computer-integrated manufacturing*. Prentice Hall, 2008, Seite 815. ISBN: 0132393212 (siehe Seite 4).
- [HL07] Martin Hardwick und David Loffredo. „STEP-NC : Smart Data for Smart Machining“. In: *Proceedings of the Intl. Conf. on Smart Machining Systems, NIST*. 2007 (siehe Seite 7).
- [Hop14] Stefan Hoppe. „Standardisierte horizontale und vertikale Kommunikation: Status und Ausblick“. In: *Industrie 4.0 in Produktion, Automatisierung und Logistik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, Seiten 325–341. DOI: 10.1007/978-3-658-04682-8\_16 (siehe Seite 5).
- [Lee06] Soo-Yen Lee. „In-process tool condition monitoring systems in CNC turning operations“. Dissertation. Iowa State University, 2006 (siehe Seiten 2, 7, 12).
- [Lee08] Edward A. Lee. „Cyber Physical Systems: Design Challenges“. In: *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)* (2008), Seiten 363–369. DOI: 10.1109/ISORC.2008.25 (siehe Seite 7).
- [Lee99] Y. Tina Lee. „Information Modeling: From Design to Implementation“. In: *Proceedings of the Second World Manufacturing Congress*. 1999, Seiten 315–321 (siehe Seite 5).
- [Lei09] Paulo Leitão. „Agent-based distributed manufacturing control: A state-of-the-art survey“. In: *Engineering Applications of Artificial Intelligence* 22.7 (2009), Seiten 979–991. DOI: 10.1016/j.engappai.2008.09.005 (siehe Seite 17).
- [Lin15] Petra Linke. „Grundlagen zur Automatisierung“. In: *Grundlagen Automatisierung*. Wiesbaden: Springer Fachmedien Wiesbaden, 2015, Seiten 1–28. DOI: 10.1007/978-3-658-05961-3\_1 (siehe Seite 4).



- [Mey10] Heiko Meyer. „Software Architecture of Manufacturing Execution Systems“. In: *Systemics, Cybernetics and Informatics* 8.2 (2010), Seiten 62–66 (siehe Seite 17).
- [MFT09] Heiko Meyer, Franz Fuchs und Klaus Thiel. *Manufacturing Execution Systems - Optimal Design, Planning, and Deployment*. Band 53. 9. 2009. ISBN: 9780071623834 (siehe Seite 4).
- [MVK06] L. Monostori, J. Váncza und S.R.T. Kumara. „Agent-Based Systems for Manufacturing“. In: *CIRP Annals - Manufacturing Technology* 55.2 (2006), Seiten 697–720. DOI: 10.1016/j.cirp.2006.10.004 (siehe Seite 17).
- [Pau+13] Florian Pauker, T Weiler, I Ayatollahi u. a. „Information Architecture for Reconfigurable production systems“. In: *DAAAM International Scientific Book 2013* January (2013), Seiten 873–886. DOI: 10.2507/daaam.scibook.2013.53 (siehe Seiten 3, 17, 18, 24, 26, 27).
- [Pau14] Florian Pauker. „OPC UA for machine tending industrial robots - Prototypic development of an OPC UA server for ABB industrial robots“. In: October (2014). DOI: 10.15224/978-1-63248-031-6-155 (siehe Seiten 5, 24).
- [PFK16] Florian Pauker, Thomas Frühwirth und Burkhard Kittl. „A systematic approach to OPC UA information model design“. In: *49th CIRP Conference on Manufacturing Systems*. May. 2016 (siehe Seite 21).
- [Sch+14] Jan Schlechtendahl, Felix Kretschmer, Armin Lechler u. a. „Communication Mechanisms for Cloud based Machine Controls“. In: *Procedia CIRP*. Band 17. Elsevier, 2014, Seiten 830–834. DOI: 10.1016/j.procir.2014.01.074 (siehe Seite 24).
- [Sch+15] Jan Schlechtendahl, Felix Kretschmer, Zhiqian Sang u. a. „Extended study of network capability for cloud based control systems“. In: *Robotics and Computer-Integrated Manufacturing*. May 2014. 2015. DOI: 10.1016/j.rcim.2015.10.012 (siehe Seiten 8, 24).
- [Sch99] Gerhard Schnell. *Bussysteme in der Automatisierungstechnik: Grundlagen und Systeme der industriellen Kommunikation*. 3. Auflage. Vieweg+Teubner Verlag, 1999. ISBN: 978-3528265694 (siehe Seite 15).
- [Tet+10] R. Teti, K. Jemielniak, G. O'Donnell u. a. „Advanced monitoring of machining operations“. In: *CIRP Annals - Manufacturing Technology* 59.2 (2010), Seiten 717–739. ISSN: 00078506. DOI: 10.1016/j.cirp.2010.05.010 (siehe Seite 14).
- [Vic+15] Axel Vick, Christian Horn, Martin Rudorfer u. a. „Control of robots and machine tools with an extended factory cloud“. In: *2015 IEEE World Conference on Factory Communication Systems (WFCS)*. IEEE, Mai 2015, Seiten 1–4. DOI: 10.1109/WFCS.2015.7160575 (siehe Seite 24).

- [Vij+08] Athulan Vijayaraghavan, Will Sobel, Armando Fox u. a. „Improving Machine Tool Interoperability Using Standardized Interface Protocols : MTConnect“. In: *Laboratory for Manufacturing and Sustainability* (2008) (siehe Seite 6).
- [Wan+04] Lihui Wang, Peter Orban, Andrew Cunningham u. a. „Remote real-time CNC machining for web-based manufacturing“. In: *Robotics and Computer-Integrated Manufacturing* 20.6 (2004), Seiten 563–571. DOI: 10.1016/j.rcim.2004.07.007 (siehe Seiten 1, 16).
- [WVG08] Yunbo Wang, Mehmet C. Vuran und Steve Goddard. „Cyber-physical systems in industrial process control“. In: *ACM SIGBED Review* 5.1 (2008), Seiten 1–2. DOI: 10.1145/1366283.1366295 (siehe Seiten 7, 8, 17).
- [XLY06] X.W. Xu, Lihui Lihui Wang und Yiming Yiming Rong. „STEP-NC and function blocks for interoperable manufacturing“. In: *IEEE Transactions on Automation Science and Engineering* 3.3 (Juli 2006), Seiten 297–308. DOI: 10.1109/TASE.2005.862147 (siehe Seite 4).
- [XN06] X.W. Xu und S.T. Newman. „Making CNC machine tools more open, interoperable and intelligent - a review of the technologies“. In: *Computers in Industry* 57.2 (2006), Seiten 141–152. DOI: 10.1016/j.compind.2005.06.002 (siehe Seite 7).
- [Yon04] Chi Yonglin. „An evaluation space for open architecture controllers“. In: *The International Journal of Advanced Manufacturing Technology* 26.4 (2004), Seiten 351–358. DOI: 10.1007/S00170-004-2111-X (siehe Seite 16).