

Inhalt

- 1 Einführung ✓
 - 1.1 Problemstellung
 - 1.2 Zielsetzung
 - 1.3 Lösungsansatz
- 2 Anwendungsszenario ✓
 - 2.1 Das Kooperatives-Überholen-Szenario
 - 2.2 Die Roboterauto-Zielplattform
- 3 Grundlagen (~~fertig~~) ✓
 - 3.1 Modellbasierte Codegeneration
 - 3.2 Modelltransformationen
 - 3.3 Die MechatronicUML und ihre Tool-Suite
 - 3.4 MechatronicUML plattformunabhängiges Modellieren
 - 3.5 MechatronicUML Hardware-Plattform-Beschreibungs-Modellieren.
 - 3.6 Continuous Integration/Continuous Deployment
 - 3.7 Eclipse-Plug-Ins
- 4 Verwandte Arbeiten ✓
 - 4.1 Taxonomien
 - 4.2 MechatronicUML
 - 4.3 Continuous Integration/Continuous Deployment
 - 4.4 Ansätze für die Verwendung von Arduinocode innerhalb der Eclipse-IDE
 - 4.5 Ansätze für die Automatisierung von Abläufen mit oder innerhalb der Eclipse-IDE
- 5 Entwurf der CI/CD-Pipeline ✓
- 6 Umsetzung ✓
 - 6.1 Volle Unterstützung von neuer oder geänderter Hardware, Software und Bibliotheken ✓
 - 6.2 Integration der Arduino-CLI ✓
 - 6.3 Vollständige Automatisierung per Pipeline ✓
- 7 Evaluation ✓
 - 7.1 Studienaufbau (✓) ✓
 - 7.2 Qualitätsbewertungen ✓
 - 7.3 Ergebnisse ☐ ✓
 - 7.4 Diskussion ☐ ✓
 - 7.5 Gefahren für die Gültigkeit ☐ ✓
- 8 Schlussfolgerung ☐

Bibliografie

Ergänzungsmaterial Übersicht der Pipelineschritte

Ergänzungsmaterial Änderungen an MUML-Modellen

Ergänzungsmaterial Änderungen an Post-Processing-Ablauf

[(((Ergänzungsmaterial Mikrocontroller)))]

Ergänzungsmaterial für die Integration der ArduinoCLI

Ergänzungsmaterial für Eclipse-Plig-In-Projekt „MUMLACGPPA“

Ergänzungsmaterial für Eclipse-Plig-In-Projekt „PipelineExecution“

[(((Ergänzungsmaterial für das Modellieren des Anwendungsszenarios)))]

Ergänzungsmaterial für die Nutzung der Pipeline

[(((Ergänzungsmaterial für die Softwareerstellung)))]

Struktur der Ausarbeitung:

Diese Ausarbeitung ist wie folgt aufgebaut:

Kapitel 2 - Anwendungsszenario: Zuerst wird das in dieser Ausarbeitung verwendete Anwendungsszenario beschrieben. Dieses beschreibt eine Beispielanwendung, die verwendeten Roboterautos und die Versuchsumgebung.

Kapitel 3 - Grundlagen: Als zweites werden die in dieser Ausarbeitung verwendeten theoretischen Grundlagen erklärt. Dies betrifft die modellbasierte Codegeneration, Modelltransformationen, MechatronicUML und ihre Tool-Suite, MechatronicUMLs plattformunabhängiges Modellieren, MechatronicUMLs Hardware-Plattform-Beschreibungs-Modellierung und die DevOps-Methode Continuous Integration/Continuous Deployment.

Kapitel 4 - Verwandte Arbeiten: Als drittes werden bereits bestehende Ausarbeitungen und Ansätze, die für diese Ausarbeitung relevant sind, erörtert. Dies betrifft Ausarbeitungen und Artikel zu den Themen der Taxonomie, MechatronicUML und Continuous Integration/Continuous Deployment. Ansätze für die Verwendung von Arduinocode innerhalb der Eclipse-IDE und für die Automatisierung von Abläufen mit oder innerhalb der Eclipse-IDE werden auch abgehandelt werden.

Kapitel 5 - Entwurf der CI/CD-Pipeline: In diesem Kapitel werden die Aspekte und Entscheidungen bei dem Entwurf der CI/CD-Pipeline beschrieben.

Kapitel 6 - Umsetzung: Danach werden die Details der verschiedenen Umsetzungen beschrieben. Hierbei wird also erklärt, wie die volle Unterstützung von neuer oder geänderter Hardware, Software und Bibliotheken sowie die Integration der Arduino-CLI erreicht wurden. Dasselbe gilt für die Automatisierung der Nachbearbeitung der generierten Code-Teile und die vollständige Automatisierung per Pipeline.

Kapitel 7 - Evaluation: Die jeweilige Qualität der umgesetzten Änderungen, der Integration der ArduinoCLI und der Automatisierung in Form der CI/CD-Pipeline wird in diesem Kapitel erörtert und bewertet. Hierfür werden die Ergebnisse der Umsetzung des Anwendungsszenarios ausgewertet und diskutiert.

Kapitel 8 - Schlussfolgerung: Als Abschluss erfolgt eine kurze Zusammenfassung der wichtigsten Errungenschaften und Begrenzungen [(((/Einschränkungen sowie erhaltenen Erkenntnissen)))]. Es werden auch Punkte und Themen für zukünftige Forschungen angeschnitten.