SYT7 4xHIT 23/24 / Industrielle Programmierung / Indlnf 7.1 Roboterprogrammierung [GK] – 4h

# 🕞 IndInf 7.1 Roboterprogrammierung [GK] – 4h

✓ Done: View ✓ Done: Make a submission ✓ Done: Receive a grade

# Roboterprogrammierung

Roboter sind IT-Systeme, die mit der Weit physikalisch interagieren. Die Programmierung unterscheidet sich in vielen Punkten von der klassischen "Applikationsentwicklung", da hier Problemfelder wie Sicherheit (im Sinne körperlicher Sicherheit, Verletzungsvermeidung, ...) und verlässlicher Betrieb in Zusammenarbeit mit vielen verschiedenen Geräten und Komp ten in Spiel kommen.

Diese Übung soll einen kleinen Einblick in diese spannende Welt geben, um "Awareness" für eine mögliche zukünftige Zusammenarbeit mit Mechatronikern und Automatisierung zu schaffen - und um einmal mit den coolen Spielzeugen hantieren zu dürfen 😩

#### Voraussetzungen

Im 3. Jahrgang hast du schon Kontakt mit einem Industrieroboter gehabt. Dabei sollten folgende Ziele erreicht worden sein:

- Du hast die Sicherheitseinweisung am spezifischen Roboter mitgemacht.
  Du kannst den Roboter grundlegend in Betrieb nehmen und einfache Meldungen des Systems bearbeiten.
  Du kannst alle Achsen des Roboters benennen und per Handbedlengerät bewegen.
- · Du kennst die Bewegungseinschränkungen und Limits aller Achsen und kannst den TCP gezielt an einen Punkt im Raum steuern

### Ziele dieser Übung

- Du kennst die verschiedenen Koordinatensysteme, die bei Roboterbewegungen eingesetzt werden, und die Gründe dafür.
   Du kannst eigene Werkzeug- und Basis-Koordinatensysteme definieren und Roboterbewegungen sinnvoll einsetzen.
- Du kannst ein einfaches Programm zum Abfahren eines Pfades schreiben, ausführen und testen.

#### Abgabe

In dieser Übung soll vor allem praktisch mit dem Roboter gearbeitet werden. Notiere dabei aber dennoch, wie du deine Aufgaben umgesetzt hast, um dies später (im 2. Semester) ohne nochmalige Erklärungen umsetzen zu könner. Seine Arbeit informativ, übersichtlich und effizient protokollieren zu können, ist ein wesentlicher Skill eines Ingenieurs, der auch in dieser Übung geübt werden soll.

#### Abzugeben sind:

- kommentierter Quelitext aller erstellten Module (Sourcecode + Variablendatei) pro Gruppe
- ein Video der Programmausführung wie weit ihr gekommen seid (alle Beteiligten müssen darauf zumindest kurz zu sehen sein) pro Gruppe
   PDF-Protokoil, das die Notizen der eigenen Tätigkeit und eine Kurzreferenz der verwendeten KRL-Befehle und der dafür benötigten Parame

Dateinamen sollen immer alle beteiligten Namen enthalten; z.B. Video; IndInf\_7.1\_Müller-schneider-Sams.mp4, Protokoll; IndInf\_7.1\_Müller.pdf

#### Aufgabenstellung

#### Teil 1 (GK)

- 1. Befestige die Schablone "Layout 4" (Format A3, Farbe) im Arbeitst
- 2. Setze 3 Würfel auf die Plätzen 1, 2 und 3. Anschließend setze mithilfe des Roboters die Würfel folgendermaßen um:

Schüler 1: 1-->4, 2-->4 (Würfel 2 horizonal greifen)

Schüler 2: 2-->5, 3-->5 (Würfel 3 horizonal greifen)

Schüler 3: 3-->4, 2-->5 (Würfel 2 horizonal greifen) Schüler 4: 1-->5, 3-->5 (Würfel 2 horizonal greifen)

#### Beim horizontalen Greifen ACHTUNG auf den Abstand zwischen Roboterarm und Tisch!

- e die Konfiguration unter Basis Nummer 10. Verwende dazu das bereits vermessene Werkzeug "LISE0 Greifer" (Nummer 2)
- 3. Vermiss die durch das blaue Koordinatenkreuz gegebene Basis und speichere die Konfiguration unter Basis Nummer 10. Verwende dazu das bereits ve 4. Wiederhole die Bewegungen aus 2. Welche Verbesserungen ergeben sich durch Benutzung des Basiskoordinatensystems beim Umsetzen der Würfel?

#### Teil 2 (GK)

Nun sollen die Bewegungen in automatisch ablaufende Programme umgesetzt werden. Damit nach jedem Schülerteil die Würfel wieder in die Ausgangslage gelegt werden können, soll jeweils eine 3 Sekunden Pause programmiert werden

- 1. Erstelle ein neues Modul mit den Name aller Schüler (ev. abgekürzt)
- 2. Programmiere die Bewegungsabläufe: "Facets" die Verdrecht die seibe Koordinate in mehrere Variablen speichern). Benenne alle Punkte nachvollziehbar!

  Achte darauf, den Greifer nicht schon während der Bewegung zu öffnen bzw. zu schließen, sondern erst bei Süllstand des Roboters.

  3. Verwende für Wege zwischen Würfelpositionen Point-to-Point-Bewegungen, für das Greifen der Würfel Linearbewegungen.
- 4. Teste das Programm ausführlich.

## Teil 3 (EK)

Die Bewegungen des Roboters können durch "Überschleifen" optimiert werden. Dabei fährt ein Roboter einen "überschliffenen" Punkt nicht mehr genau an und bleibt dort stehen, sondern setzt seine Bewegung unmittelbar zum nächsten Punkt fort. Der "überschliffene" Punkt kann dabei nur estreift" werden. Schau dir dazu auch das entsprechende Kapitel in der Dokumentation an

- 1. Baue "Überschleifen" sinnvoll in das Programm ein. **Teste sehr langsam**, da sich hier teils unvorhersehbare Bewegungen ergeben können!

  2. Durch Nutzung des Basiskoordinatensystems und Wissen um die Abstände der Würfelpositionen lassen sich viele Roboterpositiohen auch im Programm errechnen und müssen so nicht geteacht werden. Informiere dich, wie Punktvariablen gezielt im Sourcecode manipuliert werder (z.B. 50mm zur X-koordinate eines Punktes hinzufügen) und optimiere das Programm damit.

  Deklariere und verwende wo sinnvoll zusätzliche Variablen (z.B. Abstand der Position des Greifers über einem Würfel zur tatsächlichen Greifposition könnte für alle Würfel gleich sein -> Wert in einer Variable geziehtm)

Achtung: für 2. benötigt ihr mindestens "Experten"-Rechte, damit ihr manuell Code ohne Formulare ergänzen könnt.

### Dokumentationen und Ressourcen

25 September 2023, 11:01 PM 25 September 2023, 11:01 PM - KRL Cheatsheet.pdf 25 September 2023, 11:01 PM KUKA KRL-Quickguide.pdf
SYT-Lab Layout 4.pdf 25 September 2023, 11:01 PM 25 September 2023, 11:01 PM

Edit submission Remove submission

#### Submission status



## Feedback

Grade	GK vollståndig
Graded on	Tuesday, 16 January 2024, 3:00 PM
Graded by	List Erhard