



Informatikwerkstatt-MR Einführung

G. Kemnitz

Institut für Informatik, TU Clausthal (IW_Einf.pdf)
20. Oktober 2022



Wer sind wir

Arbeitsbereich: Elektronik und Digitaltechnik

- Dozent: Prof. Günter Kemnitz
- wiss. Mitarbeiter: Dr. Carsten Giesemann
- Tutor: Gustav Baier



Informatikwerkstatt



Lernziele, Inhalt, Bewertung

Lernziele:

- Kennenlernen typische Arbeitsfelder und berufsbezogene Tätigkeiten,
- Grundfertigkeiten für die Software-Entwicklung:
 - Analyse von Problemstellungen,
 - Implementierung, Test, Fehlersuche,
 - Team-Arbeit, Präsentation der Lösungen, ...

Inhalt:

- Einführung in ein bis zwei Programmiersprachen und Programmierumgebungen,
- interaktive Rechnerübungen und Mini-Projekte,
- Abschlussprojekt und Projektpräsentation.

Bewertung:

- unbenoteter Leistungsnachweis

Erforderliche Vorkenntnisse

- Keine. Die Einführung in die Programmierung startet von null. Aber auch bei Vorkenntnissen nicht langweilig.



Informatikwerkstatt-MR

Lern- und Arbeitsziel

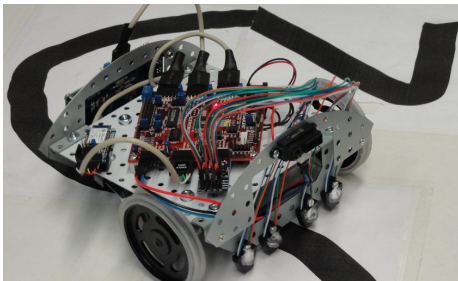


- Schrittweise Heranführung an die Entwicklung / Inbetriebnahme / Fehlersuche programmierter MR.
- Konfiguration eines Fahrzeugs mit einer Mikrorechnerbaugruppe und Ansteck- (PMod-) Modulen¹.

¹Benachbarte Arbeitsplätze bilden eine 2er- bzw. 3er-Gruppe mit je einem Satz Hardware.

Abschlussprojekt

Demonstrator für ein autonomes Fahrzeug, programmiert nach einer selbst gewählten mit dem Betreuer abgesprochenen Spezifikation oder ein anderes MR-Projekt.



Umfang:

- Bei Start ohne Vorkenntnisse ca. eine A4-Seite funktionierendes eigenes Programm plus 10 min. Vortrag.
- Wir hatten aber auch schon recht anspruchsvolle Projekte.



Zeitablauf, Leistungsnachweis

- Präsenzzeit Mo. 15:00 bis 19:00 Uhr und Di. 17:15 bis 19:00 Uhr.
- erste Semesterhälfte: Mo. 15:00 bis 17:00 und optional Di. 17:15 bis 18:00 Vorlesung.
- Rest Präsenzzeit bis 19:00 Übung.
- ca. ab Mitte Dezember bis vorletzte VL-Woche nur Übung
- letzte Vorlesungswoche: Präsentation der Abschlussprojekte.

Zu erbringende Leistungen:

- Vorführung gelöster Aufgaben. Schwierigkeit nach Vorkenntnissen in Absprache mit Betreuer.
- Abschlusspräsentation.
- Arbeitsumfang incl. Vorlesungsstunden 6 LP².

Bei Abwesenheit sind die zu erbringenden Leistungen nachzuholen

²1 LP = 30 h, 6 LP = 180 h, 14 Vorlesungswochen, ca. 13 h Arbeitsstunden, 6 h Präsenzzeit + 7 h Eigenstudium pro Vorlesungswoche. Das sind $\approx 20\%$ der Arbeitszeit in ihrem ersten Semester.



Arbeitsprogramm

- Kennenlernen der Entwicklungsumgebung (Atmel-Studio, ...).
- Bitverarbeitung: Einfache Programme mit Eingabe über Schalter und Ausgabe an LEDs, Automaten, ...
- C-Programmierung, Datentypen, Modularisierung, Modultest, ...
- PC als Ein- und Ausgabe. Logikanalyse. Python für Programmtest vom PC aus. ...
- Ansteuerung weiterer Hardware-Einheiten (Ultraschallsensor, LC-Display, Timer, ...).
- Nebenläufigkeit: Treiber, Polling, Interrupts, Überwachung von Zeitabläufen.
- Motorsteuerung: Kennlinienbestimmung, Regelung, ...

Zu jedem Thema werden zu Beginn funktionierende Programmbeispiele besprochen und gemeinsam getestet.



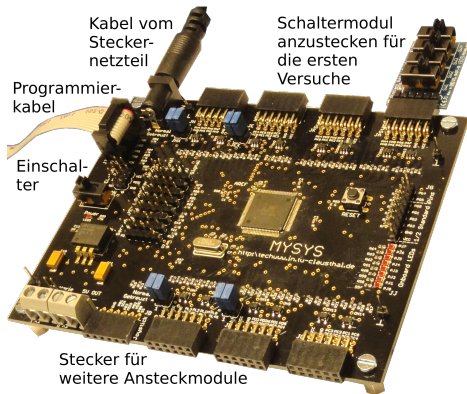
HW und SW-Tools

Inbetriebnahme der Baugruppe

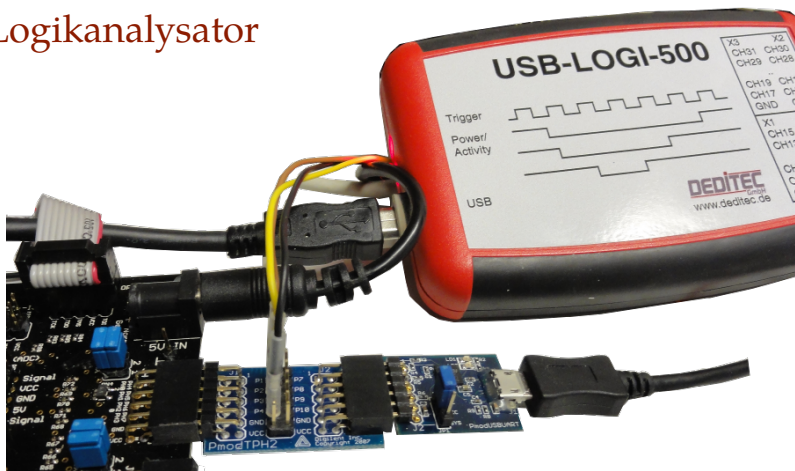
- Programmieradapter anstecken.
- Netzteil anstecken (Achtung, nur 5 V-Netzteile verwenden).
- Schaltermodul JA (Port A) anstecken.
- Verbindung zum PC herstellen, ...

Erweiterungsmodule:

- weitere Eingabe- und Ausgabemodule: Joystick, LC-Display, ...
- Motoren und Motoransteuermodule,
- Sensoren: IR-Boden, IR-Abstand, Ultraschall-Abstand, 3-Achsen Beschleunigung, GPS (nur im Freien)



Logikanalysator

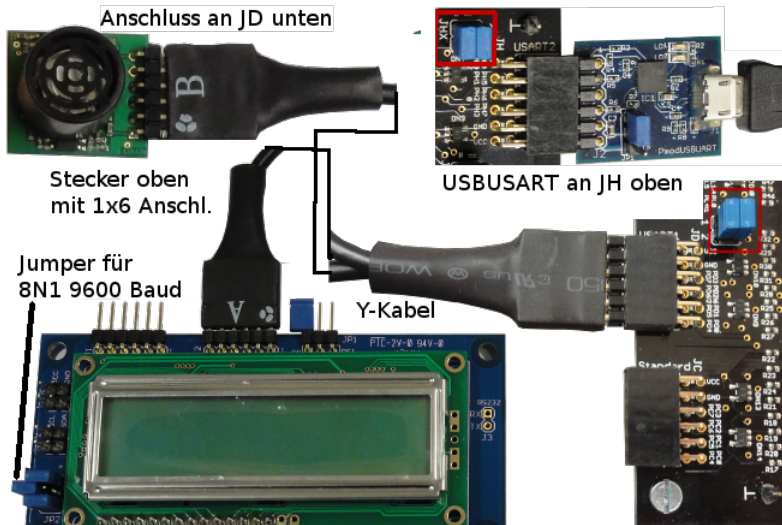


PModUSBUSART über Zwischenadapter PmodTPH2, daran

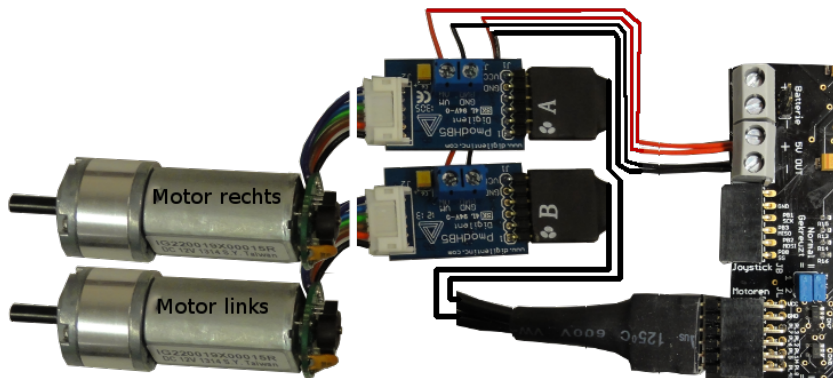
- GND (schwarz) an Masse (Adapter Gnd),
- CH0 (gelb) an RxD (Adapter P3) und
- CH1 (braun) an TxD (Adapter P2) anstecken.



Versuchsaufbau: Echolot

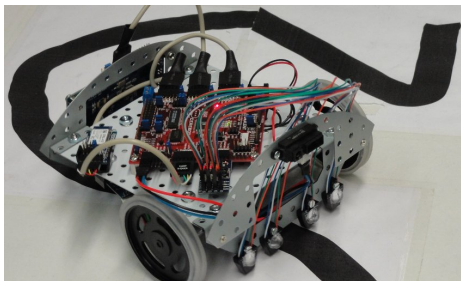


Versuchsaufbau Motorenansteuerung



- 2×H-Brücke (Motortreiber) PmodHB5 über Y-Kabel an JL,
- Motoren an die H-Brücken stecken.
- Ausgabe Geschwindigkeitswerte und Richtungsbits,
- Rücklesen: Hallsensorwerte für Drehwinkelschritte.

Beispiel für ein Abschlussprojekt



Linienverfolgung:

- Wenn zu weit links, linken Motor beschleunigen.
- Wenn zu weit rechts, rechten Motor beschleunigen, ...

Dazu Treiber, Kommunikation, ... ca. eine A4-Seite Programmtext.

Weitere Vorschläge: automatisches Einparken, geplante Bahnen abfahren, Labyrinthausgang suchen, ...



Mikrorechner oder Anwendungssysteme?

Beide Veranstaltungen liefern eine praktische Einführung in die Programmierung.

- Anwendungssysteme: Python und zahlreiche Bibliotheken
- Mikrorechner: C und Python-Basics (nur was für Test- und Steuer-Skripte auf dem PC benötigt wird).

Beide Veranstaltung geben einen praktischen Einblick in das gesamte Tätigkeitsfeld der Informatik von der Ideenentwicklung über Programmierung, Debuggen bis zur Ergebnispräsentation.

Besonderheiten Angebot MR:

- Programmierung nahe an der Hardware.
- Hilfreich für Verständnis: Funktionsweise Rechner, interne Programmabarbeitung, Sensoren, Kommunikation, ...
- Hilfreich für hardware-nahes Debuggen: Gerätetreiber, Maschinensteuerungen, Fahrzeugsteuergeräte, ...