



UNIVERSITÄT ULM

FAKULTÄT FÜR INFORMATIK

Abteilung Rechnerstrukturen

Echtzeitsysteme I

Prof. Dr. J. Kaiser / Carlos Mitidieri

Übungsblatt 1

Analog–Digital–Wandler:

Wie groß ist der maximale Quantisierungsfehler bei einem 10–Bit AD–Wandler, wenn die Eingangsspannung zwischen 0V und 10V liegt?

Ein Hitachi H8/300 Micro–Controller soll zur A/D–Wandlung eingesetzt werden mit den folgenden Eigenschaften:

1. Es soll eine kontinuierliche Wandlung der Eingänge A_0 bis A_3 durchgeführt werden.
2. Der Interrupt soll abgeschaltet sein.
3. Die Wandlungszeit soll so kurz wie möglich sein.

Geben Sie die Konfiguration des Kontrollregisters (ADCSR) an. Die notwendigen technischen Informationen finden Sie in der Dokumentation des H8/300 unter:

<http://www.informatik.uni-ulm.de/rs/projekte/core/rts1/technical.docs/Hitachi.H8.300/h83297h.pdf>

Zeitgeber und Zähler:

Die folgenden Aufgaben beziehen sich auf die 8–Bit Zeitgeber/Zähler–Einheit des Hitachi H8/300 Micro–Controllers. Die notwendige technische Dokumentation finden Sie unter:

<http://www.informatik.uni-ulm.de/rs/projekte/core/rts1/technical.docs/Hitachi.H8.300/h83297h.pdf>

Beispielprogramme, welche die Form der Lösung beschreiben, sind zu finden unter:

<http://www.informatik.uni-ulm.de/rs/projekte/core/rts1/technical.docs/Hitachi.H8.300/h8330a.pdf>

Aufgaben:

1. Spezifizieren Sie eine kontinuierliche Messung des Tastverhältnisses eines periodischen Rechtecksignals.
2. Generieren Sie eine feste Frequenz mit 50% Tastverhältnis.

Programmieraufgabe 1:

Machen Sie sich mit Lego MindStorms –Bauelementen vertraut. Bauen Sie ein Fahrzeug und statten Sie es mit einem (oder mit zwei) Bumper(s) aus. Bringen Sie das Fahrzeug zum fahren, indem Sie dafür ein NQC–Programm [] schreiben. Das Fahrzeug soll erkennen, wenn es gegen eine Wand o. ä. gefahren ist und entsprechend darauf reagieren (ausweichen).

Programmieraufgabe 2:

Ein Wettbewerb!

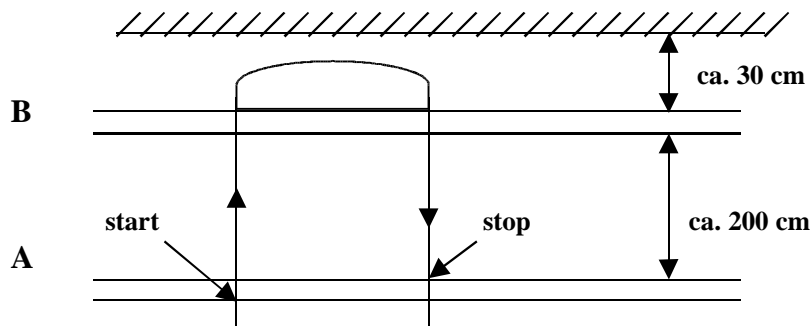
Statten Sie das Fahrzeug aus Aufgabe 1 (oder bauen Sie ein anderes) mit dem Lichtsensor aus.

Das Ziel ist es, in möglichst kurzer Zeit die Strecke zwischen der weißen Linie **A** und der weißen Linie **B** zu durchfahren, Linie **B** zu überqueren und wieder zur Linie **A** zurückzufahren (siehe Skizze). Die Zeit läuft ab dem Moment, ab dem das Fahrzeug zum ersten Mal die Linie A erreicht hat, bis zu dem Moment, zu dem es diese Linie zum zweiten Mal erreicht hat. Schreiben Sie ein NQC–Programm, welches das realisiert. Es soll anschließend die Zeit, die Ihr Fahrzeug dafür gebraucht hat, auf dem LCD ausgeben.

Die Zeit hängt natürlich nicht nur vom Algorithmus, sondern auch von der Konstruktion des Fahrzeuges ab.

Die Gruppe, deren Fahrzeug das schnellste war, hat gewonnen!

Vorführung am 15.11.01 (1. Preis: eine wertvolle Urkunde ++)



Hinweise:

- Der Not Quite C–Compiler heißt **nqc**, Programme werden also mittels **nqc prog.nqc** compiliert. Falls Sie RCX2–Befehle verwenden, müssen Sie beim Compilieren (und beim Downloaden) die Option **-Trcx2** angeben: **nqc -Trcx2 prog.nqc**.
- Der Download in den RCX geschieht mittels **nqc -d prog.nqc** bzw. **nqc -d -Trcx2 prog.nqc**, dazu muß der RCX eingeschaltet sein.
- Der RCX hat fünf Programmspeicherplätze, wenn nicht der erste verwendet werden soll, so ist beim Download der Programmspeicherplatz anzugeben, etwa **nqc -pgm 3 -d prog.nqc**.

Nützliche Links:

LEGO MindStorms allgemein:

<http://mindstorms.lego.com/>

<http://home.nexgo.de/mindrobots/de/MindStorms/Programmierung/moin.htm>

RCX:

<http://www.cs.brown.edu/courses/cs148/lectures/programming.htm>

<http://www.graphics.stanford.edu/~kekoa/rcx/>

<http://www.plazaearth.com/usr/gasper/lego.htm>

NQC:

<http://www.enteract.com/~dbaum/nqc/>

legOS:

<http://www.smallrobots.de/ws2000/dackel/>

<http://ki.informatik.uni-wuerzburg.de/~lego/praktikum/>

<http://www.noga.de/legOS/>

H8:

http://semiconductor.hitachi.com/H8/h8_family.html

und lokal unter:

<http://www.informatik.uni->

[ulm.de/rs/projekte/core/rts1/technical.docs/Hitachi.H8.300](http://www.informatik.uni-ulm.de/rs/projekte/core/rts1/technical.docs/Hitachi.H8.300)