

VORLÄUFIGE ARBEITSKOPIE!

UEBERSETZEN VON
SCHRITTMOTORPROTOKOLLEN

Entwurf eines Hardwareübersetzers

Praxisbericht

im Fachgebiet Mess- und Sensortechnik



vorgelegt von: Johannes Dielmann
Studienbereich: Technik
Matrikelnummer: 515956
Erstgutachter: Prof. Dr. Carstens-Behrens

© 2012

Dieses Werk einschließlich seiner Teile ist **urheberrechtlich geschützt**. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Autors unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen sowie die Einspeicherung und Verarbeitung in elektronischen Systemen.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation	1
1.2. Ziel der Arbeit	1
1.3. Aufbau der Arbeit	2
2. Begriffe und Komponenten	3
2.1. Koponenen zur Erfassung	4
2.1.1. Lasererfassungssystem VI-900	4
2.1.2. RapidForm2004	4
2.2. Drehtisch	4
2.2.1. Schrittmotoren	4
2.2.2. Schrittmotorkarten	4
2.3. Mikrocontroller	5
2.3.1. Entwicklungsumgebung	5
2.3.2. Entwicklerboard STK500	5
2.3.3. AVRISP mkII	5
3. Hardware	6
3.1. Mikrocontroller	6
3.1.1. MAX232	6
3.2. Schrittmotorkarten	6
3.2.1. Endschalter	6
3.2.2. Motorverkabelung	6
3.3. Spannungsversorgung	6
3.4. Platinenlayout	6
4. Entwicklung der Software	7
4.1. Entwicklungs Umgebung	7
4.2. RapidForm2004	7
4.3. Watchdog	7

4.4. Einfaches Beispiel für LEDs und Taster	8
4.5. LCD Bibliothek	8
4.6. RS-232	8
4.7. Menü Bibliothek	8
4.8. Interrupts	8
4.8.1. Endschalte	8
4.8.2. Watchdog	8
4.9. Protokoll der Stepperkarte	8
4.10. Manueller Betrieb	8
4.11. Protokolle aus RapidForm	8
4.12. Übersetungs Logik	8
4.13. Automatische Protokollwahl	8
5. Fazit und Zukunft	9
5.1. Fazit	9
Eidesstattliche Erklärung	10
A. Anhang	i
A.1. Schritt für Schritt Anleitung	ii
A.2. Vom Autor verwendete Software	iii

Abbildungsverzeichnis

2.1. Überblick des Arbeitsplatz	3
---	---

Tabellenverzeichnis

Codeverzeichnis

4.1. Funktion - Laufflicht	7
--------------------------------------	---

1. Einleitung

(TODO: KLARSTELLEN DER BEGRIFFLICHKEITEN) In der **(TODO: CAD ERKLÄREN)** *CAD-Entwicklung* kommt es vor das für ein real existierendes Objekt eine Erweiterung konstruiert werden muss. Um die Erweiterung sinnvoll konstruieren zu können müssen dazu die Abmessungen des Objektes möglichst genau bekannt sein. Das übertragen der Abmessungen, kann insbesondere für komplexe Objekte, sehr aufwendig sein. Abhilfe soll ein Laserscanner schaffen der das Objekt aus mehreren Richtungen vermisst und aus diesen Informationen ein genaues 3D-Modell davon generiert.

1.1. Motivation

Mit dem Aufbau aus RapidForm2004, Lasererfassungssystem VI-900 und Drehtisch sollen auf einfachem Wege 3D-Modelle eines Objektes erzeugt werden. Diese sollen anschließend in einer CAD-Software wie *Solidworks* nutzbar sein.

1.2. Ziel der Arbeit

Die Kommunikation zwischen der Software RapidForm2004 und dem gegebenen Drehtisch soll ermöglicht werden. Dazu werden die ASCII-Befehle der Software mit einem Mikrocontroller ausgewertet und in für den gegebenen Schrittmotor verständliche Befehle übersetzt. Es wird also ein Mikrocontroller mit 2 RS-232 Schnittstellen so programmiert das er die Befehle der Software übersetzen kann. Um den den Drehtisch manuell bedienen zu können und den aktuellen Status zu überprüfen sind noch ein LC-Display und mehrere Bedientaster vorgesehen.

Die Ansteuerung des Schrittmotors ist als Einschub für ein 19Rack realisiert. Daher wird die Platine für den Mikrocontroller auch als 19Einschub realisiert.

1.3. Aufbau der Arbeit

Der Aufbau der Arbeit gliedert sich im Wesentlichen in die Entwicklung neuer und die Nutzung vorhandener Hardware, sowie in die Entwicklung der Software.

Zur Hardware gehören die Auswahl des Mikrocontroller, die Endschalter, die Schrittmotorkarten, die Schrittmotoren, sowie die verwendeten PCs. **(TODO: KABEL ZU DEN MOTOREN NICHT VERGESSEN! SCHEMA ZEICHNUNG?)** Zur Software gehören die Entwicklungsumgebungen und die 3D-Erfassungssoftware.

(TODO: AUSBAUEN! ÜBERBLICK VERSCHAFFEN. KOMMUNIKATION MIT SCHRITTMOTOR-KARTE VON PC AUS. AUFBAUEN DES STK500. EINARBEITEN IN UC ENTWICKLUNG. STEuern DER SCHRITTMOTOR KARTE VOM UC AUS. ERARBEITEN DER PROTOKOLLE (REVERSE ENGINEERING). C-PROGRAMM ZUM VERSTEHEN EINGEHENDER BEFEHLE. ÜBERSETZEN DER BEFEHLE. NEUER MIKROCONTROLLER. UMGEBUNGS WECHSEL. LC-DISPLAY. ENDSCHALTER. MAX232. PLATINENLAYOUT.)

2. Begriffe und Komponenten

(TODO: BESSERES BILD! SCHEMA? BESSERE BESCHRIFTUNG!) (TODO:



Abbildung 2.1.: Überblick des Arbeitsplatz

BESSERE ÜBERSCHRIFT!) (TODO: FUSSNOTEN FÜR KOMPONENTEN,
HERSTELLER UND WEBSITES) (TODO: KOMPONENTEN AUF ABBILDUNG
ERWÄHNEN!)

2.1. Komponenten zur Erfassung

2.1.1. Lasererfassungssystem VI-900

Das Lasererfassungssystem VI-900 der Firma Minolta besteht aus einem Lasertriangulator und einer Kamera. Das System lässt sich über eine SCSI Schnittstelle ansprechen und konfigurieren. Zur mobilen Nutzung kann das Gerät auch auf der Rückseite eingestellt werden. Aufgenommene Daten können auf einer CF-Karte gespeichert werden. Im Projekt wurde jedoch lediglich die Ansteuerung via SCSI genutzt.

2.1.2. RapidForm2004

2.2. Drehtisch

Der Drehtisch ist eine Eigenkonstruktion der Werkstatt des RheinAhrCampus. Er besteht aus einer massiven Edelstahl Arbeitsplatte, welche auf 4 Füßen ruht. Aus dieser ist eine **(TODO: WELCHE FORM??)** ausgeschnitten. In diesem Ausschnitt befindet sich, auf einem Zweischienensystem gelagert, der Drehtisch. Mit dem Schienensystem lässt der Drehtisch sich in der Vertikalen positionieren. Mit einem Schrittmotor lässt sich der Drehtisch in der Höhe verstellen. Ein weiterer Schrittmotor ist für die Drehung des Tisches zuständig. **(TODO: GETRIEBE ERKLÄREN!)**

2.2.1. Schrittmotoren

2.2.2. Schrittmotorkarten

Die Ansteuerung für den Drehtisch besteht aus einem 19Rack. In diesem ist ein ATX-PC-Netzteil verbaut. Außerdem sind 2 Einschubkarten der Firma R+S vorhanden. Die Karten sind sogenannte Stepper-Karten. Diese übernehmen komfortabel die Ansteuerung der beiden Schrittmotoren. Mittels RS-232 Schnittstelle lassen sich die Karten konfigurieren und ansteuern. Die Konfiguration und Ansteuerung erfolgt über einen vorgegeben ASCII Befehlssatz. Außerdem können 2 oder mehr Karten als "Daisy-Chain" zusammengeschaltet werden. **(TODO: DAISY-CHAIN ERKLÄREN UND KONFIGURATION GENAUER BESCHREIBEN.)**

2.3. Mikrocontroller

2.3.1. Entwicklungsumgebung

2.3.2. Entwicklerboard STK500

Um den eingesetzten Mikrocontroller zu programmieren und die Programmierung zu überprüfen wurde mir das Entwicklerboard STK500 der Firma ATMEL zur Verfügung gestellt. Das Board enthält mehrere Mikrocontroller Steckplätze, 2 Serielle Schnittstellen, 8 Taster, 8 LEDs, 2 Erweiterungsports, ein integriertes Programmiersystem (**TODO: BESSERER NAME!**) und mehrere Jumper zum konfigurieren des Boards.

Von den beiden seriellen Schnittstellen kann die eine zur Programmierung des Mikrocontroller verwendet werden. Die andere kann zur Kommunikation mit dem Mikrocontroller genutzt werden.

Auf dem Board stehen 5 10 polige Stiftleisten¹ zur Verfügung. Diese sind direkt mit dem Mikrocontroller verbunden und können über Flachbandkabel an Peripherie wie z.B. Taster, LED und LC-Displays angeschlossen werden.

2.3.3. AVRISP mkII

¹Eine Stiftleiste (engl. pin header) ist ein Steckverbinder mit mehreren in Reihe angeordneten Stiftkontakten, der auf Leiterplatten in der Elektronik Verwendung findet. Sie hat den Zweck, eine Verbindung mit vielen Kontakten von einer Platine zu einer anderen oder zu peripheren Baugruppen herzustellen, meist mit Hilfe von Flachbandkabeln und Pfostenverbindern oder einer Buchsenleiste. ?

3. Hardware

3.1. Mikrocontroller

Zu Beginn stand mir ein ATmega 8515? im DIL-Gehäuse zur Verfügung. Dieser hatte 8Kbyte Flash, 3 externe Interrupts, 1 Serielle Schnittstelle und konnte mit bis zu 16 MHz betrieben werden. Mit diesem konnte ich mich in die Programmierung mit C einfinden und eine Serielle Schnittstelle ansteuern. Für mein Projekt sind jedoch 2 externe Schnittstellen nötig. Nach Recherche entschloss ich mich für einen ATmega 644 PA. (**TODO: VERLINKEN**) Dieser ist dem ATmega 8515 recht ähnlich, bietet jedoch die benötigten 2 seriellen Schnittstellen. Des weiteren hat er 32Kbyte Flash und verfügt über 32 externe Interrupts. (**TODO: MEHR SCHREIBEN??**)

3.1.1. MAX232

3.2. Schrittmotorkarten

3.2.1. Endschalter

(**TODO: VERKABELUNG, NEUER AUSLÖSER, ENTWICKLUNG DER VERDRAHTUNG, BERÜCKSICHTIGUNG IM UC**)

3.2.2. Motorverkabelung

3.3. Spannungsversorgung

(**TODO: VERKABELUNG STECKBAR UND UNIVERSELL GEMACHT**)

3.4. Platinenlayout

4. Entwicklung der Software

(TODO: EINFÜHRUNG SCHREIBEN) (TODO: WEITERE SOFTWARE IN BEGRIFFEN ERKLÄREN. MINOLTA)

4.1. Entwicklungs Umgebung

(TODO: AVR STUDIO ECLIPSE BUG DEFECTS BIBLIO?)

Listing 4.1: Funktion - Lauflicht

```
1
2 void led_laufflicht (void) {
3     uint8_t i = LED_PORT;
4     i = (i & 0x07) | ((i << 1) & 0xF0);
5     if (i < 0xF0)
6         i |= 0x08;
7     LED_PORT = i;
8 }
```

4.2. RapidForm2004

4.3. Watchdog

(TODO: FUSES ERKLÄREN!)

4.4. Einfaches Beispiel für LEDs und Taster

4.5. LCD Bibliothek

4.6. RS-232

4.7. Menü Bibliothek

4.8. Interrupts

(TODO: IN DEN BEGRIFFEN ERKLÄREN! VERLINKEN!)

4.8.1. Endschalter

4.8.2. Watchdog

4.9. Protokoll der Stepperkarte

4.10. Manueller Betrieb

4.11. Protokolle aus RapidForm

4.12. Übersetungs Logik

4.13. Automatische Protokollwahl

5. Fazit und Zukunft

5.1. Fazit

(TODO: FAZIT SCHREIBEN!)

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich den vorliegenden Bericht:

Uebersetzen von Schrittmotorprotokollen
Entwurf eines Hardwareübersetzers

selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel verfasst habe.

Remagen, den 2. Februar 2012



JOHANNES DIELMANN

A. Anhang

A.1. Schritt für Schritt Anleitung

Eine Schritt für Schritt Anleitung zum vollständigen Scannen und exportieren eines 3D-Objektes.

A.2. Vom Autor verwendete Software

Hier ist die verwendete Software aufgelistet. Soweit es möglich war, wurden Open-Source-Programme eingesetzt. **(TODO: ÜBERARBEITEN!!!)**

- **RapidForm2004**

asdf

- **AVRStudio 5**

Atmel.

Website: <http://www.atmel.com/>

- **Eclipse**

Eclipse mit CDT und AVRPlugin

Website: <http://www.eclipse.org/>

- **AVRDude**

Prorammer

(TODO: WEITERE?!)