

C und LEGO

Ein Projektorientiertes Studium

Mitwirkende: Benjamin Buchholz, Felix Schwarz, Norman Seiffert, Friedemann Ehrke, Daniel Elsholz, Stephan Lapoehn, Lukas Postupa, Benjamin Schwarze, Carsten Schwenke



Das Projekt C und LEGO

Von der FHB haben wir in der Einführungswoche die Aufgabe bekommen, innerhalb des Projektorientierten Studiums (PST) uns ein Thema auszusuchen, um dieses dann im Rahmen des PST zu bearbeiten. Wir haben uns für das Thema "C und LEGO" entschieden. Hier werden Grundlagen der Programmiersprache ANSI-C vermittelt. Das ganze läuft daraus hinaus, dass wir mehrere kleinere Anwendungen in ANSI-C schreiben und es dann auf das AKSEN-Board übertragen. Dieses Board ist für uns die Schnittstelle zwischen Hardware (Sensoren, Aktoren) und Software. Hier können wir kleine bis große Roboter bzw. Intelligente Systeme entwickeln. Angefangen von Lichtschranken, Tresorschlossern, Motoren, die auf Abstand reagieren, bis hin zu "Köpfen", die einem Objekt folgen.

In unserem PST wollen wir darstellen, wie man das AKSEN-Board auf einfachste Weise mit Sensoren und Aktoren bestücken kann. Des weiteren wird gezeigt, wie die Software funktioniert und darauf arbeitet. Im Folgenden wird die Aufgabe Lichtschranke (Abb. 1) beschrieben.

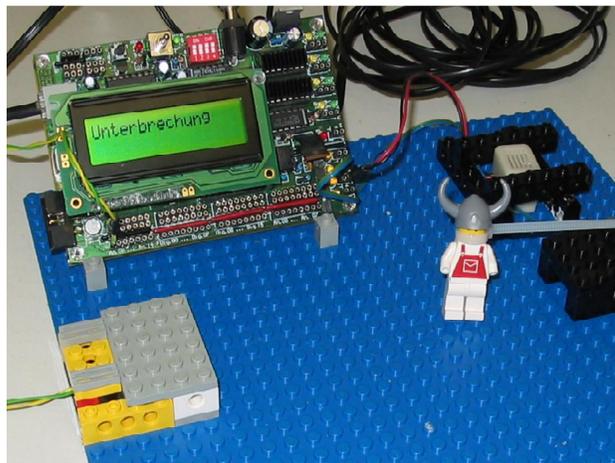


Abb.1: Auf dem Bild ist die Aufgabe Lichtschranke dargestellt. Man erkennt deutlich, dass die Unterbrechung gerade stattfindet.

Lichtschranke

Die Aufgabe bestand darin, dass man eine min. 150mm lange Lichtschranke baut, die Unterbrechungen anzeigt.

Hier muss sichergestellt werden, dass die Lichtschranke auch bei kürzester Unterbrechung auslöst. Des weiteren muss gewährleistet sein, dass die Unterbrechungsanzeige wieder erlischt, nachdem die Lichtschranke durchschritten wurde. Man muss allerdings auch beachten, dass wenig Fremdlicht einfällt. Man würde sonst ein Fehlverhalten erzielen. Dies geschieht durch die im Sonnenlicht vorhandenen IR-Strahlen, weil die Lichtschranke mit IR-Sensoren und IR-Empfängern arbeitet. Wir haben dieses Problem durch eine geschickte Konstruktion gelöst. Der Programmcode musste in ANSI-C geschrieben werden. Dabei stellten sich keine größeren Probleme ein.

Der Aufbau der Konstruktion wurde mit LEGO Bausteinen vollzogen. Wir haben uns dafür eine quadratische Grundplatte genommen und die IR-LED und den IR-Empfänger darauf befestigt. Der Abstand betrug in etwa 170mm. Nach dem der IR-Empfänger fast nur noch das Licht der IR-LED „sehen“ konnte, wurden beide mit dem AKSEN-Board verbunden. Das System arbeitet tadellos. Um in der Zukunft eine bessere Effektivität zu erreichen, kann man es auf Lasertechnik umstellen. Damit wäre es auch Tageslicht unabhängig.

Fingerfolger

Als nächstes wird die Aufgabe Fingerfolger (Abb. 2) dargestellt. Der Inhalt der Aufgabe bestand darin, einen in horizontaler Ebene drehbaren Kopf zu entwickeln, der einen Finger auf kurze Entfernung verfolgt.

Die Konstruktion (Abb. 2) wird durch einen Servomotor gesteuert. Zwei IR-Sensoren, sowie zwei IR-Sender werden dabei verwendet, um die Entfernung des Gegenstandes zu ermitteln. Es wird je ein Sensor für die linke und ein Sensor für die rechte Seite verwendet. Je nach Empfangsstärke (Höhe des Wertes) auf einer Seite im Vergleich zu der anderen, wird eine Veränderung der Bewegungsrichtung eingeleitet. Die Werte werden durch das AKSEN-Board verarbeitet und das Ausgangssignal an den Servomotor weiter gegeben. Das dazugehörige Programm wurde in ANSI-C (Abb. 4) geschrieben.

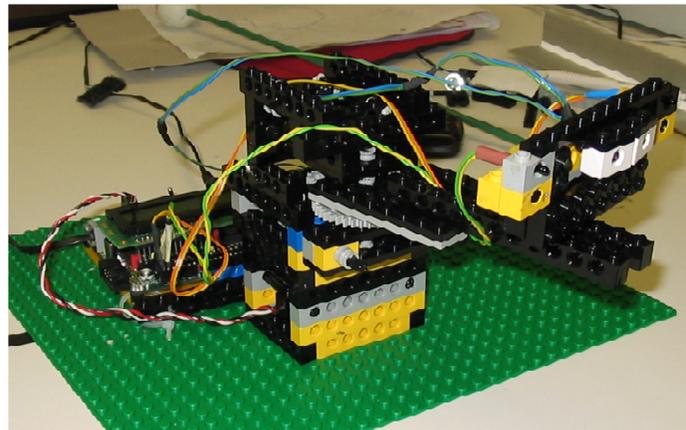


Abb.2: Der Fingerfolger vollständig funktionstüchtig und einsatzbereit.

Die aufgetretenen Probleme bei dieser Aufgabe sind folgende, die Beleuchtung des Raumes enthält ebenfalls IR-Licht, was zur Folge hat, dass die Werte des IR-Sensors durch äußere Einflüsse beeinflusst werden. Daher wurde zur Problemlösung vor jeder Messung die derzeitige Infrarotintensität im Raum, ohne die Infrarotsender gemessen und von späteren Messwert subtrahiert. Ein weiteres Problem war, dass der Fingerfolger zu einer Pendelbewegung neigt, wenn das zu verfolgende Objekt still steht. Dieses Problem wurde gelöst, indem ein bestimmter Grenzwert eingegeben wurde, welcher besagt, dass eine bestimmte Differenz (> 1) zwischen linkem und rechtem Sensor bestehen muss, um den Motor anzusteuern.

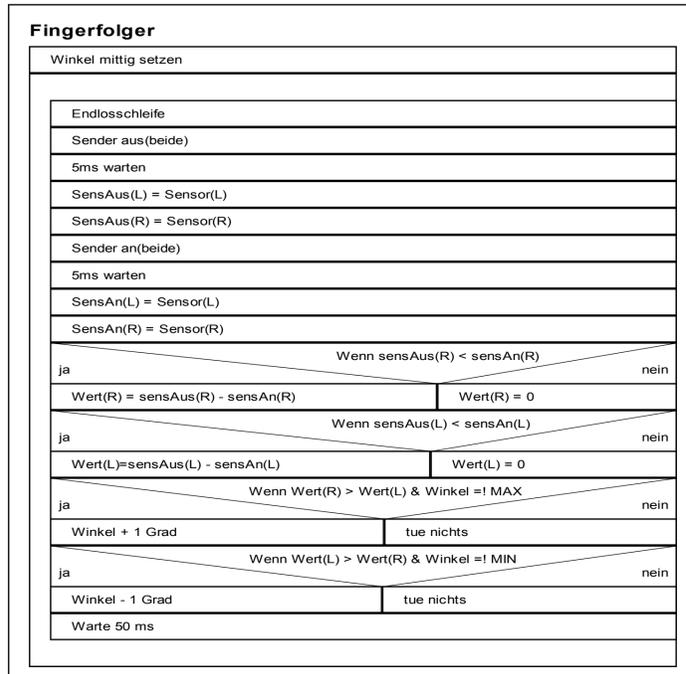


Abb.3: Das obige Struktogramm beschreibt die Aufgabe Fingerfolger.

Unter dem **Quellcode** (engl. source code) oder Quelltext versteht man in der Informatik den für Menschen lesbaren Teil eines Computerprogramms. Dieser Teil ist in einer Programmiersprache verfasst. In unserem Fall wäre dies ANSI-C. Dieser Code ist so formal exakt verfasst, dass der Computer ihn vollständig und unmissverständlich ausführen kann.

```
#include <stub.h>

#define MAX 140
#define MIN 40
#define KO 1

void AksenMain(void) {

    unsigned char s_links, s_rechts, winkel;
    unsigned char s_IR_off_R, s_IR_off_L;
    unsigned char s_IR_on_R, s_IR_on_L;

    winkel = 90;
    servo_arc(1,winkel);
    while(1){
        led(2,0); led(1,0); sleep(5);
        s_IR_off_R = analog(8);
        s_IR_off_L = analog(9);

        led(2,1); led(1,1); sleep(5);
        s_IR_on_R = analog(8);
        s_IR_on_L = analog(9);

        if (x<infT_a) s_rechts = s_IR_off_R - s_IR_on_R;
        else s_rechts = 0;

        if (y<infT_b) s_links = s_IR_off_L - s_IR_on_L;
        else s_links = 0;

        if(s_rechts > s_links && winkel < MAX
            && (s_rechts - s_links) > KO)
            winkel++;
        else if(s_rechts < s_links && winkel > MIN
            && (s_links - s_rechts) > KO)
            winkel--;

        servo_arc(1,winkel);

        lcd_cls();
        lcd_puts("R:"); lcd_ubyte(s_rechts);
        lcd_puts("L:"); lcd_ubyte(s_links);
        lcd_setxy(1,1);
        lcd_puts("Winkel :"); lcd_ubyte(winkel);
        sleep(50);
    }
}
```

Abb.4: Der obige Quellcode gehört zu der Aufgabe Fingerfolger. Aus Platzgründen auf diesem Poster, wurden die Kommentare im Quellcode weggelassen.

Hier nun noch ein kleine Liste der von uns bearbeiteten Aufgaben:

1. Helligkeitslaufflicht
2. Codeschloss
3. Lichtschranke (Abb. 1)
4. Unsichtbares Gaspedal
5. Fingerfolger (Abb. 2)
6. Der winkende Servo
7. Fußball-Schussanlage
8. Zähler
9. Schreibmaschine
10. Lichtschalter

Fazit

Auf dem AKSEN-Board lassen sich auf einfachste Art und Weise schon erstaunliche Programme und "Maschinen" realisieren. In dem PST haben wir gelernt, einfache Programme in ANSI-C zu schreiben, kompilieren und auf das AKSEN-Board zu laden. Des weiteren kennen wir einfache Sensorik und Aktorik. Wir können eine Lösungsidee in einen Algorithmus und einen Versuchsaufbau umsetzen. Zum Abschluß kann man sagen, das wir uns alle für das richtige Thema entschieden haben.