
Projektorientiertes Studium 2007

C und LEGO

Eine Multimedia-Präsentation mit Videos

Projektgruppen:

FuM: Benjamin S. & Friedemann

FCL: Felix, Carsten & Lukas

BuLa: Benjamin B. & Stephan

NuD: Norman & Daniel



Inhalte

1. Vorstellung des Projekts „C und LEGO“
 - a) Einführung
 - b) AKSEN-Board
 - c) Aktoren & Sensoren
2. Der Praxisteil
 - a) Aufgabenüberblick
 - b) Eine Maschine entsteht
 - c) Projektvorstellung: „Lichtschanke“ (leicht)
 - d) Projektvorstellung: „Fingerfolger“ (fortgeschritten)
3. Fazit

Inhalte

1. **Vorstellung des Projekts „C und LEGO“**
 - a) **Einführung**
 - b) **AKSEN-Board**
 - c) **Aktoren & Sensoren**
2. **Der Praxisteil**
 - a) **Aufgabenüberblick**
 - b) **Eine Maschine entsteht**
 - c) **Projektvorstellung: „Lichtschanke“ (leicht)**
 - d) **Projektvorstellung: „Fingerfolger“ (fortgeschritten)**
3. **Fazit**

1.a Einführung

Lernziele:

1. Probleme erkennen und lösen

Bau einer **Maschine** nach Vorgaben einer einfachen Aufgabenstellung

à Erläuterung der Funktionsweise einiger Aktoren & Sensoren und deren Verwendungen mit dem AKSEN-Board

1.a Einführung

Lernziele:

1. Probleme erkennen und lösen

Bau einer **Maschine** nach Vorgaben einer einfachen Aufgabenstellung

à Erläuterung der Funktionsweise einiger Aktoren & Sensoren und deren Verwendungen mit dem AKSEN-Board

2. algorithmisches Denken und Kreativität

Umsetzung eines **Programms**, das der Maschine unter Verwendung zuvor gewählter Sensoren & Aktoren die gewünschte Funktionsweise einhaucht

à Vermittlung grundlegender Kenntnissen in der Programmiersprache C (Variablen, Kontrollstrukturen, Funktionen)

1.b AKSEN-Board



Wissenswertes:

- Name bildet sich aus **AK**toren und **SEN**soren
- Eigenentwicklung der FHB

Ausstattung:

- 8-bit-Mikroprozessor C515 von Siemens
- serielle Schnittstelle
- 16 analoge und 16 digitale Sensorports
- diverse Elektronik zur Steuerung von Motoren, Servos, LED, LCD, usw.

1.c Aktoren & Sensoren

Aktoren:

Wandeln Energie in Bewegung, Licht oder andere Ausgabegrößen um.

Beispiele:

- Motoren und Getriebe
- Servo-Motoren
- Glühlämpchen, Infrarotsender, etc.

Sensoren:

Messen physikalische Größen und wandeln diese in elektrische Signale um.

Beispiele:

- Mikroschalter (Berührung)
- Fotosensoren (sichtbares Licht)
- Infrarotempfänger (infrarote Signale oder Entfernungen)

Inhalte

1. Vorstellung des Projekts „C und LEGO“
 - a) Einführung
 - b) AKSEN-Board
 - c) Aktoren & Sensoren
2. **Der Praxisteil**
 - a) **Aufgabenüberblick**
 - b) **Eine Maschine entsteht**
 - c) **Projektvorstellung: „Lichtschanke“ (leicht)**
 - d) **Projektvorstellung: „Fingerfolger“ (fortgeschritten)**
3. **Fazit**

2.a Aufgabenüberblick

<p>Aufgabe 16 8 Punkte</p> <p>Unsichtbare Lichtschranke</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 17 10 x (Boardanzahl -1) Punkte</p> <p>Signalstrecke</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 18 7 Punkte</p> <p>Entfernungsmesser</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 19 13 Punkte</p> <p>lichtabhängiger Entfernungsmesser</p> <p><small>10 Punkte</small></p>
<p>Aufgabe 20 20 Punkte</p> <p>Fingerfolger</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 21 20 Punkte</p> <p>Programmierbarer Blinker</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 22 6 Punkte</p> <p>Binärzähler</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 23 6 Punkte</p> <p>Anzeige DIP-Schalter</p> <p><small>10 Punkte</small></p>
<p>Aufgabe 24 21 Punkte</p> <p>Fußball-Schussanlage</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 25 20 Punkte</p> <p>Fliegenklatsche</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 26 9 Punkte</p> <p>Schwarz-Weiß Anzeiger</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 27 20 Punkte</p> <p>Sortierer</p> <p><small>10 Punkte</small></p>
<p>Aufgabe 28 30 Punkte</p> <p>Die Maschine</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 29 16 Punkte</p> <p>Schreibmaschine</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 30 17 Punkte</p> <p>Fernsteuerung</p> <p><small>10 Punkte</small></p>	<p>Aufgabe 31 16 Punkte</p> <p>Unsichtbares Gaspedal</p> <p><small>10 Punkte</small></p>

2.b Eine Maschine entsteht

Erst das Kinde im Mann...



... und dann, das Programm à

2.b Eine Maschine entsteht (Programm)



2.b Eine Maschine entsteht (Programm)

The screenshot shows the Visual Studio IDE with a C program named '2b_programm.c'. The code defines a function 'ReflektionMessen' that reads sensor data and controls a servo motor. A terminal window shows the compilation process using 'sdcc' and 'aslink.exe'. A 'AkSen-Flasher' dialog box is also visible, showing the file path and status.

```
1 // Fingerfolger
2 // (c) Benjamin Schwarze & Friedemann Ehke 2007
3
4 /*
5 Das Programm steuert einen Kopf auf einem Servomotor, sodass dieser
6 einem Finger in ca. 10cm Reflektion folgt.
7 */
8
9 // Alle Funktionen der AkSen-Bibliothek bekannt machen
10 #include <stub.h>
11
12 #define SERVO_MIN 10
13 #define SERVO_MAX 170
14 #define REAKTION 2
15
16 unsigned char ReflektionMessen(unsigned char port)
17 {
18     unsigned char ohne_led, mit_led, r;
19
20     led(port, 0); // IR-Diode ausschalten
21     sleep(10);
22     ohne_led = analog(port); // Sensorwert ohne IR-Diode
23
24     led(port, 1); // IR-Diode anschalten
25     sleep(10);
26     mit_led = analog(port); // Sensorwert mit IR-Diode
27
28     // Rückgabewert clippen
29     if(ohne_led > mit_led) r = ohne_led - mit_led;
30     else r = 0;
31
32     return (r); // kalibrierten Wert zurückliefern
33 }
34
35 //Hauptprogrammroutine
36 void AkSenMain(void)
37 {
38     unsigned char servo_pos = 90; // Servomotor in Grundstellung initialisieren
39     unsigned char reflektion_links = 0; // Wert des linken Sensorpaars
40     unsigned char reflektion_rechts = 0; // Wert des rechten Sensorpaars
41
42     while(1)
43     {
44         // Reflektionswerte messen
45         reflektion_links = ReflektionMessen(0);
46         reflektion_rechts = ReflektionMessen(1);
47
48         if((reflektion_links - reflektion_rechts) >= REAKTION) // Objekt ist links: nach links drehen
49             if(servo_pos < SERVO_MAX-1) servo_pos+=2; // Servowinkel clippen / erhöhen
50         if((reflektion_rechts - reflektion_links) >= REAKTION) // Objekt ist rechts: nach rechts drehen
51             if(servo_pos > SERVO_MIN+1) servo_pos-=2; // Servowinkel clippen / erniedern
52     }
```

```
MI-Labor@kila9 ~
$ cd aksen
MI-Labor@kila9 ~/aksen
$ cd fun_fingerfolger
MI-Labor@kila9 ~/aksen/fun_fingerfolger
$ make
sdcc -U -c --model-large --data-loc 0x18 --stack-auto -I C:\cygwin\usr\local\Naksen-lib\include -I fingerfolger.c
+ c:\PROGRAM\SDCC\bin\sdccpp.exe -nostdinc -Wall -std=c99 -I"C:\cygwin\usr\local\Naksen-lib\include" -obj-ext=.rel -DSDCC_STACK_AUTO -DSDCC_MODEL_LARGE -DSDCC-270 -DSDCC_mcs51 -D__mcs51 -I"C:\Programme\SDCC\bin\..\include\mcs51" -I"C:\Programme\SDCC\bin\..\include" -I fingerfolger.c
+ c:\PROGRAM\SDCC\bin\aslink.exe -plogff "fingerfolger.asm"
cp "/usr/local/aksen-lib/stub/AkSen.rel" .
cp "/usr/local/aksen-lib/stub/AkSen.lst" .
sdcc -U --model-large --data-loc 0x18 --stack-auto -Llarge AkSen.rel fingerfolge
r.rel
+ c:\PROGRAM\SDCC\bin\aslink.exe -of "AkSen.lnk"
as AkSen.ihx fingerfolger.ihx
rm *.lst *.sym *.rel *.asm -f
rm *.rst *.map *.mem *.lnk -f
ls -l fingerfolger.ihx
-rwxr-xr-x 1 MI-Labor kein 2591 Nov 22 13:01 fingerfolger.ihx
```

AkSen-Flasher

Dateiauswahl: in:\home\MI-Labor\aksen\fun_fingerfolger\fingerfolger.ihx [Datei]

Schnittstelle: COM2

Status: Version 1.4, 21.06.2006, Kai-Uwe Mikor

Buttons: Flashen, Beenden

1. programmieren / editieren

2.b Eine Maschine entsteht (Programm)

The screenshot displays the Microsoft Visual Studio IDE with a C program named `2D_programm.c` open. The code is a servo motor control program that reads an IR sensor and adjusts the servo position. A blue arrow points from the text **2. compilieren** to the code. In the background, a terminal window shows the compilation process using `sdcc` and `aslink.exe`. Another window titled **AkSen-Flasher** is visible, showing a file selection dialog and a status box.

```
1 // Fingerfolger
2 // (c) Benjamin Schwarze & Friedemann Ehke 2007
3
4 /*
5 Das Programm steuert einen Kopf auf einem Servomotor, sodass dieser
6 einem Finger in ca. 10cm Reflektion folgt.
7 */
8
9 // Alle Funktionen der AkSen-Bibliothek bekannt machen
10 #include <stub.h>
11
12 #define SERVO_MIN 10
13 #define SERVO_MAX 170
14 #define REAKTION 2
15
16 unsigned char ReflektionMessen(unsigned char port)
17 {
18     unsigned char ohne_led, mit_led, r;
19
20     led(port, 1); // IR-Diode anschalten
21     sleep(10);
22     ohne_led = analog(port); // Sensorwert mit IR-Diode
23
24     led(port, 0); // IR-Diode ausschalten
25     sleep(10);
26     mit_led = analog(port); // Sensorwert mit IR-Diode
27
28     // Rückgabewert clippen
29     if(ohne_led > mit_led) r = ohne_led - mit_led;
30     else r = 0;
31
32     return (r); // kalibrierten Wert zurückliefern
33 }
34
35 //Hauptprogrammroutine
36 void AkSenMain(void)
37 {
38     unsigned char servo_pos = 90; // Servomotor in Grundstellung initialisieren
39     unsigned char reflektion_links = 0; // Wert des linken Sensorpaares
40     unsigned char reflektion_rechts = 0; // Wert des rechten Sensorpaares
41
42     while(1)
43     {
44         // Reflektionswerte messen
45         reflektion_links = ReflektionMessen(0);
46         reflektion_rechts = ReflektionMessen(1);
47
48         if((reflektion_links - reflektion_rechts) >= REAKTION) // Objekt ist links: nach links drehen
49             if(servo_pos < SERVO_MAX-1) servo_pos+=2; // Servowinkel clippen / erhöhen
50         if((reflektion_rechts - reflektion_links) >= REAKTION) // Objekt ist rechts: nach rechts drehen
51             if(servo_pos > SERVO_MIN+1) servo_pos-=2; // Servowinkel clippen / erniedern
52     }
53 }
```

```
MI-Labor@k1ab9 ~
$ cd aksen
MI-Labor@k1ab9 ~/aksen
$ cd fun_fingerfolger
MI-Labor@k1ab9 ~/aksen/fun_fingerfolger
$ make
sdcc -U -c --model-large --data-loc 0x18 --stack-auto -I C:\cygwin\usr\local\Naksen-lib\include -I fingerfolger.c
+ c:\PROGRAM\TSDCC\bin\sdccpp.exe -nostdinc -Wall -std=c99 -I"C:\cygwin\usr\local\Naksen-lib\include" -obj-ext-.rel -DSDCC_STACK_AUTO -DSDCC_MODEL_LARGE -DSDCC-27
B -DSDCC_mcs51 -D__mcs51 -I"C:\Programme\SDCC\bin\..\include\mcs51" -I"C:\Programme\SDCC\bin\..\include" -I fingerfolger.c
+ c:\PROGRAM\TSDCC\bin\asx8051.exe -plogoff "fingerfolger.asm"
cp /usr/local/aksen-lib/stub/AkSen.rel .
cp /usr/local/aksen-lib/stub/AkSen.lst .
sdcc -U --model-large --data-loc 0x18 --stack-auto -Llarge AkSen.rel fingerfolge
r.rel
+ c:\PROGRAM\TSDCC\bin\aslink.exe -of "AkSen.lnk"
as AkSen.ihx fingerfolger.ihx
rm *.lst *.sym *.rel *.asm -f
rm *.rst *.map *.mem *.lnk -f
ls -l fingerfolger.ihx
-rwxr-xr-x 1 MI-Labor kein 2591 Nov 22 13:01 fingerfolger.ihx
$
```

2. compilieren

AkSen-Flasher

Dateiauswahl:

Schnittstelle: Status:

2.b Eine Maschine entsteht (Programm)

The screenshot shows the Visual Studio IDE with a C program named '2b_programm.c'. The code includes comments in German and defines constants for servo motor control. It features a function 'ReflektionMessen' that reads sensor data and a 'main' function that initializes a servo motor and enters a loop to measure reflections and control the servo's position.

```
1 // Fingerfolger
2 // (c) Benjamin Schwarze & Friedemann Ehke 2007
3
4 /*
5 Das Programm steuert einen Kopf auf einem Servomotor, sodass dieser
6 einem Finger in ca. 10cm Reflektion folgt.
7 */
8
9 // Alle Funktionen der AkSen-Bibliothek bekannt machen
10 #include <stub.h>
11
12 #define SERVO_MIN 10
13 #define SERVO_MAX 170
14 #define REAKTION 2
15
16 unsigned char ReflektionMessen(unsigned char port)
17 {
18     unsigned char ohne_led, mit_led, r;
19
20     led(port, 0); // IR-Diode ausschalten
21     sleep(10);
22     ohne_led = analog(port); // Sensorwert ohne IR-Diode
23
24     led(port, 1); // IR-Diode anschalten
25     sleep(10);
26     mit_led = analog(port); // Sensorwert mit IR-Diode
27
28     // Rückgabewert clippen
29     if(ohne_led > mit_led) r = ohne_led - mit_led;
30     else r = 0;
31
32     return (r); // kalibrierten Wert zurückliefern
33 }
34
35 //Hauptprogrammroutine
36 void AkSenMain(void)
37 {
38     unsigned char servo_pos = 90; // Servomotor in Grundstellung initialisieren
39     unsigned char reflektion_links = 0; // Wert des linken Sensorpaares
40     unsigned char reflektion_rechts = 0; // Wert des rechten Sensorpaares
41
42     while(1)
43     {
44         // Reflektionswerte messen
45         reflektion_links = ReflektionMessen(0);
46         reflektion_rechts = ReflektionMessen(1);
47
48         if(reflektion_rechts > reflektion_links) // Objekt ist links: nach links drehen
49             servo_pos -= 1;
50         if((reflektion_rechts - reflektion_links) >= REAKTION) // Objekt ist rechts: nach rechts drehen
51             servo_pos += 1;
52     }
53 }
```

The terminal window shows the compilation process:

```
MI-Labor@k1lab9 ~
$ cd akSen
MI-Labor@k1lab9 ~/akSen
$ cd fun_fingerfolger
MI-Labor@k1lab9 ~/akSen/fun_fingerfolger
$ make
sdcc -U -c -mmodel-large -data-loc 0x18 --stack-auto -I C:\cygwin\usr\local\Naksen-lib\include -fingerfolger.c
+ c:\PROGRAM\SDCC\bin\sdcc.exe -nostdinc -Wall -std=c99 -I"C:\cygwin\usr\local\Naksen-lib\include" -obj-ext=.rel -DSOCC_STACK_AUTO -DSOCC_MODEL_LARGE -DSOCC-270 -DSOCC_mcs51 -D__mcs51 -I"C:\Programme\SDCC\bin\..\include\mcs51" -I"C:\Programme\SDCC\bin\..\include" -fingerfolger.c
+ c:\PROGRAM\SDCC\bin\asx8051.exe -plogoff "fingerfolger.asm"
cp /usr/local/aksen-lib/stub/AkSen.rel .
cp /usr/local/aksen-lib/stub/AkSen.lst .
sdcc -U -mmodel-large -data-loc 0x18 --stack-auto -Llarge AkSen.rel fingerfolge
r.rel
+ c:\PROGRAM\SDCC\bin\aslink.exe -of "AkSen.lnk"
mv AkSen.ihx fingerfolger.ihx
rm *.lst *.sym *.rel *.asm -f
rm *.rst *.map *.mem *.lnk -f
ls -l fingerfolger.ihx
-rwxr-xr-x 1 MI-Labor kein 2591 Nov 22 13:01 fingerfolger.ihx
$
```

3. auf AKSEN-Board übertragen

2.c Projektvorstellung: „Lichtschranke“

Aufgabe: Bauen Sie eine mindestens 15cm lange unsichtbare Lichtschranke und zeigen Sie Unterbrechungen an.

Problemstellung:

1. IR-Sender/Empfänger vor Fremdlicht schützen, sonst keine saubere Auslösung
2. Bestimmung Sensorwertebereich einer aktiven Unterbrechung
3. Unterbrechungsbehandlung muss entsprechend wieder gestoppt werden

Lösung:

1. Empfänger wurde versenkt eingebaut, um das Sehfeld einzuschränken
2. Probeläufe liefern Wertebereiche für Unterbrechung
3. Mit Hilfe von Fallunterscheidung

2.c Projektvorstellung: „Lichtschranke“

Aufgabe: Bauen Sie eine mindestens 15cm lange unsichtbare Lichtschranke und zeigen Sie Unterbrechungen an.

Die Maschine:

1 x AKSEN-Board

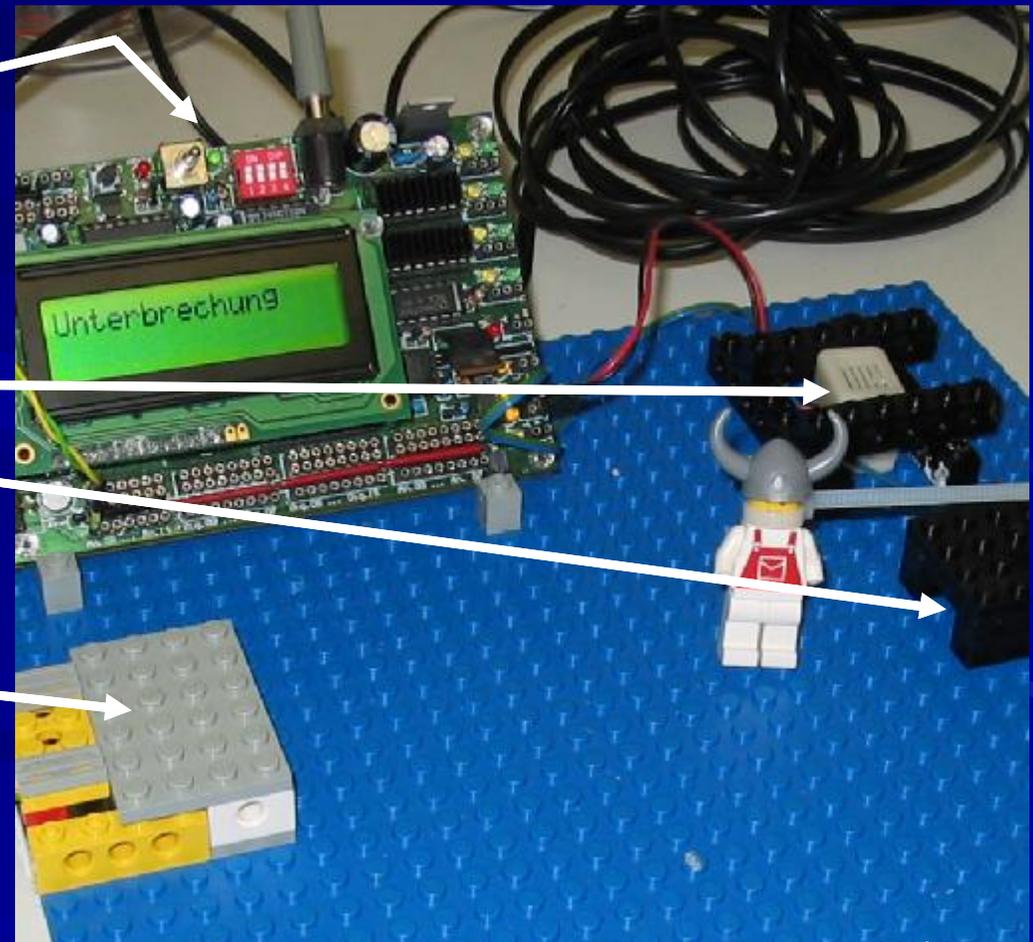
Aktoren:

1 x Piepser

1 x Infrarot-Diode

Sensoren:

1 x IR-Empfänger



2.c Projektvorstellung: „Lichtschranke“

Struktogramm:

Lichtschranke

Variable für IR-Sensor initialisieren		
IR-LED einschalten		
AKSEN Endlos-Schleife		
W H - M E	Wert von Infrarot-Empfänger abfragen	
	Infrarot-Wert > 150	
	LCD-Cursor auf 0,0 (x,y)	
	Pieper ausschalten	
	LCD-Anzeige löschen	
Pieper einschalten		

2.d Projektvorstellung: „Fingerfolger“

Aufgabe: Ein Servomotor trägt einen Kopf, der einen Finger in ca. 10cm Entfernung hin und her folgt.

Problemstellung:

1. dem Finger (Reflektor) in die richtige Richtung folgen
2. Reflektionstärke des infraroten Lichts von der Umgebungshelligkeit unabhängig messen und auswerten

Lösung:

1. Zwei Sensorenpaare als „Augen“ → in welche Richtung bewegt sich der Finger?
2. Doppelte Messung der Reflektion: einmal mit und einmal ohne zusätzlichem IR-Licht durch die Diode → durch Subtraktion entsteht der eigentliche Wert

2.d Projektvorstellung: „Fingerfolger“

Aufgabe: Ein Servomotor trägt einen Kopf, der einen Finger in ca. 10cm Entfernung hin und her folgt.

Die Maschine:

1 x AKSEN-Board

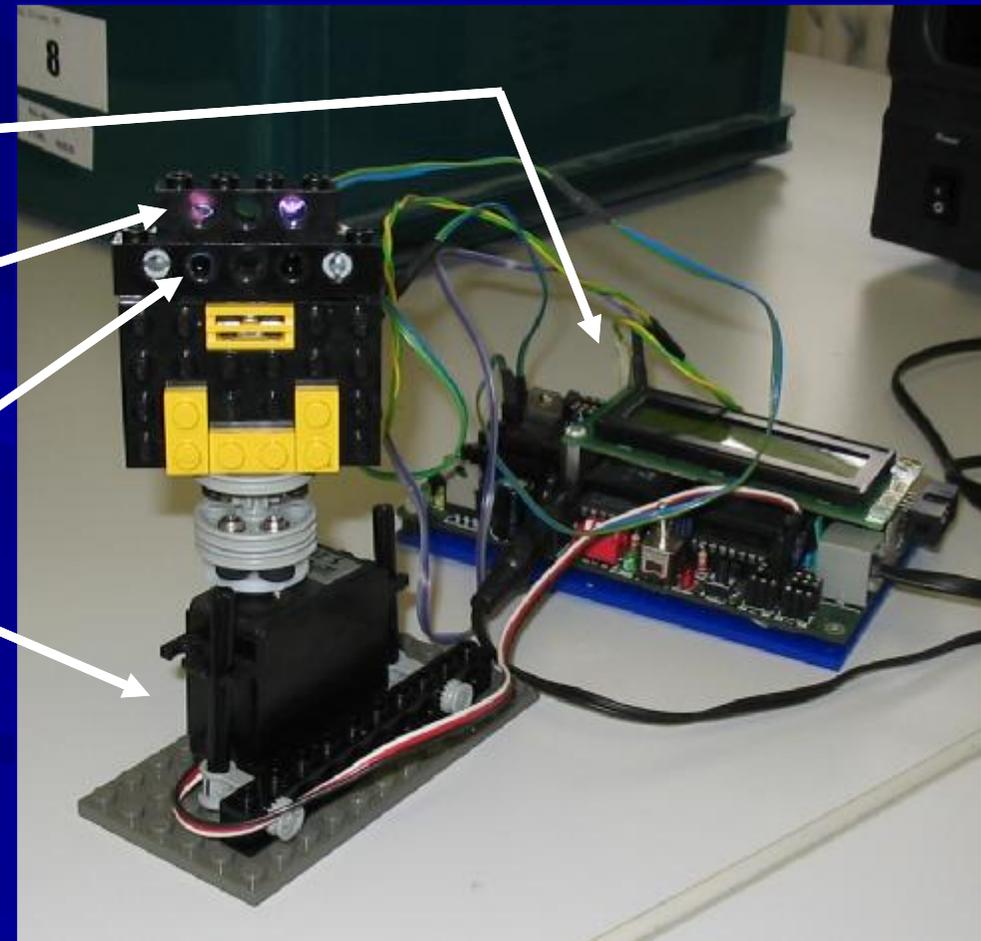
Aktoren:

2 x Infrarot-Diode

1 x Servo-Motor

Sensoren:

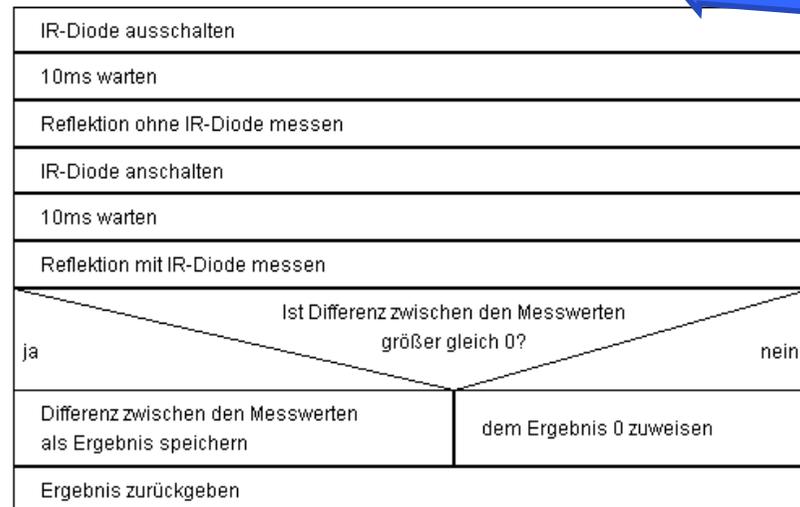
2 x IR-Empfänger



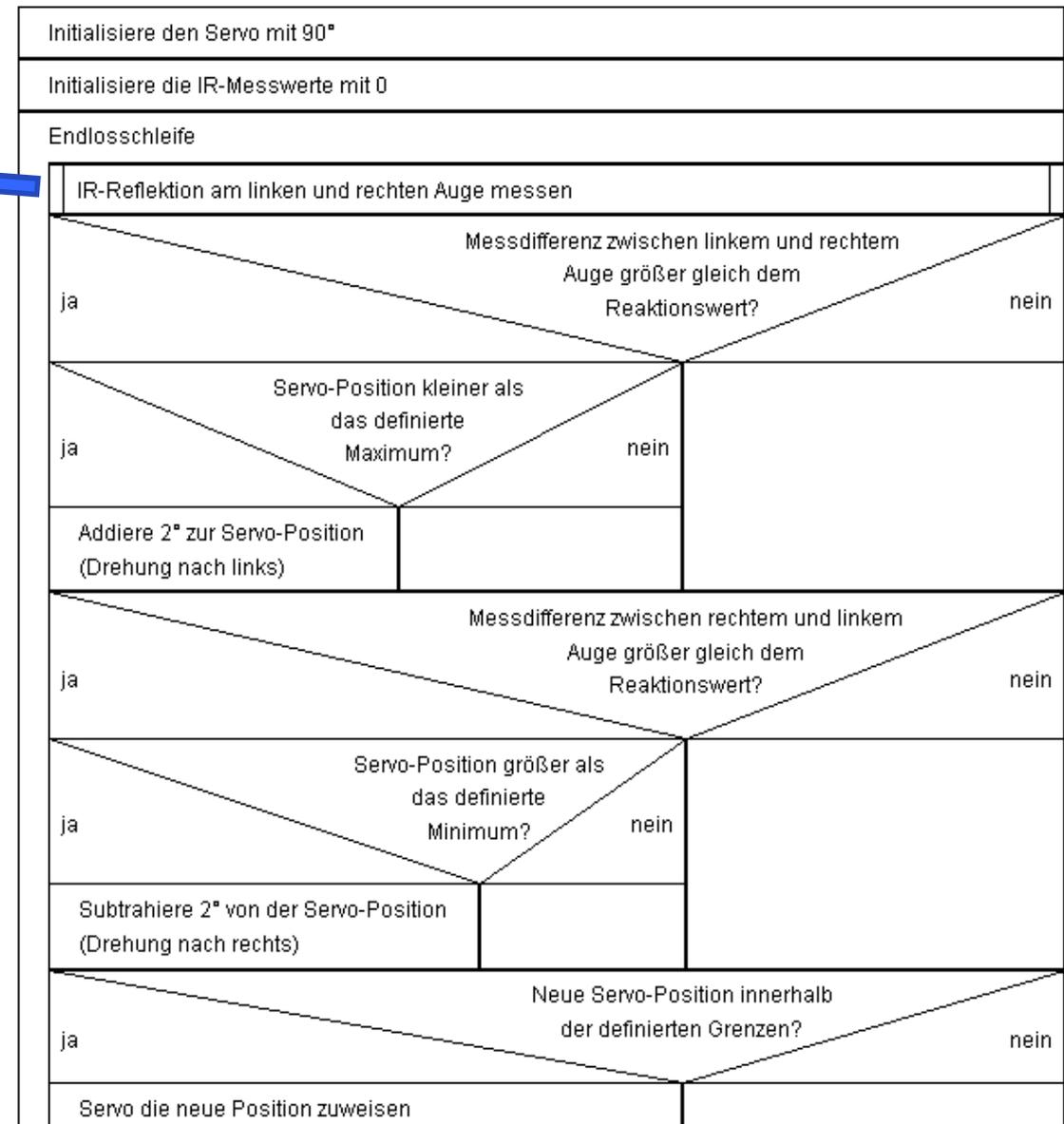
2.d Projektvorstellung: „Fingerfolger“

Struktogramm:

ReflektionMessen



Fingerfolger



Inhalte

1. Vorstellung des Projekts „C und LEGO“
 - a) Einführung
 - b) AKSEN-Board
 - c) Aktoren & Sensoren
2. Der Praxisteil
 - a) Aufgabenüberblick
 - b) Eine Maschine entsteht
 - c) Projektvorstellung: „Lichtschanke“ (leicht)
 - d) Projektvorstellung: „Fingerfolger“ (fortgeschritten)
3. **Fazit**

3. Fazit

2007 C und LEGO

Gruppe	Studenten	Zusätzlich	#	Pkte
BuLa	Buchholz Benjamin, Lapoehn Stephan		2	30
FCL	Schwarz Felix, Postupa Lukas, Schwenke Carsten	Poster	3	44
Norman U Daniel	Elsholz Daniel, Seiffert Norman		4	48
FuM	Schwarze Benjamin, Ehrke Friedemann	Vortrag	2	12

Aufgabennr	Titel	Gruppe	Program					Titel	Aufgabe	Bild 300px	Summe (16=ok)
			Versuch	PAP/SG	m	Film	.c				
7	Lichtschalter Flur	Norman U Daniel	1 (*1)	1 (*2)	1 (*3)	2	2	2	2	2	10
8	Zähler (Taster)	Norman U Daniel	2	2	2	2	2	2	2	2	16
10	Codeschloß	Norman U Daniel	2	1 (*6)	1 (*5)	2	2 (*4)	2	2	2	10
11	Der winkende Servo	FCL	2	2	2	2	2	2	2	2	16
14	Helligkeitslauflicht	BuLa	2	2	2	2	2	2	2		14
16	Unsichtbare Lichtschranke	FCL	2	1 (*7)	1 (*7)	2	2	2	2	2	12
20	Fingerfolger	BuLa	2	2	2	2	2	2	2	2	16
20	Fingerfolger	FuM		2		2	2			2	8
24	Fussball-Schußanlage	FCL	2	2	2	2	2	2	2	2	16
29	Schreibmaschine	Norman U Daniel	2		1 (*5)	2	2	2	2	2	12
31	Unsichtbares Gaspedal	FuM				2	2				4
Summe	Zum Veröffentlichen geeignet:		8	8	5	11	10	9	9	9	

Legende:

- 2 = vorhanden und zur Veröffentlichung geeignet
- 1 = vorhanden, aber Nacharbeiten notwendig
- leer = nicht vorhanden

Vielen Dank für eure Aufmerksamkeit! ;-)