



Lego 4 LavA

Entwicklung einer Lego Mindstorms
Experimentierplattform für FPGA-basierte,
konfigurierbare Multiprozessorsysteme

Nico Stute

Bachelorarbeit Abschlussvortrag

Lehrstuhl für Informatik 12
TU Dortmund

nico.stute@udo.edu

26. September 2011



ESS



Inhalt

Ziele der Arbeit

Anforderungen

Sensorinterface

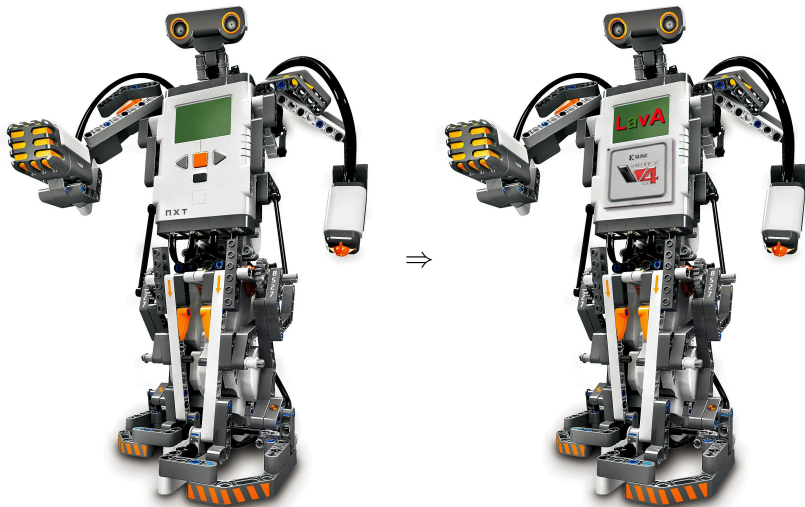
Aktorinterface

Errata

Prototyp

Demo







Ziele der Arbeit

- Implementierung eines konfigurierbaren Sensor/Aktor-Interfaces in VHDL
- Einbindung des Interfaces in LavA
- Entwicklung der Hardware um Lego Sensoren/Aktoren anzusteuern
- Schreiben einer Steuerbibliothek (C/C++)
- Bau eines Lego-Demonstrators zur Evaluierung



Anforderungen

- LEGO
 - I²C-Interface
 - Analogeingang
 - H-Brücke
 - Quadratureingang

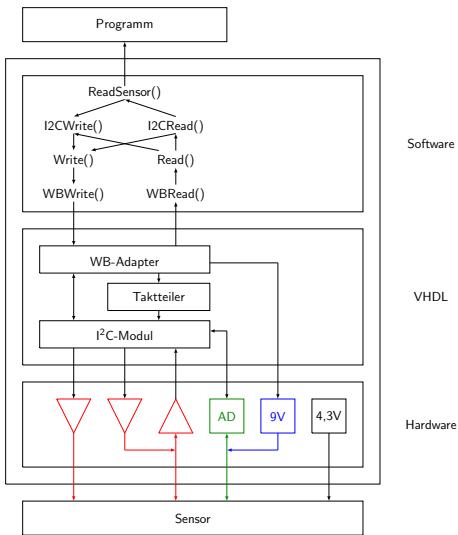


Anforderungen

- LEGO
 - I²C-Interface
 - Analogeingang
 - H-Brücke
 - Quadratureingang
- LavA
 - Wishbone-Interface
 - C-Bibliothek

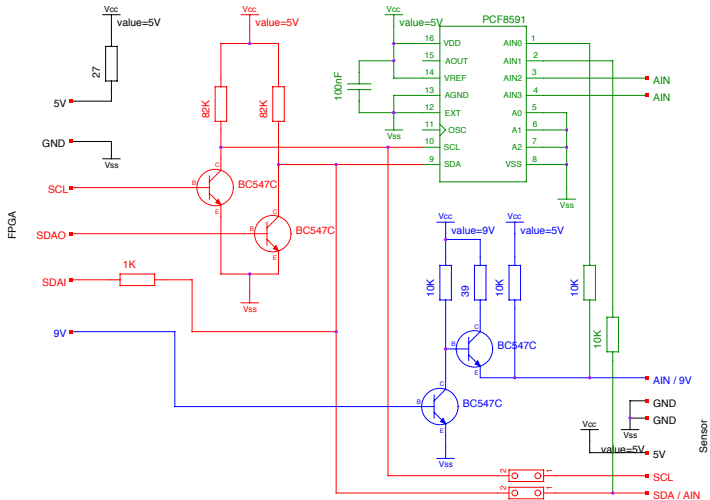


Sensorinterface



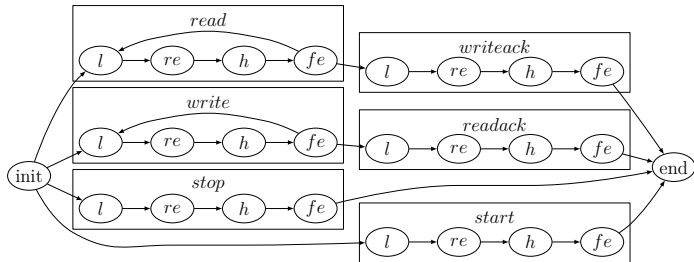
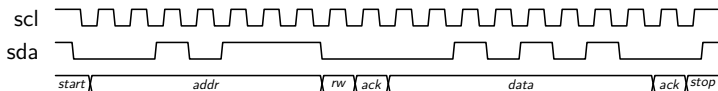


Sensorinterface - Hardware





Sensorinterface - VHDL





Sensorinterface - Software

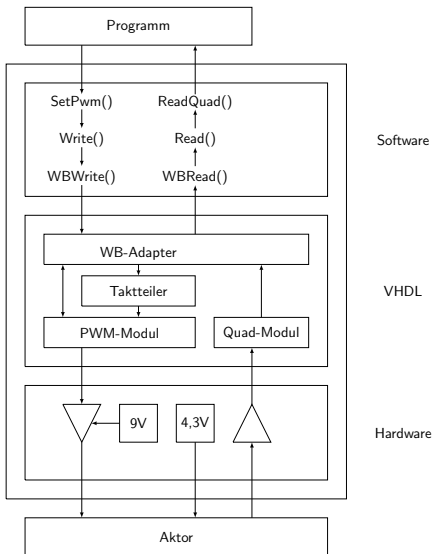
```
unsigned char read_LEGO(unsigned char c, volatile int * addr){
    unsigned char i;
    I2C_start(addr);
    I2C_write(LEGO, addr);
    I2C_write(c + 0x42, addr);
    I2C_stop(addr);

    I2C_start(addr);
    I2C_write(LEGO | READ, addr);
    i = I2C_read(0, addr);
    I2C_stop(addr);
    return(i);
}

unsigned char read_sensor(unsigned char type, unsigned char c, volatile int * addr){
    if(type == LEGO){
        return(read_LEGO(c, addr));
    }
    return(read_PCF(c, addr));
}
```

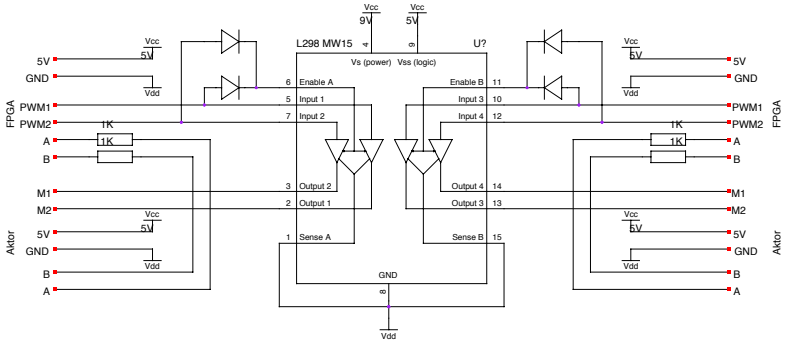


Aktorinterface





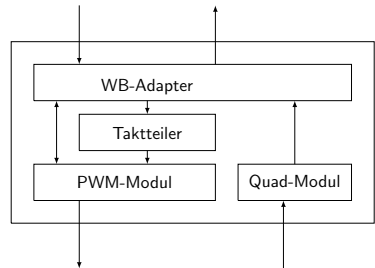
Aktorinterface - Hardware





Aktorinterface - VHDL

- PWM
 - 8 Bit
 - Bremsen, Links-, Recht-, Freilauf
- Quadratur
 - 16 Bit
 - Reset bei Lesezugriff
 - Fehlererkennung





Aktorinterface - Software

```
void set_speed(int speed, struct MOTOR motor){
    if(speed < 0){
        set_PWM(REV, (unsigned char) -speed, motor);
    }
    else{
        set_PWM(FWD, (unsigned char) speed, motor);
    }
}

int read_QUAD(struct MOTOR motor){
    int result = read(motor.addr) & 0x0000FFFF;
    if(result > 32768){
        result = result - 65536;
    }
    return(result);
}
```



Errata

- Firmwarebug im Ultraschallsensor



Errata

- Firmwarebug im Ultraschallsensor
- Reverse Engineering der Firmware



Errata

- Firmwarebug im Ultraschallsensor
- Reverse Engineering der Firmware
- 8 Bit AD-Wandler



Errata

- Firmwarebug im Ultraschallsensor
 - Reverse Engineering der Firmware
- 8 Bit AD-Wandler
 - MAX127 (8 × 12 Bit)
 - ADS7828 (8 × 12 Bit)
- Helligkeitssensor 1.0 blockiert den I²C-Bus



Errata

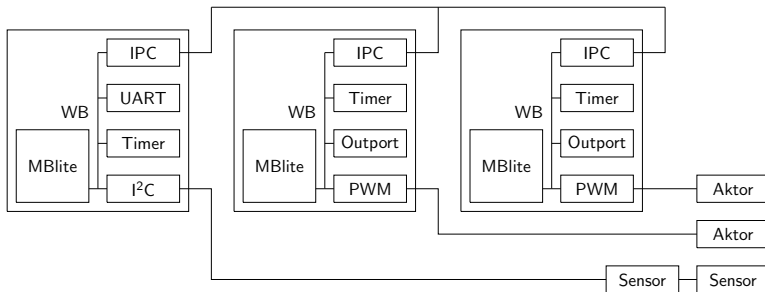
- Firmwarebug im Ultraschallsensor
 - Reverse Engineering der Firmware
- 8 Bit AD-Wandler
 - MAX127 (8 × 12 Bit)
 - ADS7828 (8 × 12 Bit)
- Helligkeitssensor 1.0 blockiert den I²C-Bus
- Farbsensor 2.0 verwendet anderes Interface



- Digilent Nexys 2
- Spartan 3E 1200
- 2 NXT Motoren mit Quadraturencoder
- Gyroskop und Beschleunigungssensor von HiTechnic

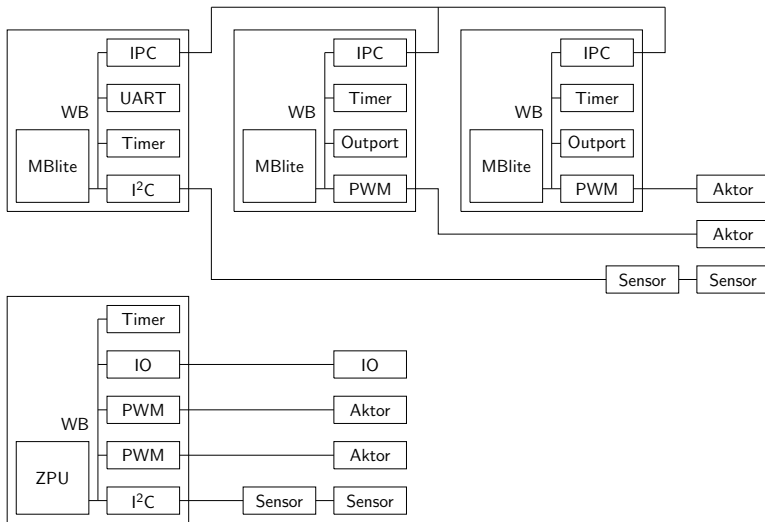


Prototyp





Prototyp





- 8 Bit AD-Wandler



Prototyp

- 8 Bit AD-Wandler
- Beschleunigungssensor



Prototyp

- 8 Bit AD-Wandler
- Beschleunigungssensor
 - Rauschen
 - Rückkopplung



Prototyp

- 8 Bit AD-Wandler
- Beschleunigungssensor
 - Rauschen
 - Rückkopplung
- Massenträgheit und Spiel der Motoren



- 8 Bit AD-Wandler
- Beschleunigungssensor
 - Rauschen
 - Rückkopplung
- Massenträgheit und Spiel der Motoren
 - geschachtelter PID-Regler



- 8 Bit AD-Wandler
- Beschleunigungssensor
 - Rauschen
 - Rückkopplung
- Massenträgheit und Spiel der Motoren
 - geschachtelter PID-Regler
 - Überschwingen
 - zu langsam



Demo



Lego MINDSTORMS Developer Files

<http://mindstorms.lego.com/en-us/support/files/default.aspx>



LavA Projekt

<http://ess.cs.uni-dortmund.de/DE/Research/Projects/LavA/index.html>



Wishbone bei OpenCores

<http://opencores.org/opencores,wishbone>



I²C-Bus Spezifikation 3.0

http://www.nxp.com/acrobat_download/usermanuals/UM10204_3.pdf



Fragen?