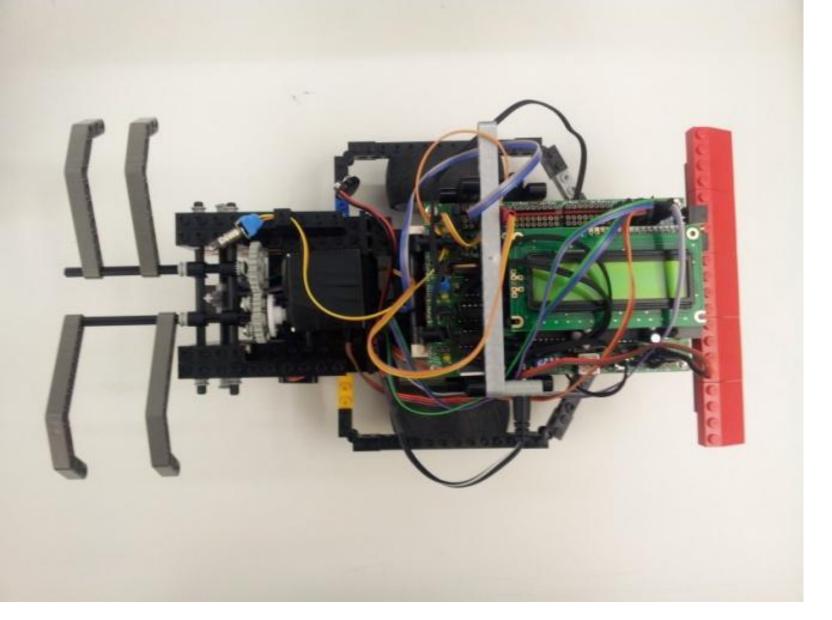


AMS-Kit

Bausatz zur Konstruktion Autonomer Mobiler Systeme

Dipl.-Inform. I. Boersch
FB Informatik und Medien





Patrick Pohlmann, Maximilian Orlowski, Heiko Ruhm: Projekt-Dokumentation ROBI, Wintersemester 2012/13





- Home
- Abschlussarbeiten



- AKSEN
- AMS-Projekte
- Kontakt
- LEGO League
- Maschinelles Lernen
- RobotBuildingLab
- Skripte
- Fotos
- Impressum

Aktuell

- LEGO League 2016
- Masterarheit von Reniamin Hoffmann

Willkommen im

Lehre und Forse

- Wissensve
- Soft Com
- Evolutionå
- KI-Prograi
- Semantisc

Anwendung wis

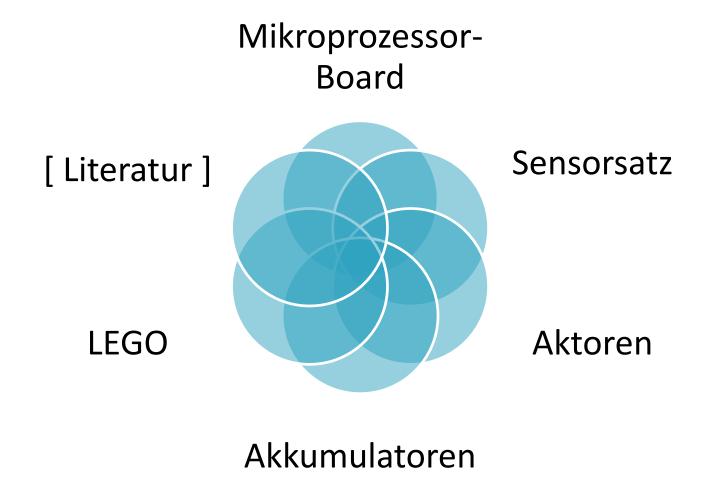
- Integrierte
- Planung ui

Studienrichtung '

LEGO League 2

Inhalt eines AMS-Kits





Mikroprozessorboard



Gehirn des Roboters

AKSEN = **AK**tor & **SEN**sor

Use as

- 1. Standalone controller for reactive robots
 OR
- 2. Sensor and actuator server in a RCUBE system

AKSEN-Board





I. Boersch, TH Brandenburg, 28.09.2022 AKSEN 6

AKSEN-Board



- Entwicklung der FH Brandenburg
 - Nach MIT 6.270-Board Rev. 2.21.
- 8-bit-Mikroprozessor C515 von Siemens
- serielle Schnittstelle
- 16 analoge und 16 digitale Sensorports
- diverse Elektronik zur Steuerung von Motoren, Servos, LED, LCD, ...

AKSEN board



- connections for simple sensors
- motor drivers, power drivers and control outputs
- enough performance for reactive robots
- program survives and reboots after a power blackout
- small power consumption
- small hardware and software costs
- solderable by user, distribution as module or kit
- native GPL C-compiler (Linux, Windows)
- Optional: bluetooth
- Possible: more than one AKSEN board

Sensorsatz



Augen und Ohren des Roboters

- digitale und analoge Sensoren
- leicht und billig
- Sensoren für

Licht:5 Fotosensoren

Berührung: 8 Mikroschalter

kurze Entfernungen: 5 Optoreflexkoppler

– lange Entfernungen: 5 Infrarotempfänger

Aktoren



Hände und Füße des Roboters

- wichtig:
 - geringer Stromverbrauch,
 - geringes Gewicht
- Motore und Getriebe
- Glühlämpchen, Infrarot-Sender
- besonders interessant: Servo-Motor

Akkumulatoren



- NiMH-Mignon Akkupacks
- 5 x 1.2 V, >2500 mAh
- 2 Stück pro Kit
- Ladestand kontrollieren!
- Abends aufladen!

LEGO



- mechanische Basis
- intuitiv nutzbar und erweiterbar
- schnelle Umsetzung von Ideen
- Sets:
 - Basiskasten 8062, Autobausatz 8880, Grundplatte 815
 - zweimal zusätzliche Zahnräder 9966 und 9965
 - zusätzliche Kreuzachsen 9971
 - Bausteine 9824, Platten 9973
- Bausteine aller Arten, Achsen, Zahnräder, Ketten, Riemenscheiben, Gelenke,
 Stoßdämpfer, bereifte Räder und vieles mehr

Alles erlaubt, jedoch keine irreversiblen Veränderungen

Neu: Polymorph



Nicht verwenden:

- Kleber
- Schmieröl
- [Isolierband]
- [Klebeband]

Literatur



- AKSEN-Handbuch rechts oben beim Desktop
 - Icon => AKSEN Handbuch.pdf
 - Liste aller Funktionen, Anschlüsse
- Regeln und Aufgabenstellung des aktuellen Projektes.
- Bibliothek:
 - "Mobile Roboter", Anita Flynn und Joseph Jones

AKSEN-Bibliothek



- Sammlung von C-Funktionen
- Im Flash des AKSEN-Boards
- Abfrage von analogen und digitalen **Sensoren** jeglicher Art
- Ansteuerung der Motoren mit PWM (also in verschiedenen Geschwindigkeiten)
- Ansteuerung Servo
- Senden und Empfangen modulierter Infrarotstrahlung
- Ausgabe von Texten auf die LCD-Anzeige
- Abfrage der Knöpfe, DIP-Schalter und Potentiometer der Platine
- Funktionsliste im Anhang des AKSEN-Handbuches

AKSEN@Home



Kältepause um Weihnachten?

- → Puzzle@Home
- Was ist im Paket?
- Einrichten der Arbeitsumgebung
- Praktikum

Ausleihe: Was ist im Paket?



- 1. 1 Kiste
- AKSEN-Board
- 3. Steckernetzteil 6V zur stationären Stromversorgung
- 4. 2 Akkupacks zur mobilen Stromversorgung
- 5. Kabel vom Akkupack zum AKSEN-Board
- 6. Ladegerät für zwei Akkupacks
- 7. USB-Seriell-Adapter + serielles Kabel (Telefonstecker)
- 8. Sensoren: 2 Licht, 4 Optokoppler, 5 Taster, 2-3 IR-Sender, 2-3 IR-Empfänger, 2-3 Lämpchen

Ausleihe nicht nötig

- 9. Aktoren: 3x2 LEGO-Motoren (8883,88003,8882), 6 Kabel, Servomotor
- 10. 1.3 kg LEGO-Steine eigener Wahl im FBI-Sack
- 11. USB-Kamera für die Webkonferenzen und Videos





- 1. 1 Kiste
- 2. AKSEN-Board
- 3. Steckernetzteil 6V zur stationären Stromversorgung
- 4. 2 Akkupacks zur mobilen
 - Stromversorgung
- 5. Kabel vom Akkupack zum AKSEN-Board
- 6. Ladegerät für zwei Akkupacks
- 7. USB-Seriell-Adapter + serielles Kabel (Telefonstecker)
- S. Sensoren: 3 Licht, 2 Optokoppler, 6 Taster, 3 IR-Sender, 4 IR-Empfänger, 3 Lämpchen
- 9. Aktoren: 3x2 LEGO-Motoren (8883,88003,8882), 6 Kabel,
- Servomotor

 10. 1.3 kg LEGO-Steine eigener Wahl im FBI-Sack
- USB-Kamera für die Webkonferenzen und Videos

Einrichten der Arbeitsumgebung



- 1. Das Handbuch
- 2. cygwin
- 3. Der Compiler sdcc
- 4. AKSENLib und Beispiele
- 5. USBSeriell-Adapter und Flasher

Einrichten der Arbeitsumgebung



Für unsere Arbeitsumgebung richten wir 4 Icons ein, am Besten rechts oben auf dem Desktop. So soll es am Ende aussehen:



1. Das Handbuch

- Entpacken Sie aus Moodle die Datei <u>AKSEN@Home Software.zip</u>
- Legen Sie die Datei Handbuch.pdf aus dem ZIP-File auf dem Desktop ab.



2. cygwin



Ziel: Cygwin in C:\cygwin64 installieren, extra-Paket make

Aus dem ZIP-File: cygwin\install.bat ausführen

Kurz und knapp:

- Alle Default-Einstellungen lassen, außer:
 - Deutschen Server wählen,
 - Package-Directory auf C:\cygwin64_localpackages,
 - "make" in Kategorie "Devel" auswählen

Es folgt die ausführliche Anleitung (kann übersprungen werden).



X



E Cygwin Setup

Cygwin Net Release Setup Program

This setup program is used for the initial installation of the Cygwin environment as well as all subsequent updates. The pages that follow will guide you through the installation.

Please note that we only install a base set of packages by default. Cygwin provides a large number of packages spanning a wide variety of purposes.

You can always run this program at any time in the future to add, remove, or upgrade packages as necessary.

Setup version 2.905 (64 bit)

Copyright 2000-2020

https://cygwin.com/

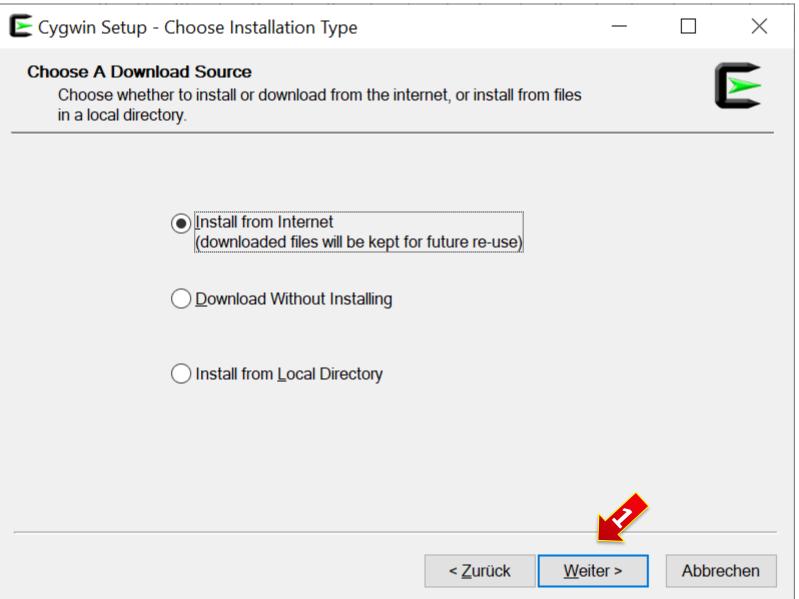


< Zurück

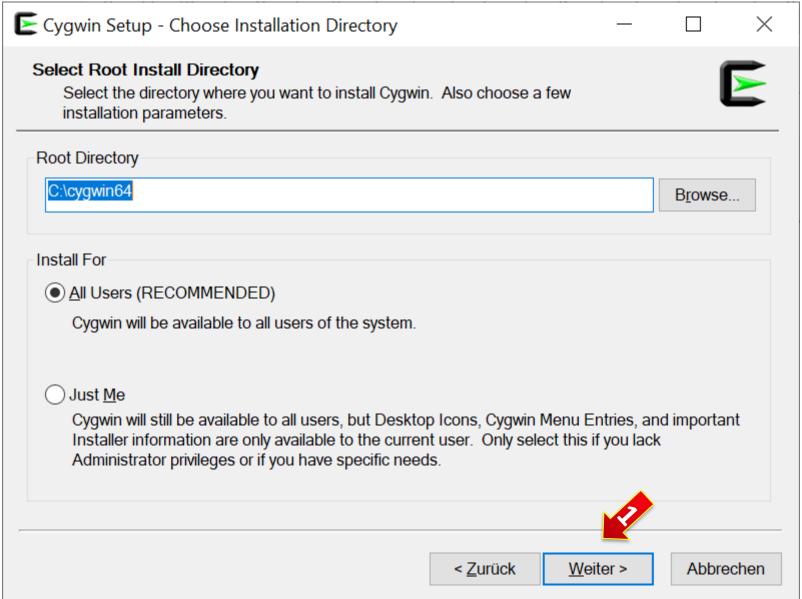
Weiter >

Abbrechen

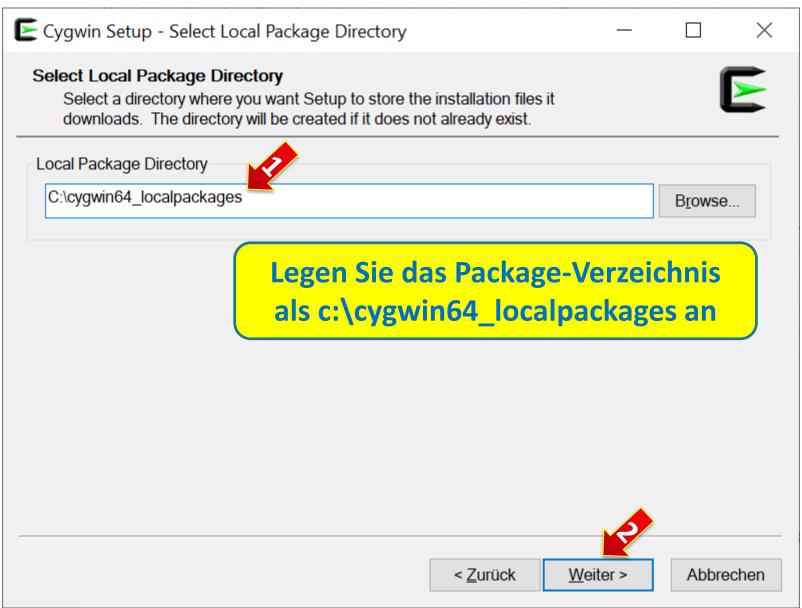




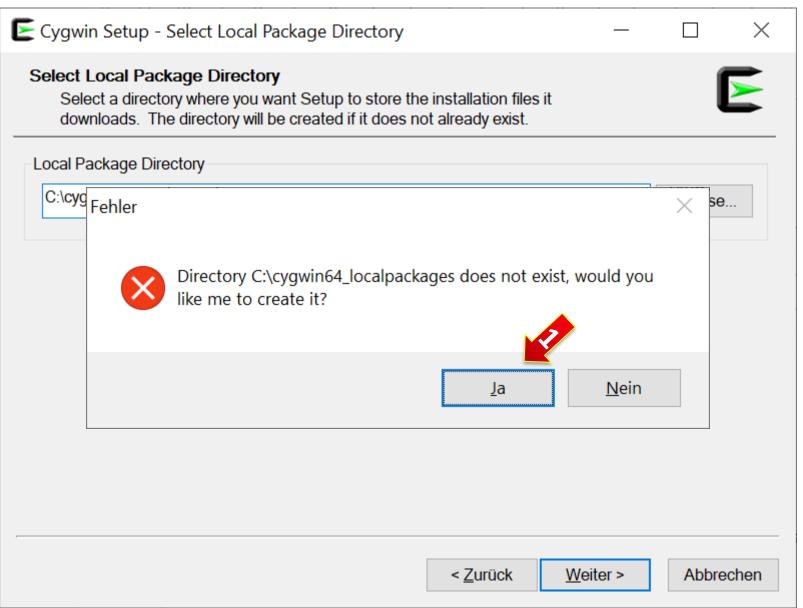




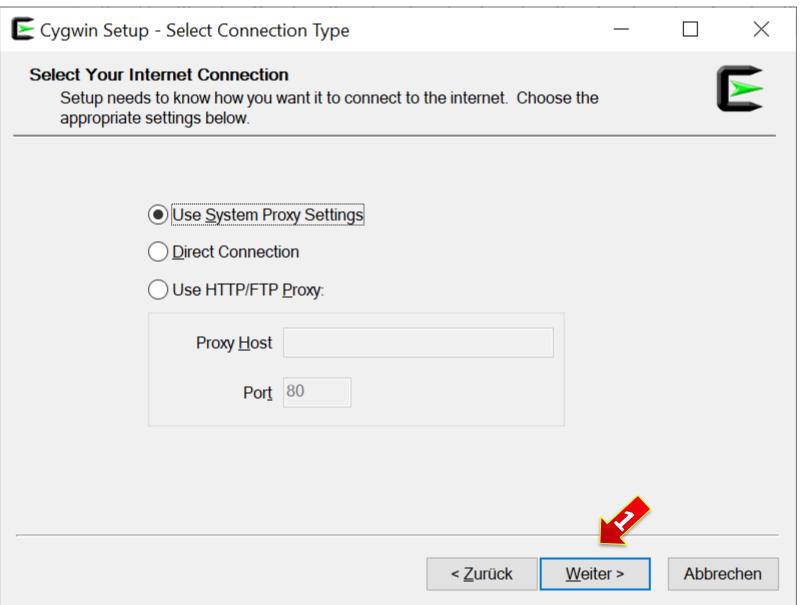


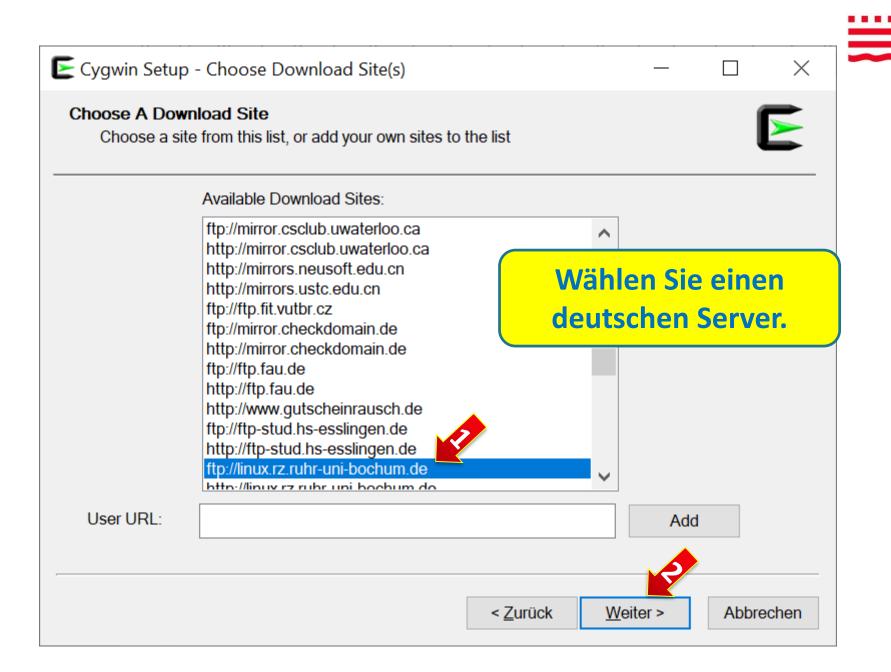




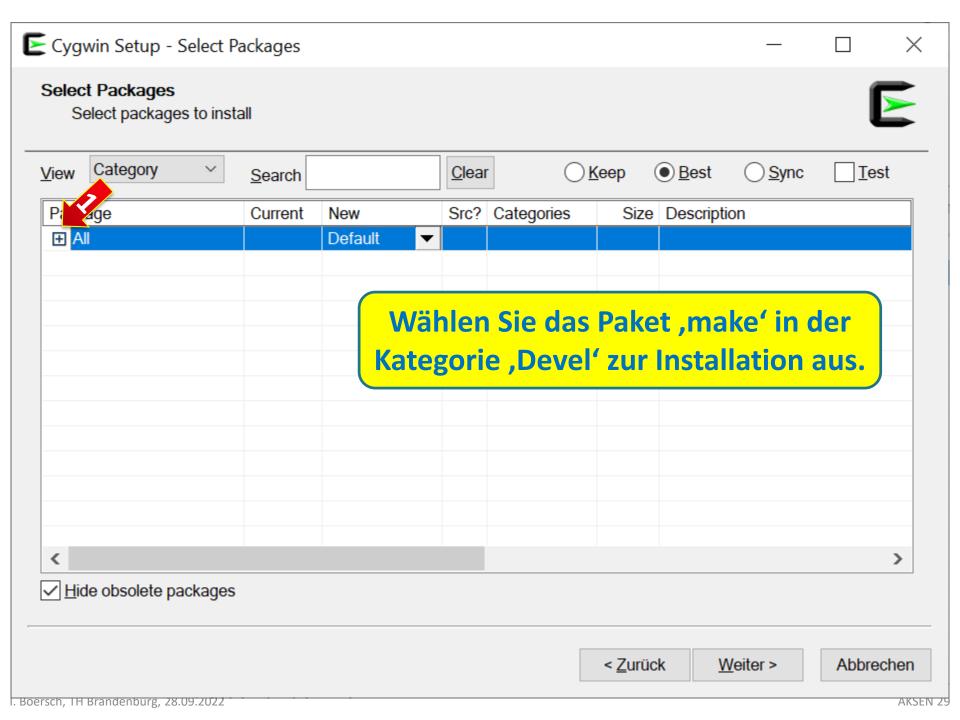


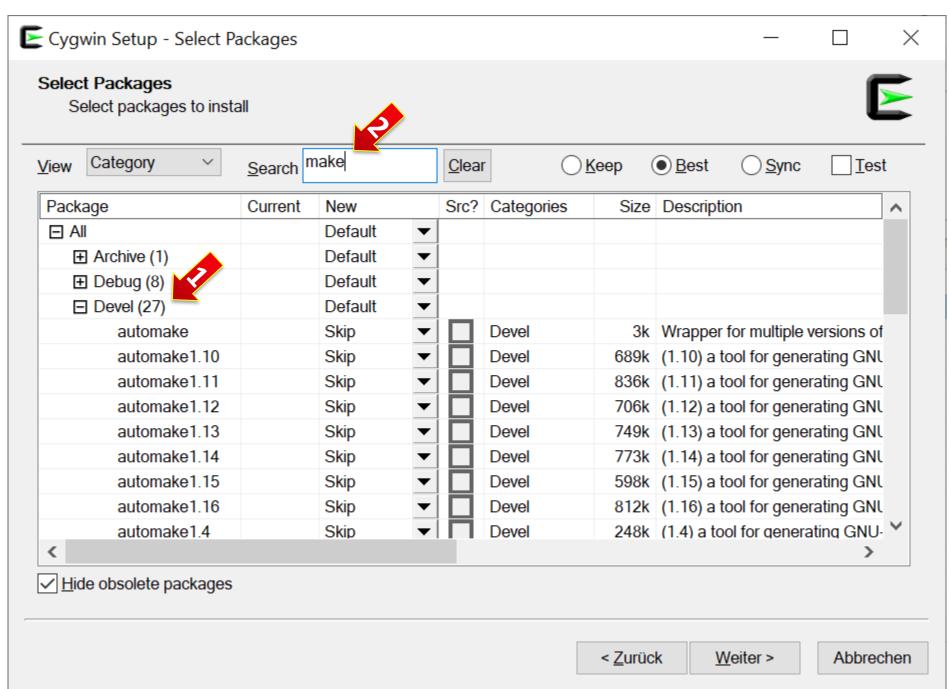






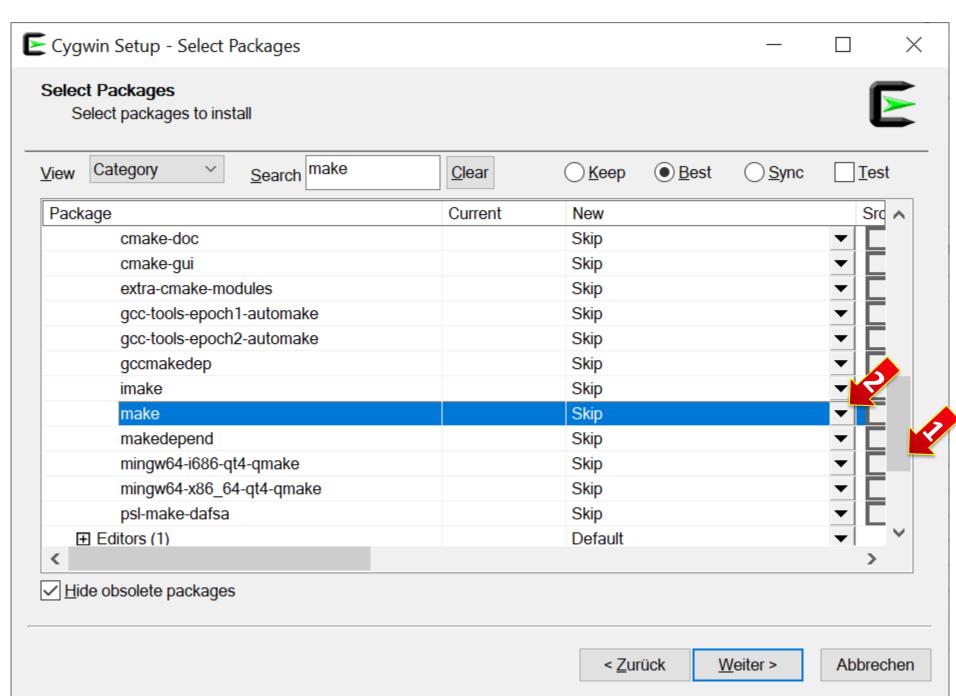


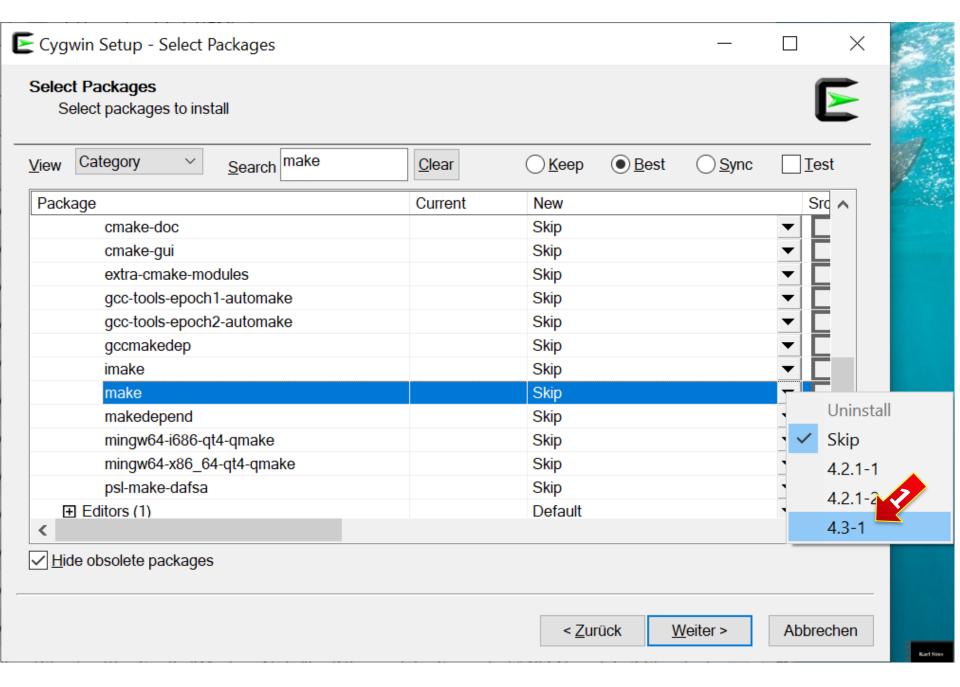


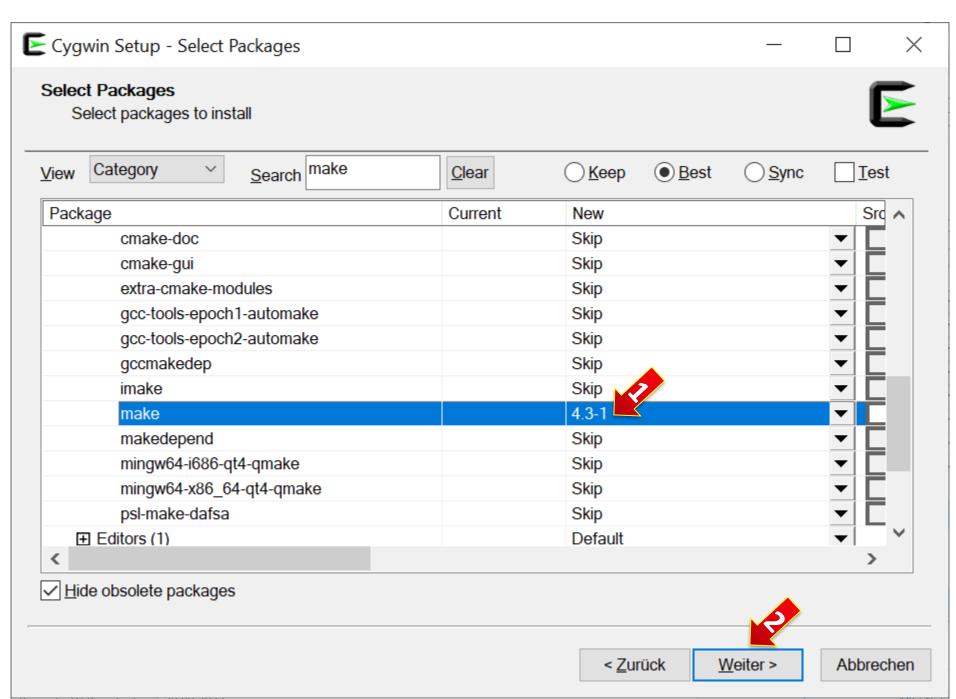


l. Boersch, TH Brandenburg, 28.09.2022

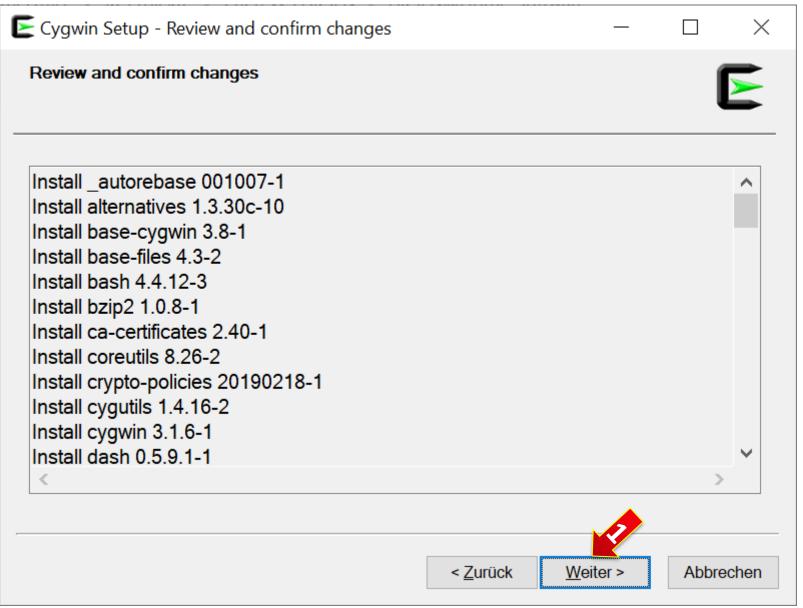
AKSEN 30



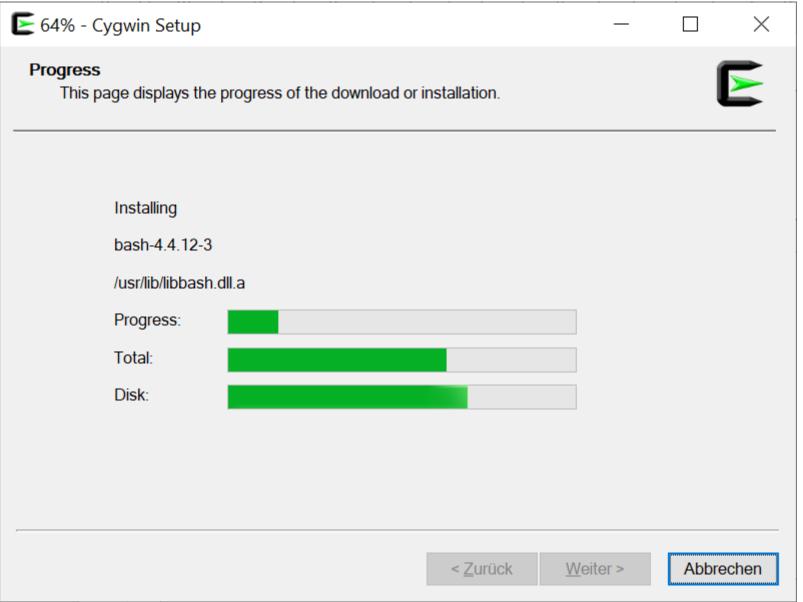




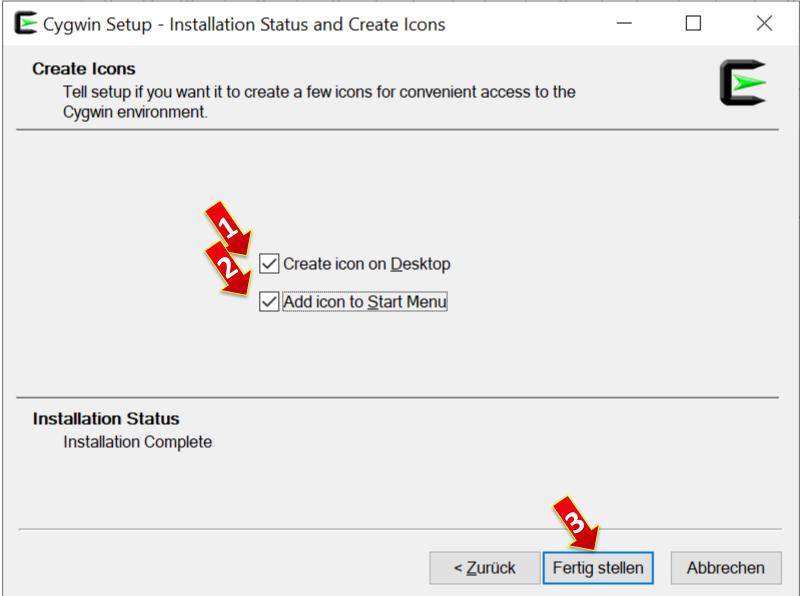








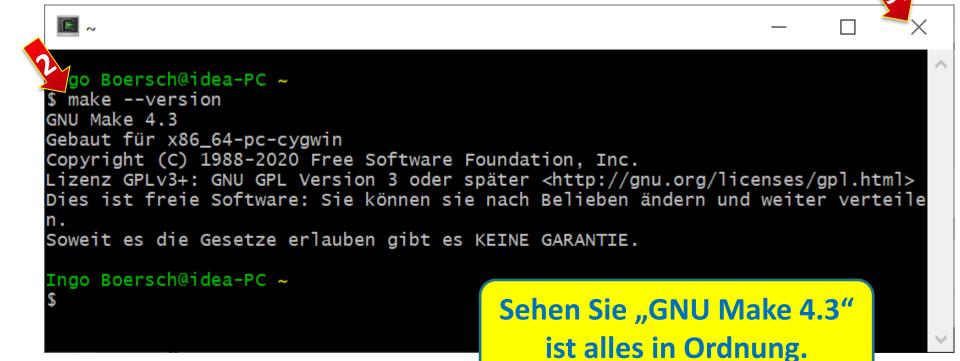






- Cygwin-Icon rechts oben platzieren
- Cygwin starten
- make testen mit: make --version









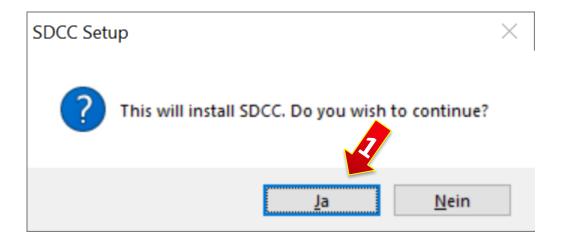
Ziel: sdcc\sdcc-2.7.0-setup.exe in C:\sdcc installieren

- Administratorrechte werden benötigt

Kurz und knapp:

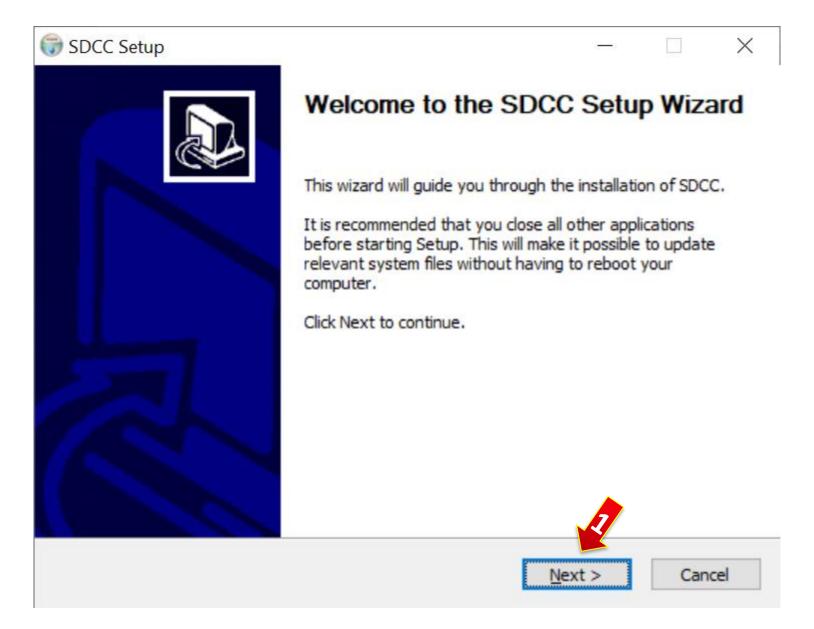
- Alle Default-Einstellungen lassen, außer:
 - Zielpfad C:\sdcc
 - C:\sdcc zum Pfad hinzufügen (damit cygwin den sdcc findet)

Es folgt die ausführliche Anleitung (kann übersprungen werden).

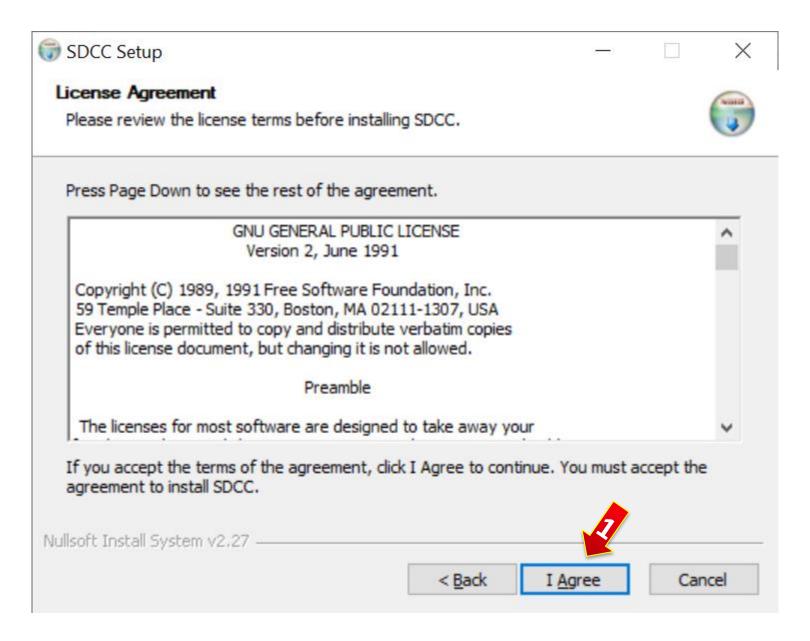


I. Boersch, TH Brandenburg, 28.09.2022

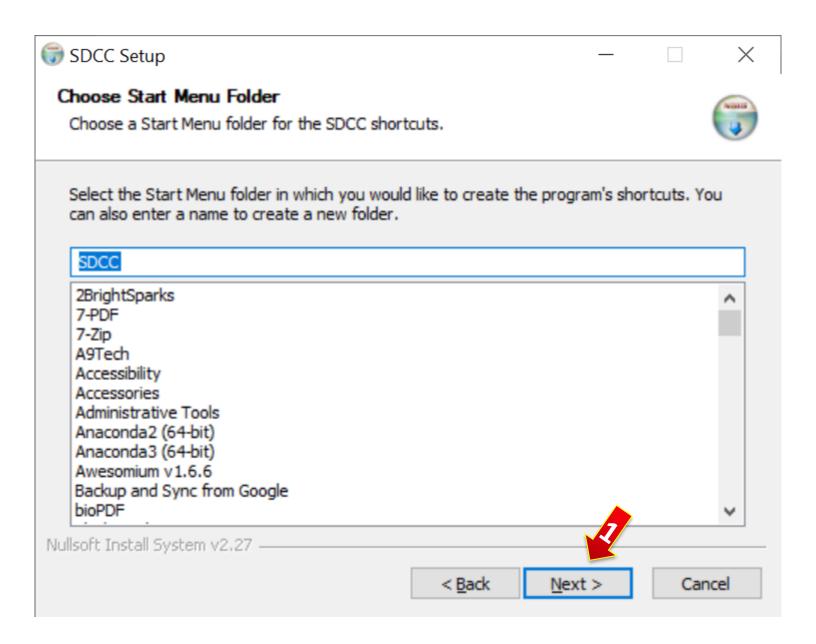




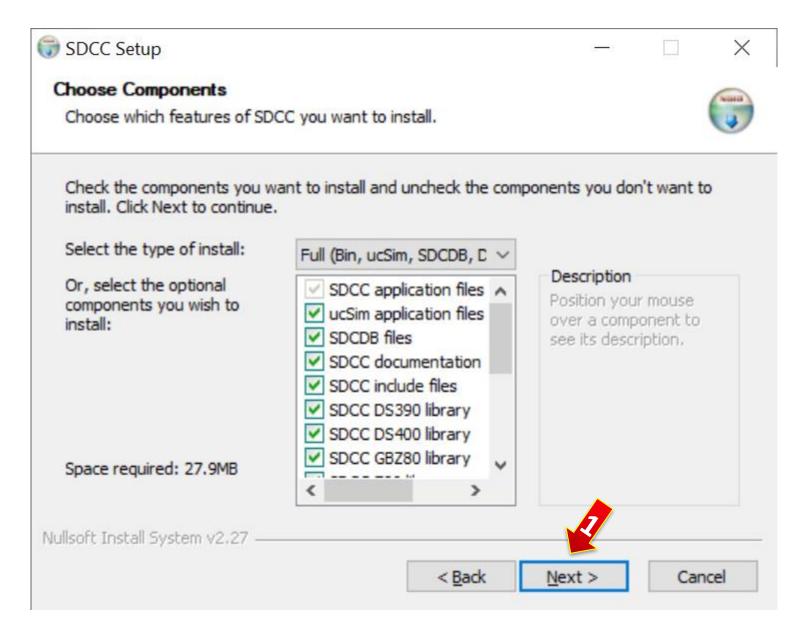




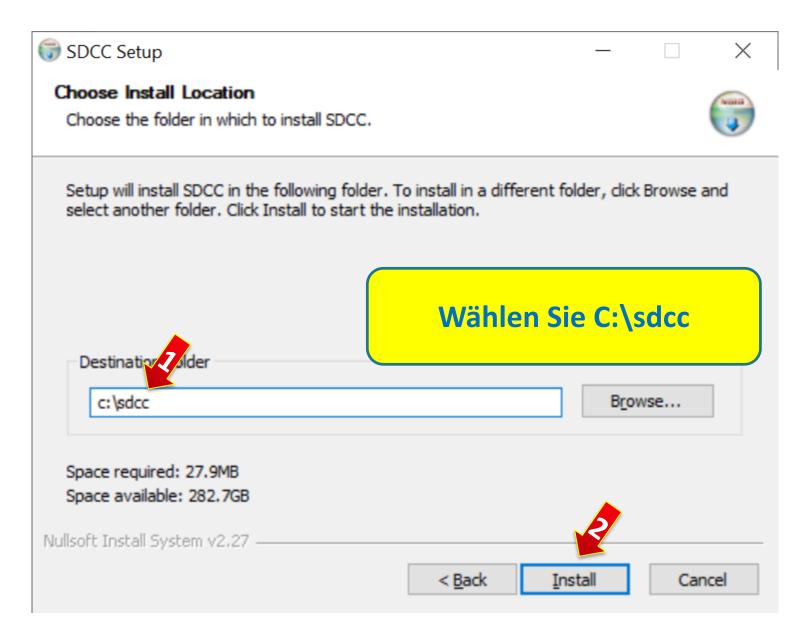




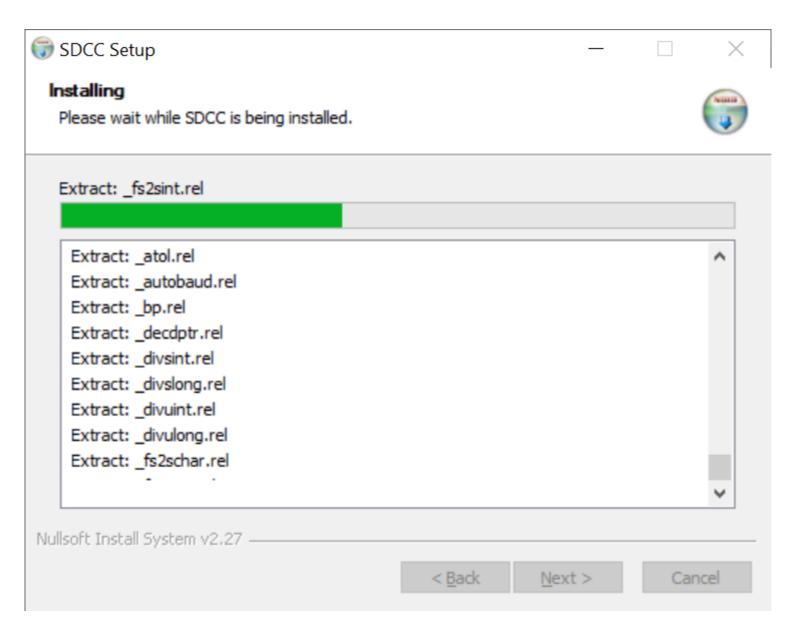












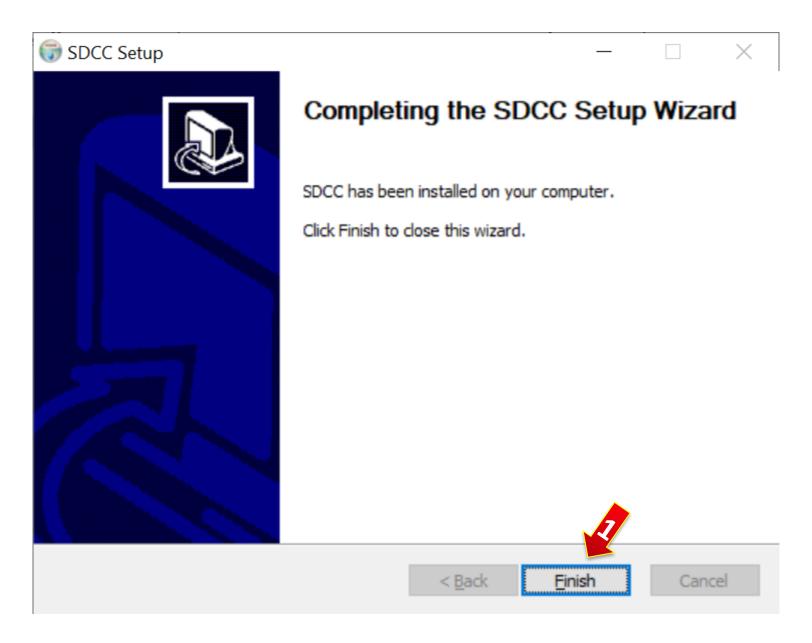


Wählen Sie "Ja".



I. Boersch, TH Brandenburg, 28.09.2022





Test SDCC



SDCC testen im neu gestarteten Cygwin-Terminal mit: sdcc --version



- Schlägt der Test fehl, ergänzen Sie den Pfad mit folgendem Befehl in Cygwin:

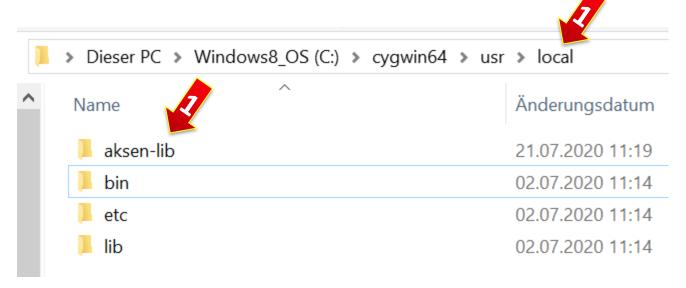
 echo "export PATH=\\$PATH:/cygdrive/c/sdcc/bin" >> \$HOME/.bashrc
- Starten Sie Cygwin neu, nun sollte der sdcc laufen.





Kopieren Sie das Verzeichnis "aksen-lib" aus dem Zip-File nach

C:\cygwin64\usr\local







Um die AKSEN-Lib zu **kompilieren**, wechseln Sie im Cygwin-Terminal mit "cd" ins Verzeichnis /usr/local/aksen-lib/large-stack-auto und starten dort "make":

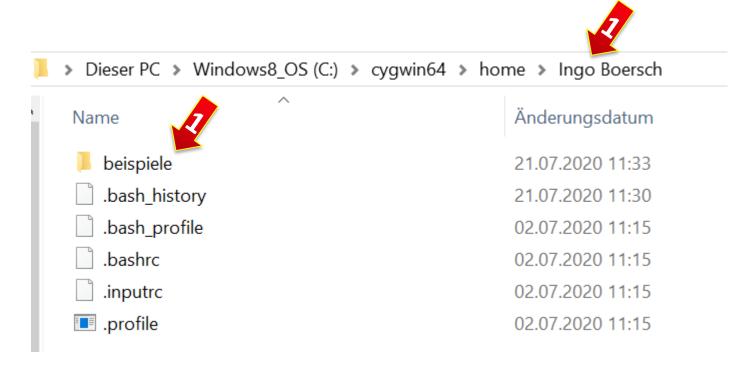


Die Erstellung läuft mit einigen Warnungen, aber ohne Fehler durch.





Kopieren Sie das Verzeichnis "beispiele" aus dem Zip-File nach C:\cygwin64\home\IhrName



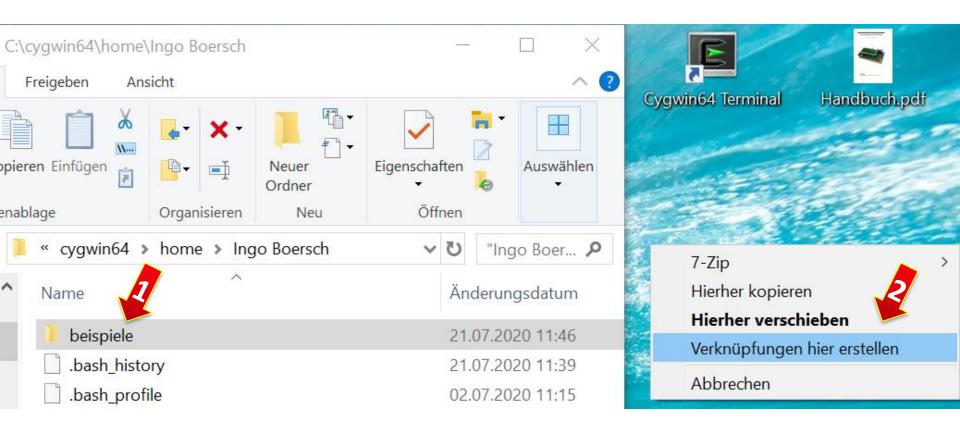
4. Beispiele in Cygwin kompilieren



```
~/beispiele
                                                                               X
Ingo Boersch@id🍾
$ cd beispiele
Ingo Bogras @idea-PC ~/beispiele
$ make
                                    Es werden alle 37
    ~/beispiele
                                   Beispiele kompiliert.
   cp "/usr/local/aksen-lib/stub/ak
   cp /usr/local/aksen-lib/stub/AkSen.is
   sdcc -V --model-large --data-loc 0x18 --stack-auto -Llarge AkSen.rel werbung.re
   + C:\SDCC\bin\aslink.exe -nf "AkSen.lnk"
   mv AkSen.ihx werbung.ihx
   rm *.lst *.sym *.rel *.asm -f
   rm *.rst *.map *.mem *.lnk -f
   ls -l werbung.ihx
   -rwxr-xr-x 1 Ingo Boersch Ingo Boersch 8811 21. Jul 11:42 werbung.ihx
   make[1]: Verzeichnis "/home/Ingo Boersch/beispiele/werbung" wird verlassen
   Ingo Boersch@idea-PC ~/beispiele
```



4. Desktop-Verknüpfung für Beispiele



I. Boersch, TH Brandenburg, 28.09.2022

Zwischenstand



Wir haben bisher drei Icons angelegt:



5. USB-Seriell-Adapter und Flasher



Das Kompilieren erzeugt den Maschinencode für den Mikrocontroller: das IHX-File. Dieses File laden wir über den USB-Seriell-Adapter auf das AKSEN-Board. Dazu müssen wir herausfinden, auf welchem COM-Port der Prolific USB-Seriell-Adapter wartet:

1. USB-Seriell-Adapter an einen USB-Port anschließen.

2. Geräte-Manager → Anschlüsse (COM & LPT) → Prolific USB-to-Serial

Port (COM5)*

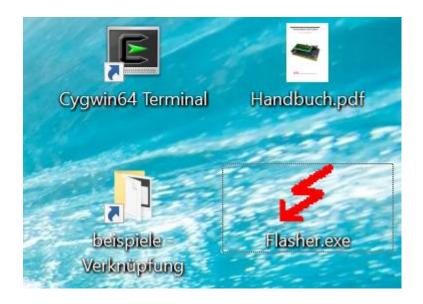
Hier also **COM5**, es kann auch ein anderer Port sein. **Port merken!**

* Bei Problemen: "Nach geänderter Hardware suchen" oder einen Prolific Treiber für PL2303 > 3.8.37.0 finden





Legen Sie den Flasher aus dem ZIP-File auf dem Desktop ab:



Die Einrichtung ist fertig.



Praktikum

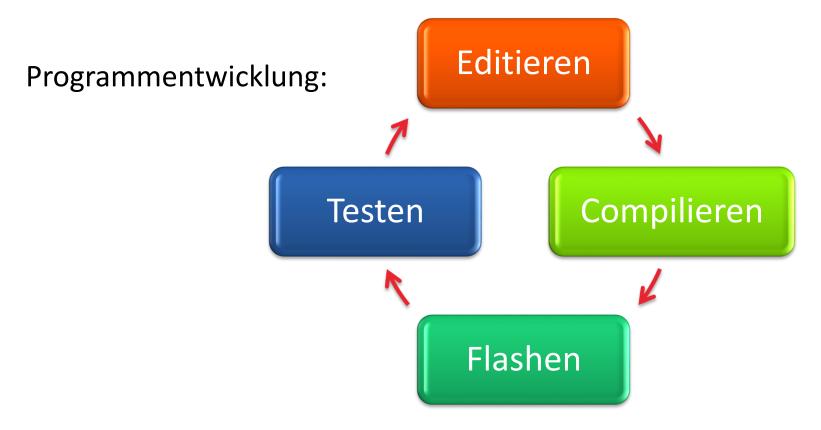
Programmierung, Sensoren, Aktoren

Vorbereitung



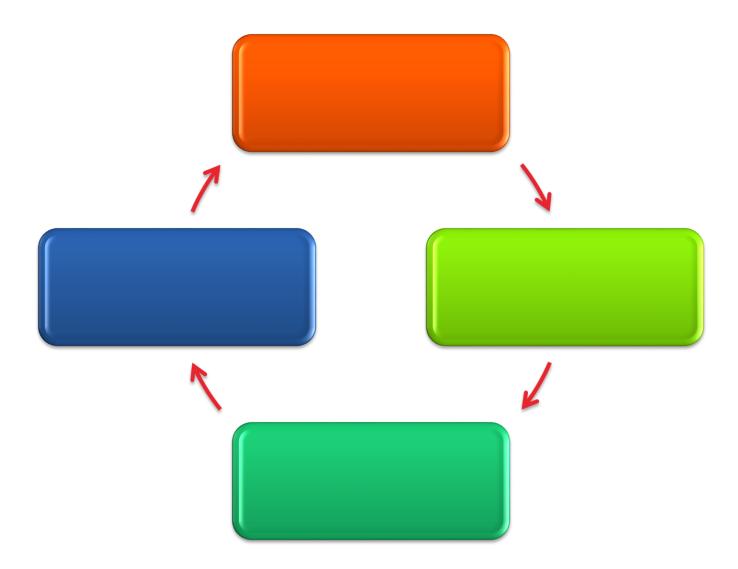
Anschließen der Stromversorgung 6V an die Platine Verbinden des USB-seriell-Adapters mit dem seriellen Kabel

→ Anschließen dieses Kabels an den PC und Platine



Programmentwicklung

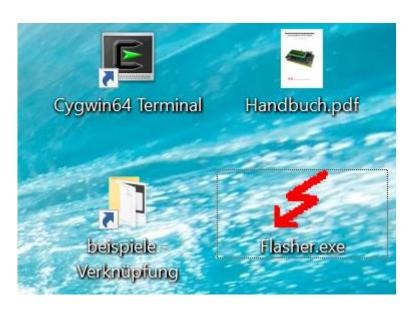




1. Editieren



Für jedes Projekt wird ein eigenes Verzeichnis angelegt!



- Öffnen der Verknüpfung "beispiele"
- Neues Projektverzeichnis durch Kopieren des Ordners "helloworld" nach "tollerName"
- Darin helloworld.c öffnen

1. Editieren



```
⊡ //Autor: K.-U. Mrkor
 //Standard-Include-Files

    Ersetzen Sie die

 #include <stdio.h>
 #include <regc515c.h>
                                                     Zeichenkette
                                                     durch eine
 //Diese Include-Datei macht alle Funktionen der
 //AkSen-Bibliothek bekannt.
                                                     eigene.
 //Unbedingt einbinden!
 #include <stub.h>

    Speichern

 //Hauptprogrammroutine
□ void AksenMain(void)
 {
     //Der erste Satz
     lcd puts("Hallo Welt");
     // Die folgende Endlosschleife "erzeugt" das Programmende
     while(1);
```

2. Compilieren



- Start Cygwin
- Linux-Shell unter Windows
 - \$ cd beispiele
 - \$ cd tollerName
 - \$ make
 - \$ ls
- => helloworld.ihx wird erzeugt (Maschinencode)

3. Flashen



- Start Flasher
- Datei helloworld.ihx wählen
- COM-Port wählen (hatten wir vorhin bestimmt)
- Board in den Download-Modus:
 - Schalter auf rote LED
 - Reset
- Flashen
- Board in den Programm-Modus
 - Schalter auf grüne LED
 - Reset

Motor starten



Anschließen eines Motors an Motor-Port 3

```
⊡//Diese Include-Datei macht alle Funktionen der
 //AkSen-Bibliothek bekannt.
 //Unbedingt einbinden!
 #include <stub.h>
 //Hauptprogrammroutine
□ void AksenMain(void)
 {
     motor_richtung(3, 0);
     motor pwm(3, 10);
     // Die folgende Endlosschleife "erzeugt" das Programmende
     while(1);
```

Blinker 1



Anschließen eines Lämpchens an LED-Port 2

```
void AksenMain(void) {
    led(2,1);
    while(1);
}
```

Lampe leuchtet? Dann Ihre Aufgabe:

Programmieren Sie ein Blinken einmal pro Sekunde (1 Hertz).

sleep(zeit_in_millsekunden);

Blinker 2



```
#include <stub.h>
void AksenMain(void)
        char schalter;
        schalter = 1;
        while (1) {
          if (schalter == 1) led(2,1);
          else led(2,0);
          schalter = schalter * -1;
          sleep (500);
```

- Variable
- Schleife

• einfacher?





```
#include <stub.h>

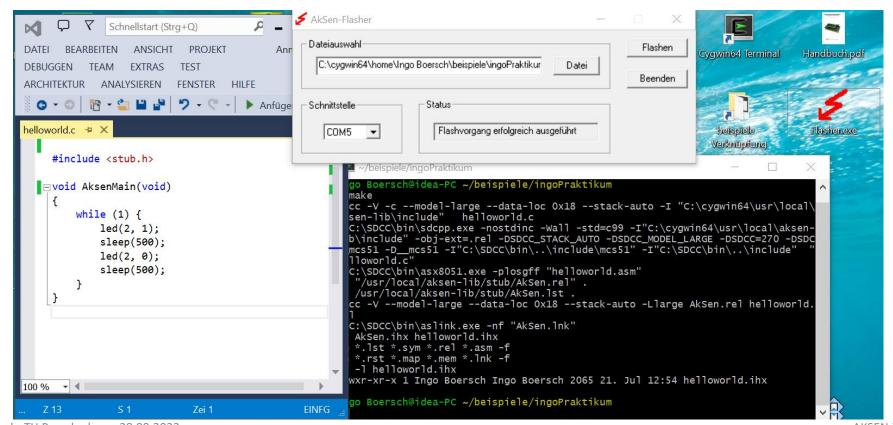
void AksenMain(void)

{
    while (1) {
        led(2, 1);
        sleep(500);
        led(2, 0);
        sleep(500);
    }
}
```

Tipps



- In Cygwin:
 - Autovervollständigen mit Tabulatortaste
 - Letze Befehle mit Cursor-Up-Taste
- Lassen Sie die drei Fenster von Editor, Cygwin und Flasher geöffnet...



Sensoren: Taster



Taster an Digital-Port O anschließen

```
#include <stub.h>

void AksenMain(void) {
    while(1) {
        lcd_cls();
        lcd_puts("Wert: ");
        lcd_ubyte(digital_in(0));
        sleep(200);
    }
}
```

Welcher Wert wird bei gedrücktem Taster angezeigt?





Fotosensor an Analog-Port 8

```
#include <stub.h>
void AksenMain(void){
    while(1) {
        lcd_cls();
        lcd_puts("Wert: ");
        lcd_ubyte(analog(8));
        sleep(200);
    }
}
```

Anzeige funktioniert? Dann Ihre Aufgabe:

Bei wenig Licht (>100): Lampe anschalten





```
#include <stub.h>
void AksenMain(void)
{
        char schalter:
        while (1) {
          if (analog(8) > 100) {
                 led(2,1);
          else {
                 led(2,0);
```

- Automatisches Programm
- EVA

- ⇒ Starten des Roboters
- \Rightarrow einfacher?

Sensoren: Infrarotes Licht

Unsichtbare Strahlen

Programmieren Sie:

- Infrarot-Sender an LED-Ausgang 3
- Infrarot-Empfänger an Analog-Eingang 0
- Anzeige des Empfänger-Wertes
- Guter Empfang: Werte kleiner 10
- IR-Sender können verpolt werden, dann leuchten sie nicht.



Stimmt das: Handykameras können Infrarot aufnehmen?

Infrarot: Anzeige



Optokoppler



- Gleiches Programm
- Optokoppler anschließen
- Auf schwarzer und weißer Fläche testen
- ⇒ Linienfolgen!

Aufgabe gelöst?



Kriterium der Wahrheit ist die Praxis Aufgabe gelöst, wenn:

Funktion

- funktioniert?
- funktioniert immer?

Einfachheit

- überflüssiger Code?
- geht es einfacher?

Verständlichkeit

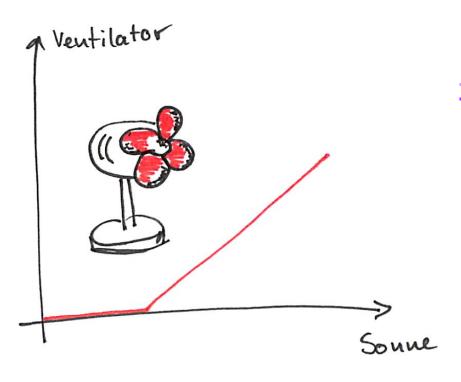
- strukturiert?
- kommentiert?

Fragen?

Basisaufgabe: schwer



- Es ist heiß, die Sonne brennt. Ein Ventilator wäre gut.
- Bei wenig Sonne ist der Ventilator aus.
- Je mehr die Sonne scheint, desto schneller dreht sich der Ventilator.



if (AufgabeGelöst)
 FertigFürHeute()

