



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi
FACHBEREICH INFORMATIK

RECHNERARCHITEKTUR

WS2023/24

Termin1

Laborordnung; Zahlensysteme; Einführung in Konzepte der Informatik; Einführung in eine Entwicklungsumgebung

Name, Vorname	Matrikelnummer	Anmerkungen
Datum	Raster (z.B. Mi3x)	Testat/Datum

Legende: V:Vorbereitung, D: Durchführung, P: Protokoll/Dokumentation, T: Testat

Hinweise zur Laborordnung für D10/0.32

Sinn der Laborordnung ist die Festlegung von Regeln für die Benutzung der Labore in D10.

Jede/r ordentliche Studierende des Fachbereich ist in den jeweils ersten Veranstaltungen auf die Laborordnung hinzuweisen.

Die Laborordnung und weitere wichtige Informationen hängen beim Laboreingang.

1. Der Notausschalter des Labors D10/00.32 befindet sich über dem Lichtschalter. Der Notausschalter ist im Notfall zu betätigen, um sämtliche elektrische Geräte stromlos zu schalten. Achten Sie bei der Einführung auf die entsprechenden Hinweise.
2. Es ist nicht gestattet sich alleine im Labor aufzuhalten.
3. Im Labor darf maximal Paarweise an den Laborplätzen gearbeitet werden. Das Labor ist mit 8 Arbeitsplätzen ausgelegt.
4. Die Ersthelfer für das Gebäude D10 sind Herr Rudi Scheitler (Raum 00.36 / Tel.: 68465), Herr Sergio Vergata (Raum 00.37 / Tel.: 68491), Frau Bettina Kurz (Raum 00.36 / Tel.: 68453) und Herr Manfred Pester (Raum 00.33 / Tel.: 68428). Wenden Sie sich bitte im Falle einer Verletzung an einen der Ersthelfer.
5. Es ist nicht gestattet Kabel zu entfernen, Gehäuse zu öffnen und Hardware zu installieren oder sonstige Änderungen an der Laborinfrastruktur vorzunehmen. Sollte etwas nicht funktionieren, oder es wird etwas benötigt, welches die vorhandene Infrastruktur nicht abdeckt, so wenden Sie sich an den Betreuer des Labors oder direkt an den zuständigen Laboringenieur Manfred Pester (D10/00.33 / Tel.: 68428).
6. Fahren Sie die von Ihnen benutzten Geräte am Ende Ihres Praktikum/Ihrer Übung herunter und schalten diese aus, es sei denn Sie bekommen vom zuständigen Betreuer andere Anweisungen.
7. Speisen und Getränke sind an den Laborarbeitsplätzen nicht gestattet.
8. Bei der Benutzung des Labordrucker (steht in D10/00.35) ist Sorgfalt und Sparsamkeit oberstes Gebot.
9. Evtl. ausgestellte Dokumentationen dienen der Laborarbeit und müssen im Raum verbleiben.
10. Die Benutzung von Mobiltelefonen ist untersagt. Schalten Sie vor dem Betreten des Raumes die Geräte (nicht nur lautlos) ab. In dringenden Fällen können Sie sich über das Labortelefon mit der Nummer 06151 53368433 anrufen lassen.
11. Hängen Sie Ihre Kleidung (Mäntel, Jacken, ..) an die dafür vorgesehenen Kleiderständer und nicht über die Stühle.
12. Deponieren Sie Taschen, Laptops u.s.w. nicht in den Gängen, sondern möglichst an den Seiten des Labors oder unter den Tischen.
13. Verlassen Sie Ihren Arbeitsplatz aufgeräumt! Müll gehört in die vorhandenen Mülleimer, Altpapier in die dafür vorgesehene blaue Altpapierwanne.
14. Die Fluchtwege sind frei zu halten.

Bei Verstößen gegen die Laborordnung kann die Benutzungsberechtigung versagt werden.

Ziel der folgenden Aufgaben:

Die ersten Aufgaben sollen Ihnen eine Einführung in einige Konzepte der Informatik mit historischen Hilfsmitteln geben. Sie bekommen hierzu ein Aufgabenblatt. Mit den zur Verfügung gestellten historischen Rechenwerkzeugen, sollten Sie, mit **der richtigen Vorbereitung**, in der Lage sein alle Aufgaben zu lösen.

Auch ohne Hilfsmittel sollten wir in der Lage sein, Werte zwischen verschiedenen Zahlensystemen zu wandeln. Hierzu einige Übungen.

1. Wandlung zwischen Zahlensystemen

Beispiel: Divisionsverfahren zur Umwandlung von Dezimalzahlen in ein anderes Zahlensystem (b = Basis des Zahlensystems z.B. 2)

Division	Rest	binäre Stelle
71 : b = 35	1	$1 * 2^0$
35 : b = 17	1	$1 * 2^1$
17 : b = 8	1	$1 * 2^2$
8 : b = 4	0	$0 * 2^3$
4 : b = b	0	$0 * 2^4$
b : b = 1	0	$0 * 2^5$
1 : b = 0	1	$1 * 2^6$

am Beispiel 71 von dezimal \rightarrow b = 2
Das Ergebnis ist $1 * 2^6 + 0 * 2^5 + 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0$ also 1000111

Übung: Umwandlung von Zahlen in andere Zahlensysteme **ohne** Taschenrechner

1. Dezimal \rightarrow binär
Wandeln sie die Zahl 90_{10} in eine binäre Zahl mit Hilfe des Divisionsverfahrens
2. Dezimal \rightarrow hexadezimal
Wandeln sie die Zahl 165_{10} in eine hexadezimale Zahl mit Hilfe des Divisionsverfahrens
3. Hexadezimal \rightarrow dezimal
Wandeln sie die Zahl $FF0_{16}$ in eine dezimale Zahl mit Hilfe der Exponentialdarstellung

4. Hexadezimal → binär
Wandeln sie die Zahl $A5A_{16}$ in eine Binäre Zahl

5. Binär → Hexadezimal
Wandeln sie die Binärzahl 0101101010100101_2 in eine Hexadezimalzahl

6. Binär nach dezimal
Wandeln sie die Binärzahl $0100\ 0000\ 0000_2$ in eine Dezimalzahl

Für die folgenden Aufgaben 2. bis 5. gibt es eine Sammlung „Termin1_Brief.pdf“ mit vielen zugehörigen Aufgaben. Lösen Sie die Aufgaben nur mit den dafür vorgesehenen Geräten. Überprüfen Sie Ihre Ergebnisse ohne weitere zusätzliche Hilfsmittel. Stifte und Papier sind erlaubt.

2. Rechnen mit einer Walther Rechenmaschine

Mithilfe einer Walther Rechenmaschine sollen sie eine Addition, eine Subtraktion und eine Multiplikation durchführen. Sie lernen dabei das Konzept des Akkumulators kennen und lernen wie man eine Multiplikation auf mehrere Additionen zurückführen kann.

Sehen sie sich als Vorbereitung auch an, wie eine schriftliche Multiplikation funktioniert.

Link:

https://www.didaktik.mathematik.uni-wuerzburg.de/history/ausstell/sammlung/sprossenrad/sprossenrad_interaktiv.html

3. Rechnen mit dem Abakus

Mithilfe eines Abakus werden sie eine Addition und eine Subtraktion durchführen. Sie studieren dabei das Konzept des Übertrags, das für alle arithmetischen Rechnungen von hoher Bedeutung ist.

Sehen sie sich als Vorbereitung die schriftliche Addition und die schriftliche Subtraktion an und überlegen sie sich, wie diese im binären Zahlensystem aussehen könnten. Suchen sie sich eine Anleitung für das Rechnen mit einem Abakus

Link:

<https://www.mathematik.uni-marburg.de/~thormae/lectures/ti1/code/abacus/sanpan.html>

4. Der Rechenschieber

Mithilfe eines Rechenschiebers führen sie eine Multiplikation und eine Division aus. Diese Aufgabe soll sie in das Rechnen mit Zahlen in Exponentialdarstellung, d. h. Zahlen die eine Mantisse und einen Exponenten besitzen, einführen. Dies ist wichtig zum Verständnis von Gleitpunktzahlen in der Informatik.

Sehen sie sich als Vorbereitung an, wie man mit einem Rechenschieber rechnet. Anleitungen findet man im Internet. Verstehen sie, wie man eine Zahl in Mantisse und Exponent aufspaltet. Verstehen sie die Funktion des Logarithmus und warum damit eine Multiplikation auf eine Addition zurückgeführt werden kann.

Im Unterordner Rechenschieber finden sich verschiedene Rechenschieber-Simulatoren. Durch Öffnen der zugehörigen html-Dateien werden diese im Web-Browser geöffnet.

Oder versuchen Sie den Link: <https://www.arndt-bruenner.de/mathe/scripts/aristoscholar0903.htm>

5. Der Taschenrechner mit UPN

Die umgekehrte polnische Notation (UPN) stellt ein wichtiges Konzept in der Informatik dar. Jeder Compiler zerlegt eine komplexe Rechnung in der Form, wie Sie es in der UPN anwenden. Sie sollen in dieser Aufgabe einen ersten Kontakt mit dem Konzept eines Stack und seinen Operationen push und pop bekommen.

Vorbereitung: Die Darstellung eines komplexen algebraischen Ausdrucks mit der UPN. Das Konzept eines Stacks und die Operationen push und pop.

Im Unterordner HP-35C findet sich eine UPN-Taschenrechner-Simulation. Durch Öffnen der html-Datei kann diese mit einem Web-Browser genutzt werden. Oder versuchen Sie den Link: <https://www.hpmuseum.org/simulate/hp35sim/hp35sim.htm>

6. Einführung in eine Entwicklungsumgebung

Ab dem Termin 3 sollen Aufgaben gelöst werden bei denen auch programmiert werden soll. Es sollen Programme in Assembler entwickelt, geschrieben, übersetzt, gebunden und getestet werden.

Nachdem wir eine Lösung auf Papier haben, muss diese eingegeben und getestet werden. Hierzu gehen wir im Labor D10/0.32 Mikroprozessor/Mikrocontrollerlabor wie folgt vor:

Schalten Sie einen der Laborrechner MI-2014-2 bis MI-2014-9 ein.

Wählen sie das Linux-Betriebssystem (wird - wenn Sie warten auch automatisch gestartet) aus.

Der gestartete Rechner bietet Ihnen einen Anmeldebildschirm an.

Melden Sie sich mit Ihrem vom der h_da erhaltenen Namen (stxyabcd) an.

„st“ steht für Student,

„xy“ steht für die ersten Buchstaben ihres Vornamens und

„abcd“ steht für die ersten Buchstaben Ihres Nachnamens.

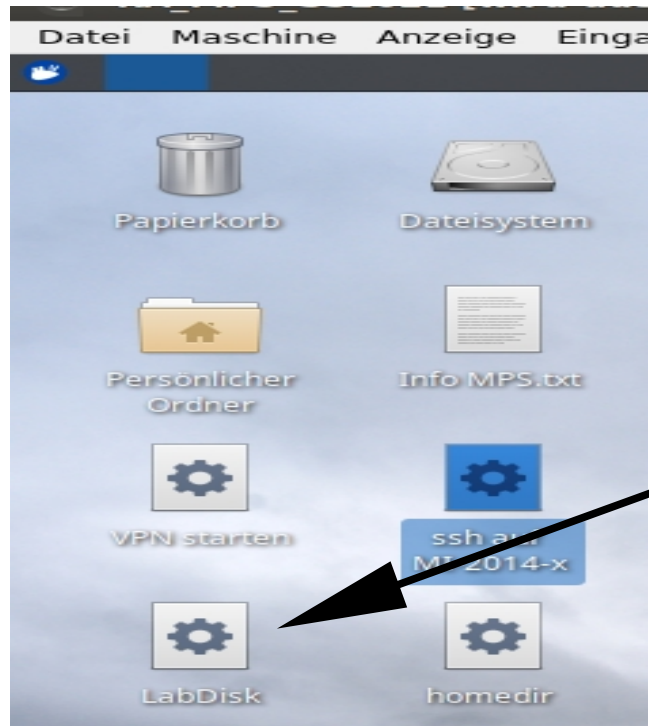
Wählen Sie vor der Bestätigung Ihres Passwort über den Schalter unten rechts in der Ecke die zu nutzende Oberfläche Plasma.

Oder auf eigenem Rechner:

Installieren und starten Sie VirtualBox mit z.B. dem Image „RA_MPS_TGI_WS2023/24.ova“. Zu dem Image gelangen Sie auch über die Seite des Mikroprozessorlabor (<https://fbi.h-da.de/studium/labore/mikroprozessor/>).

Das Passwort für Nutzer fs im zur Verfügung gestellten Image RA_MPS_TGI_WS2023/24.ova ist „fs“.

Holen Sie sich zuerst die aktuellen Informationen und Aufgabenstellungen. Beachten Sie die Informationen in Ihrer zugehörigen Vorlesung.



Möglichkeiten der Aktualisierung der Aufgaben:

- Schauen Sie in Moodle
- Schauen Sie im Internet auf den Seiten des verantwortlichen Professor/Lehrenden.
- Denken Sie an die Informationen, welche Sie in der zugehörigen Vorlesung erhalten haben.
- ..

In der angebotenen Virtuellen Umgebung RA_MPS_TGI_WS2023/24 können wir uns direkt mit `../LabDisk/MI` verbinden, um dann den Ordner „raWS2023/24“ in einen lokalen Ordner zu kopieren.

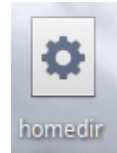
Es folgt ein vorbereitendes Beispiel im Umgang mit der Entwicklungsumgebung.

1. Wechseln Sie in Ihren gewählten Arbeitsordner.

Info:

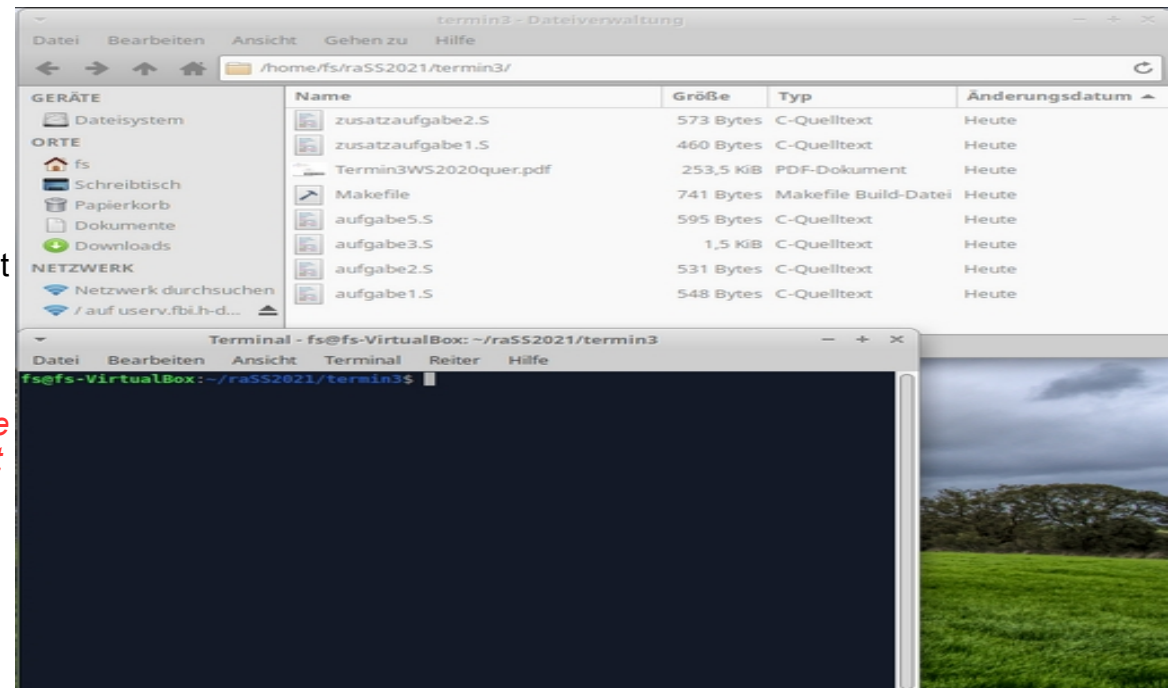
Ein Ihnen auf `userv-shell.fbi.h-da.de` zur Verfügung gestelltes Heimatverzeichnis `~/home/stxyabcd` ist wie folgt zu erreichen:

„[sftp://stxyabcd@userv-shell.fbi.h-da.de/home/stxyabcd](ftp://stxyabcd@userv-shell.fbi.h-da.de/home/stxyabcd)“ oder verwenden Sie das zur Verfügung gestellte Icon
Hier können Sie Ihre Dateien auch ablegen. Das Arbeiten auf dem Netzwerklaufwerk ist sicherlich nicht so performant wie ein Arbeiten auf einem lokalen Laufwerk.

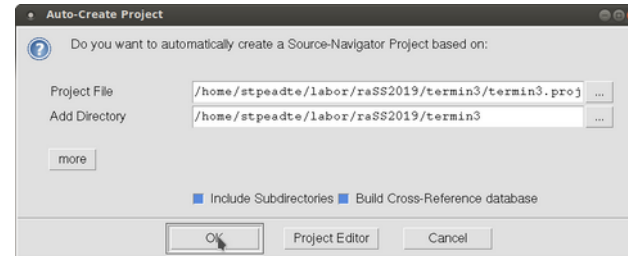
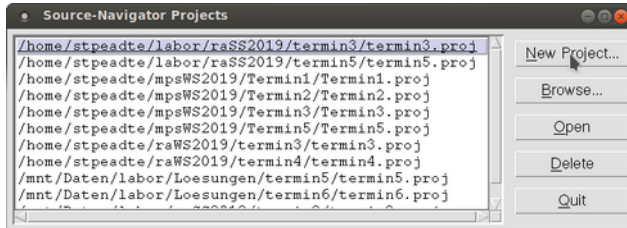


2. Wechseln Sie nun in Ihren gewählten Arbeitsordner des zu erledigenden Termins z. B. `~/raWS2023/24/termin3`
3. Öffnen Sie durch Drücken der rechten Maustaste (beim Dateimanager Thunar) im Fenster eine Konsole (Terminal). Evtl. funktioniert dies auch über die Funktionstaste „F4“.
4. Die Eingabeaufforderung (Prompt) in der Konsole sollte den Pfad des Projektordners, in dem gearbeitet werden soll, zeigen. Starten Sie nun die zur Verfügung gestellte Entwicklungsumgebung mit „**snavigator**“.

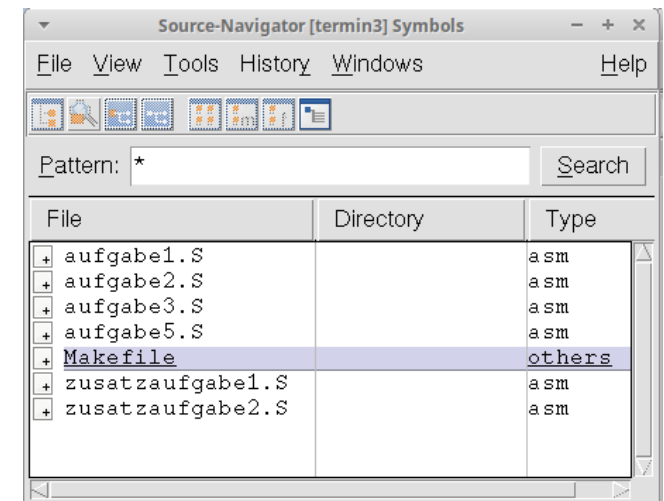
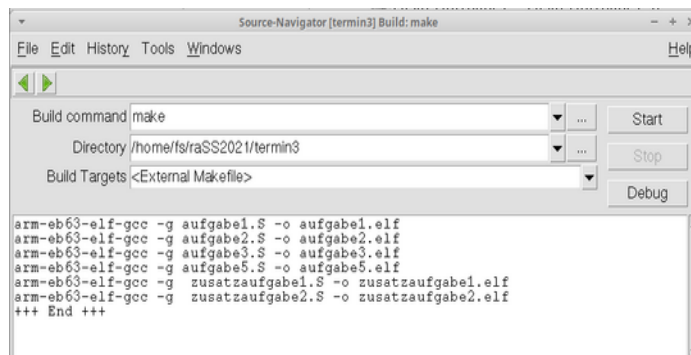
ACHTUNG HINWEIS: Verwenden Sie für Datei- und Verzeichnisnamen keine Sonderzeichen, keine Leerzeichen (Blank) und nur Zeichen aus dem 7Bit ASCII-Zeichensatz.



5. Legen Sie möglichst immer ein neues Projekt an und übernehmen die gegebenen Pfade.

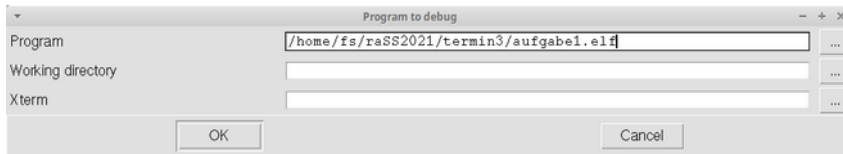


6. Im sich öffnenden Fenster Symbols können Sie durch doppeltes Anklicken einer Datei veranlassen, dass ein Editierfenster geöffnet wird.
7. Machen Sie sich auch mit dem zur Verfügung gestellten makefile vertraut.
8. Über „Tools“ „Build“ können Sie ein Build-Fenster öffnen. In diesem Fenster wird „make“ als Build command angeboten. Starten Sie make und das im Projektverzeichnis vorhandene makefile wird abgearbeitet.

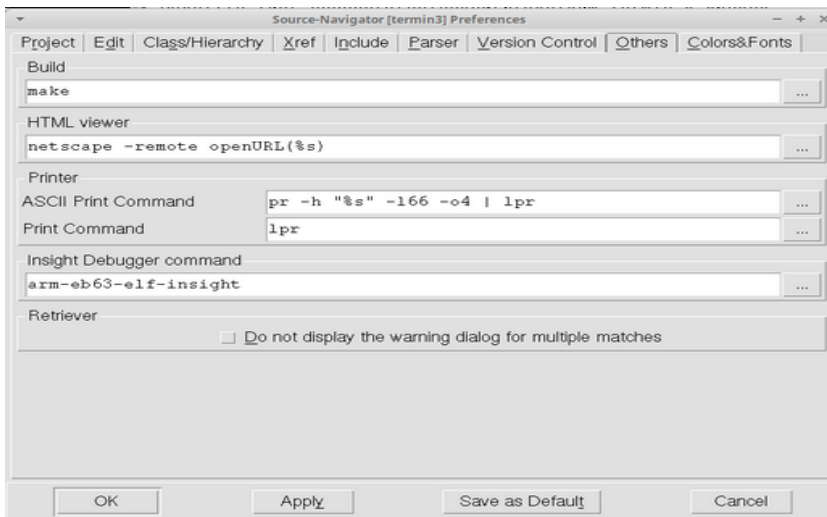


9. Im Ausgabefenster können Sie nachlesen, ob das Makefile fehlerfrei abgearbeitet wurde.

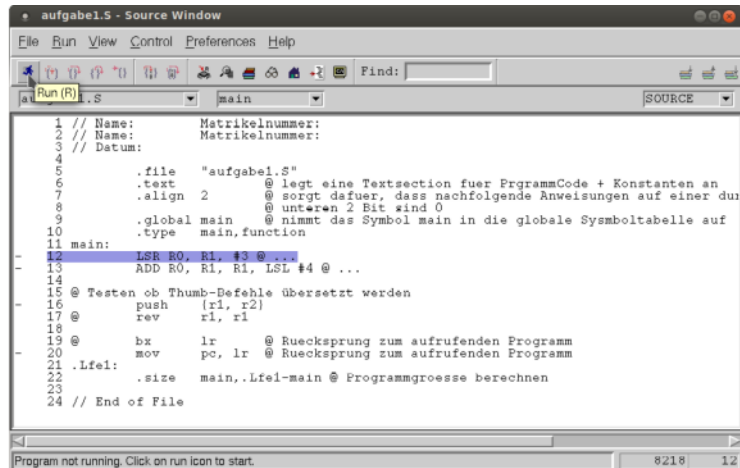
10. Zum Testen eines fehlerfreien übersetzten Programms starten Sie „Debug“ und wählen die zu testende Datei.elf.



11. Sollte der Debugger (Insight) nicht starten, stellen Sie im Fenster Symbols unter File → Project Preferences → Others das richtige Insight Debugger command (z.B. arm-eb63-elf-insight) ein. Danach sollte der Debugger funktionieren.

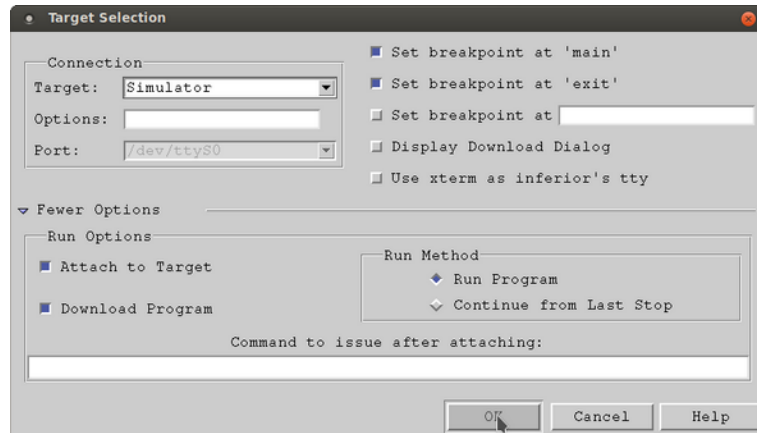


12. Drücken Sie auf den Läufer RUN(R).

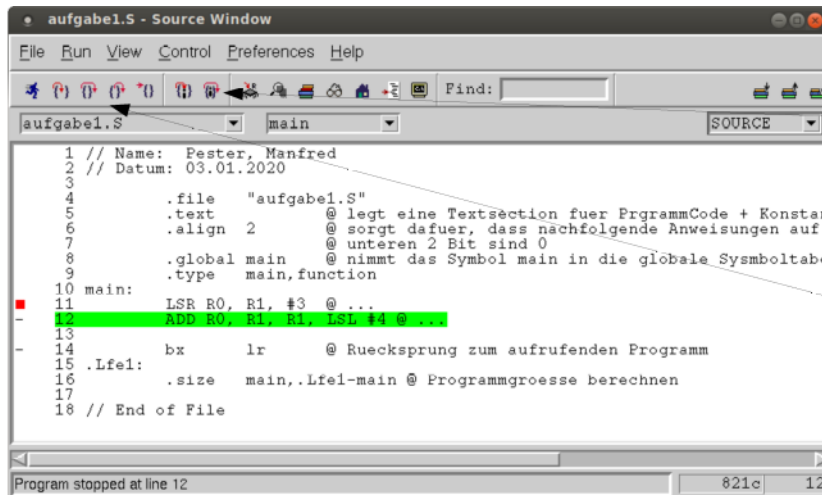


```
1 // Name:      Matrikelnummer:
2 // Name:      Matrikelnummer:
3 // Datum:
4
5     .file      "aufgabe1.S"
6     .text     @ legt eine Textsection fuer ProgrammCode + Konstanten an
7     .align 2  @ sorgt dafuer, dass nachfolgende Anweisungen auf einer du
8               @ unteren 2 Bit sind 0
9     .global  main @ nimmt das Symbol main in die globale Sysmboltabelle auf
10    .type    main, function
11 main:
12    LSR R0, R1, #3 @ ...
13    ADD R0, R1, R1, LSL #4 @ ...
14
15 @ Testen ob Thumb-Defehle übersetzt werden
16    push    {r1, r2}
17 @
18    rev    r1, r1
19
20    bx     lr @ Ruecksprung zum aufrufenden Programm
21    mov    pc, lr @ Ruecksprung zum aufrufenden Programm
22
23    .size  main, .Lfel-main @ Programmgroesse berechnen
24 // End of File
```

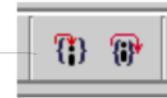
13. Im sich evtl. öffnenden Fenster Target Selection wählen Sie Target: Simulator. Sollte nur das erste mal auftreten.



14. Sie können nun Ihr Programm ausführen oder in Einzelschritten testen.

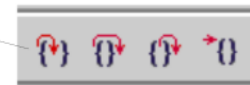


Assemblerbefehle werden durch Bedienen der Icon



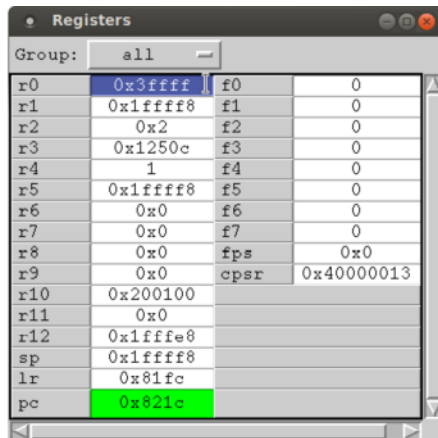
ausgelöst.

Die Icon



nur für Hochsprachendebugging verwenden.

15. Öffnen Sie ein Fenster mit den Registern um die Auswirkung der Befehle zu sehen. Änderungen in Registern immer mit der großen ENTER-Taste (nicht die Kleine vom Nummernblock) bestätigen.



Folgen Sie den Anweisungen im Labor, machen Sie sich Notizen um ein Protokoll zu diesem Termin erstellen zu können.

Haben Sie den Mut und stellen Sie Fragen!