



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbi

FACHBEREICH INFORMATIK

Hochschule Darmstadt

- Fachbereich Informatik -

Rechnerarchitektur

Einrichten

SourceNavigator Anleitung

erstellt durch:

Alexander Felger

Referent: Andreas Müller

22.10.2018

SourceNavigator starten:

Um den SourceNavigator zu starten, muss man mit Hilfe der Konsole in das korrekte Verzeichnis wechseln, in dem die Aufgabenstellungen liegen. Dazu dient unter Linux der Befehl `cd <Verzeichnispfad>`.

Um herauszufinden, wie dieser Verzeichnispfad aussieht, verwendet man den Befehl `ls`.

```
pg@pg-VirtualBox:~$ ls
Bilder      eagle      Musik      Schreibtisch workspace
Dokumente  eclipse    NetBeansProjects Videos
Downloads  MPS       Öffentlich  Vorlagen
pg@pg-VirtualBox:~$
```

```
pg@pg-VirtualBox:~$ cd Dokumente/Ra/termin3
```

Anschließend kann man mit dem `ls` Befehl noch einmal überprüfen, ob man sich im richtigen Verzeichnis befindet.

```
pg@pg-VirtualBox:~/Dokumente/Ra/termin3$ ls
aufgabe1.elf  aufgabe2.S  aufgabe5.S  SNDBA      Zusatzaufgabe.elf
aufgabe1.S   aufgabe3.S  aufgabe6.S  Termin3.pdf  Zusatzaufgabe.S
aufgabe2.elf  aufgabe5.elf  makefile    termin3.proj
pg@pg-VirtualBox:~/Dokumente/Ra/termin3$
```

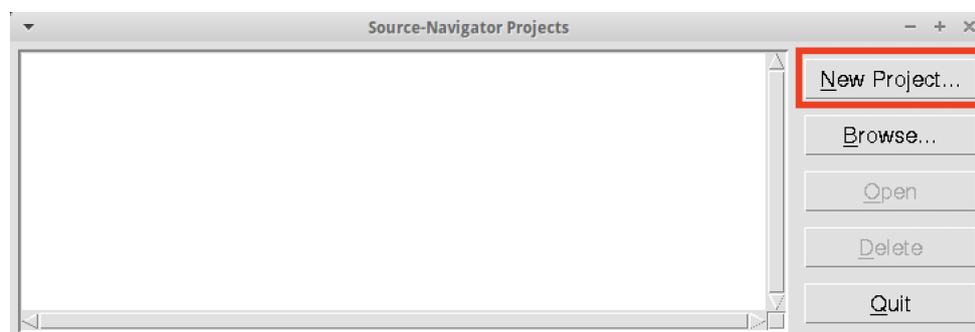
Dort angelangt, wird der SourceNavigator mit dem Befehl `snavigator` gestartet.

Es erscheint das Fenster, in dem man sich ein Projekt erstellt.

Alternativ kann man auch unter Kubuntu einfach eine Konsole im Filemanager öffnen. Dort dann den Befehl `snavigator` eingeben und es öffnet sich automatisch der Snavigator im richtigen Verzeichnis.

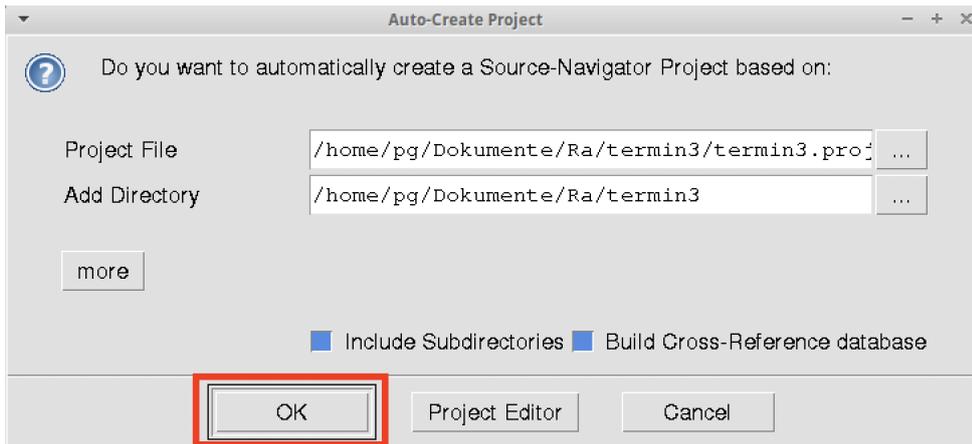
Der Snavigator sollte unbedingt im richtigen Verzeichnis gestartet werden, damit auch nur die Dateien angezeigt werden, die zu dem entsprechenden Projekt bzw. Termin gehören!

Projekt anlegen:

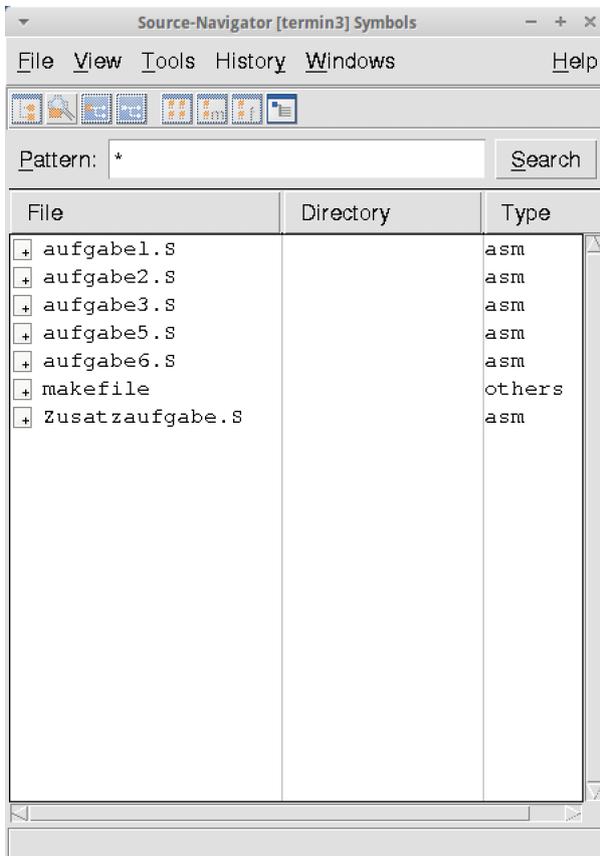


Dazu muss man lediglich auf **NewProject** klicken. Da der SourceNavigator bereits im richtigen Verzeichnis gestartet wurde, werden die Pfade für die Quelle und das Ziel des Projektes automatisch gesetzt.

Der Dialog muss lediglich bestätigt werden.



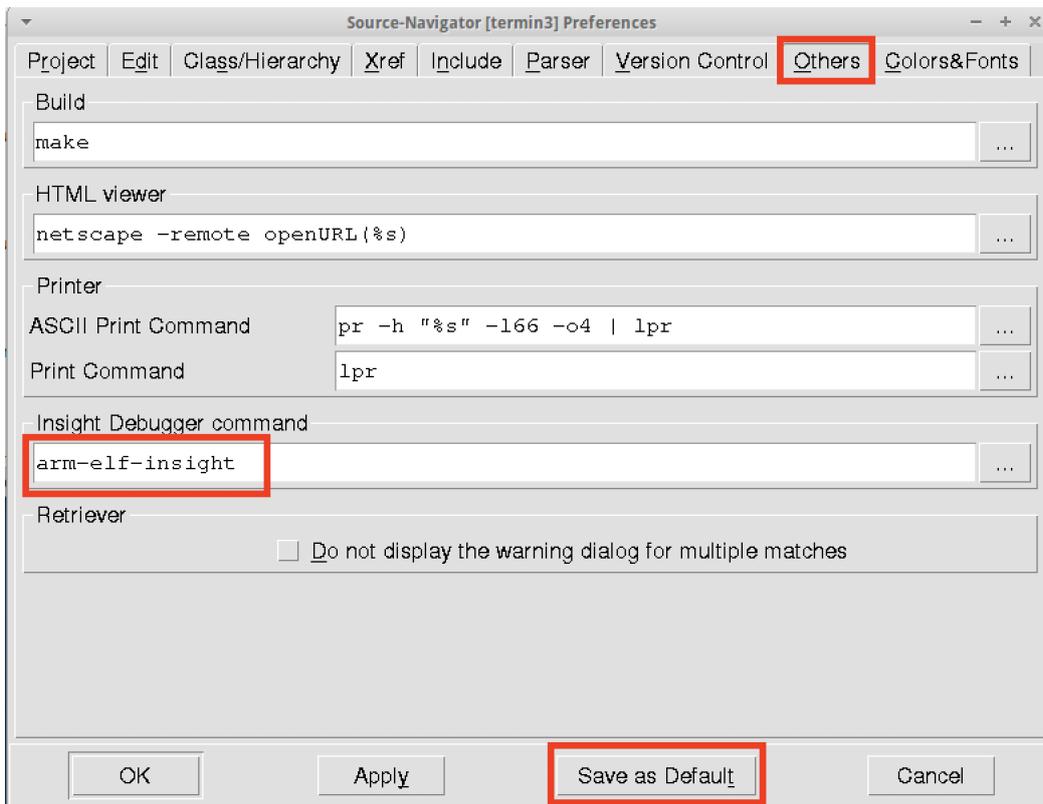
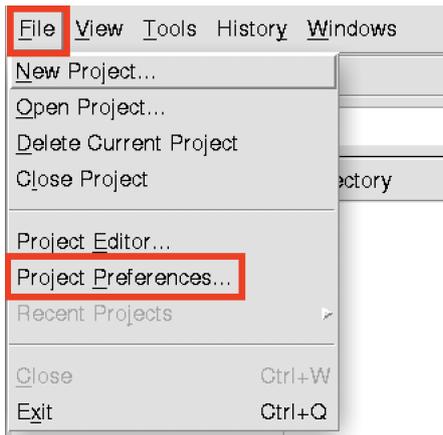
Anschließend öffnet sich ein Übersichtsfenster, in dem alle dem Projekt zugeordneten Files angezeigt werden.



Einstellungen für das Projekt:

Um das Projekt ordnungsgemäß debuggen zu können, muss das Debugger-Command angepasst werden.

Dazu muss man im File Menü die Project Preferences anpassen.



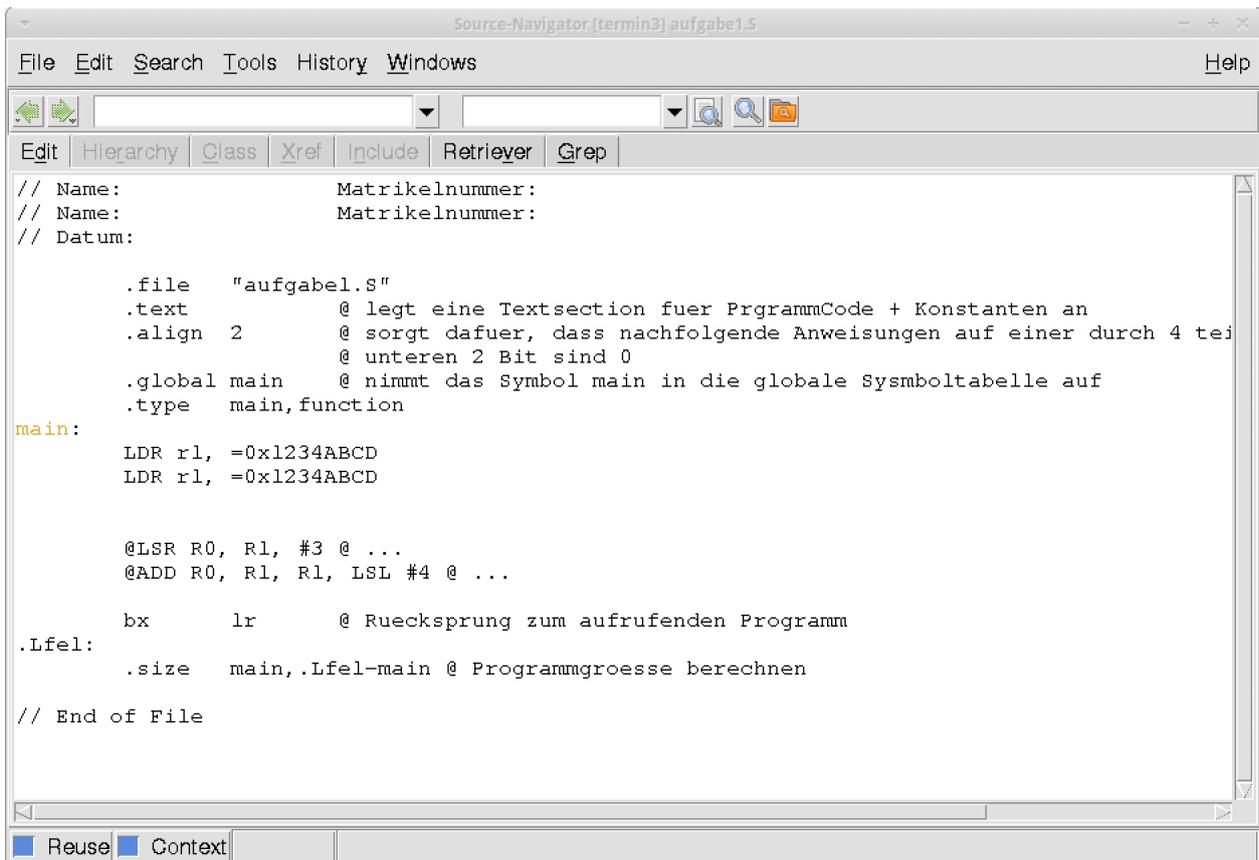
Die Zeile muss folgendermaßen lauten:



Editieren der Aufgaben:

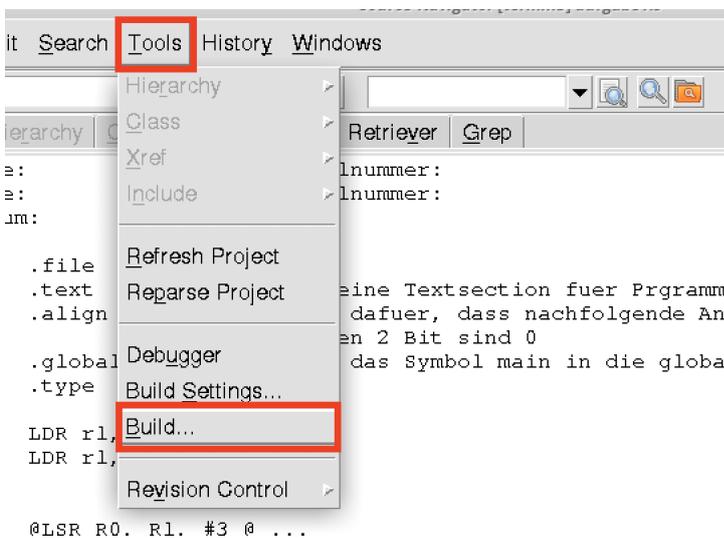
Um die Aufgaben bearbeiten zu können, kann der im SourceNavigator integrierte Editor verwendet werden. Dazu muss im Übersichtsfenster der Files einfach auf das entsprechende File für die Aufgabe doppelt geklickt werden.

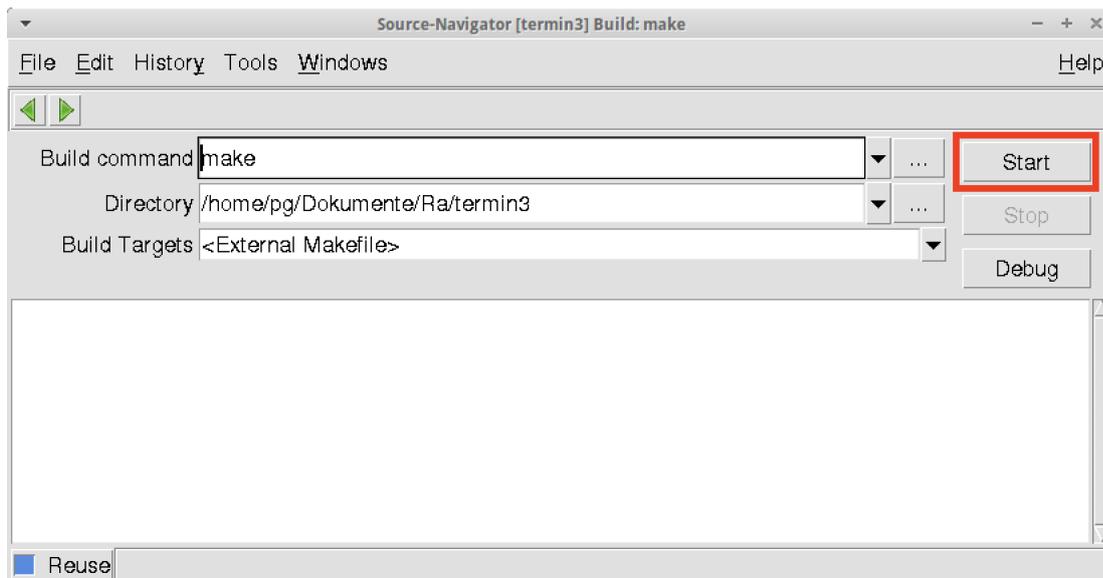
Jetzt öffnet sich der Editor.



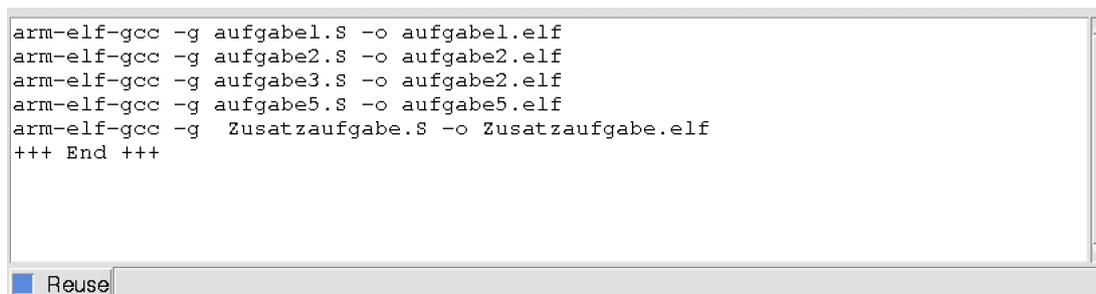
Erstellen und Debuggen des Programms:

Um das Programm zu bauen und danach zu debuggen, im Menüpunkt **Tools** auf **Build...** und dann auf **Start** klicken.

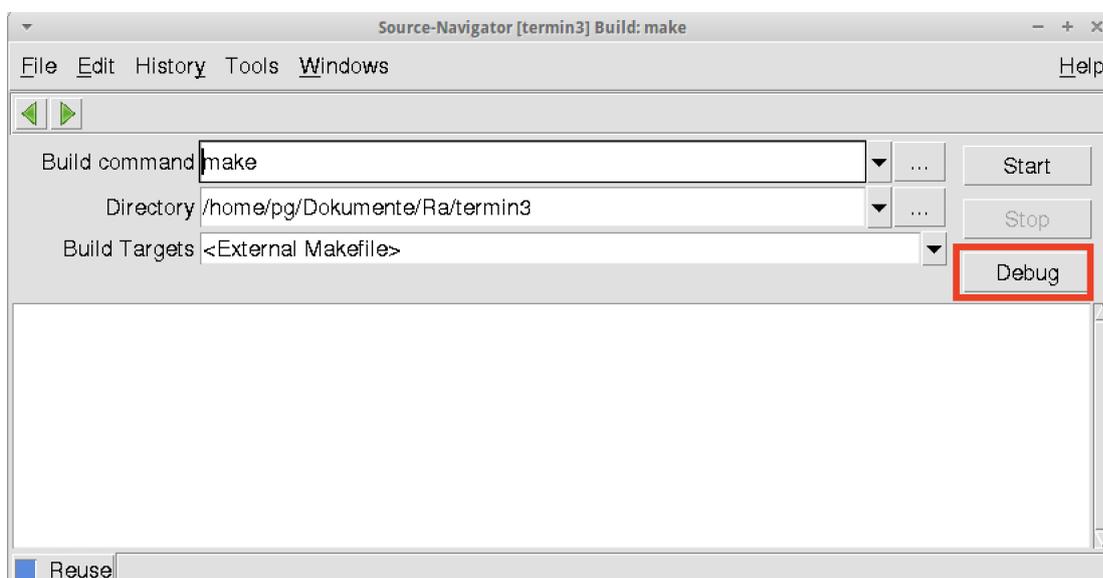


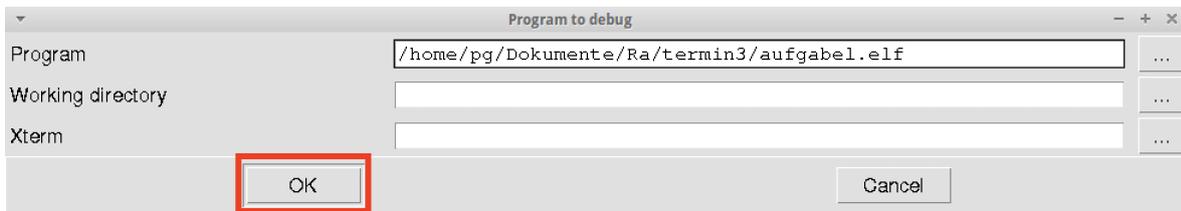
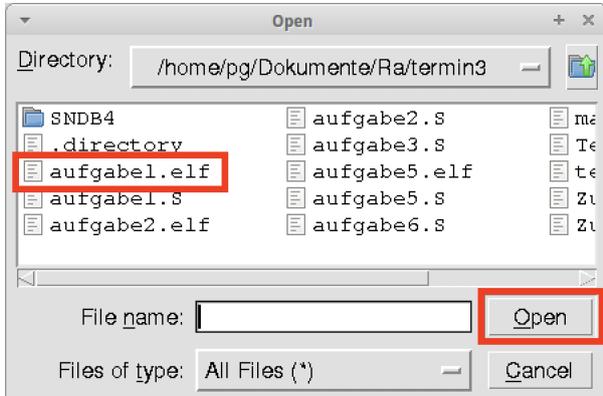
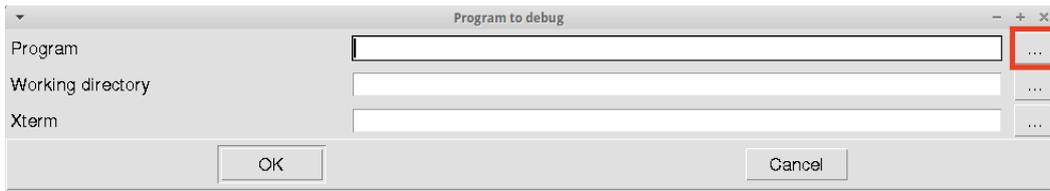


Wenn das Erstellen des Projektes erfolgreich war, kommt folgende Ausgabe:

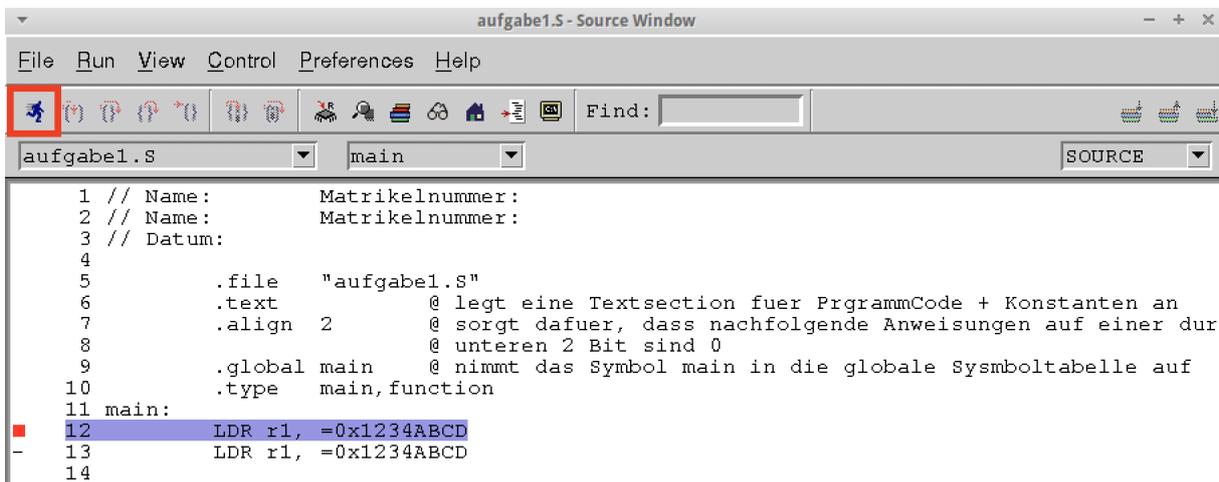


Anschließend muss das Debug-File ausgewählt werden, das debuggt werden soll. Dazu muss man auf Debug klicken und im folgenden Menü über die drei Punkte das Dateifenster zur Auswahl öffnen. Dort muss dann das **.elf** File der entsprechenden Aufgabe ausgewählt werden und mit **Open** bestätigt werden. Anschließend mit **Ok** bestätigen und es öffnet sich der Debugger.





Um das Programm zu starten, muss auf das kleine, blaue Männchen geklickt werden, welches sich oben links in der Menüleiste befindet.



Anschließend kann man mit den Steps durch das Programm schrittweise durchgehen und sich ansehen, was jeweils passiert.

```

File Run View Control Preferences Help
aufgabel.s main
1 // Name: Matrikelnummer:
2 // Name: Matrikelnummer:
3 // Datum:
4
5 .file "aufgabel.s"
6 .text @ legt eine Textsection fuer Pr
7 .align 2 @ sorgt dafuer, dass nachfolgen
8 @ unteren 2 Bit sind 0
9 .global main @ nimmt das Symbol main in die
10 .type main,function
11 main:
12 LDR r1, =0x1234ABCD
13 LDR r1, =0x1234ABCD
14

```

Um zu sehen, was passiert, sollte man sich eine Übersicht über die Register und ihre Inhalte anzeigen lassen.

```

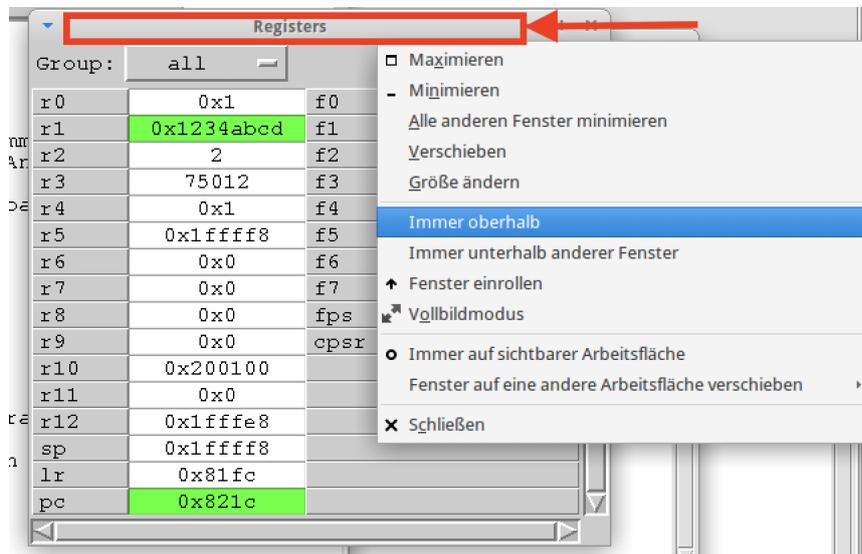
File Run View Control Preferences Help
aufgabel.s main
1 // Name: Matrikelnummer:
2 // Name: Matrikelnummer:
3 // Datum:
4
5 .file "aufgabel.s"
6 .text @ legt eine Textsection fuer Pr
7 .align 2 @ sorgt dafuer, dass nachfolgen
8 @ unteren 2 Bit sind 0
9 .global main @ nimmt das Symbol main in die
10 .type main,function
11 main:
12 LDR r1, =0x1234ABCD
13 LDR r1, =0x1234ABCD
14

```

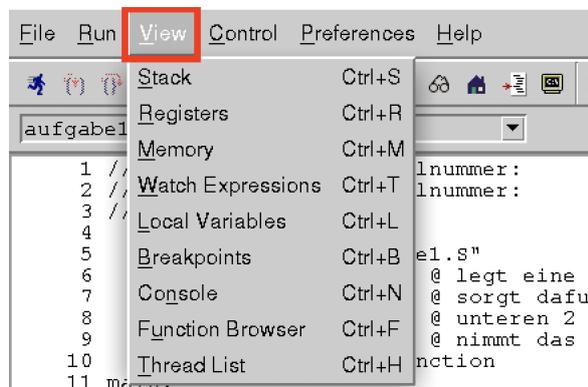
Das Registerfenster zeigt die Register und ihre Inhalte. Wenn sich in einem Befehl ein Inhalt ändert, wird dieser grün markiert.

Register	Value	Register	Value
r0	0x1	f0	0
r1	0x1234abcd	f1	0
r2	2	f2	0
r3	75012	f3	0
r4	0x1	f4	0
r5	0x1ffff8	f5	0
r6	0x0	f6	0
r7	0x0	f7	0
r8	0x0	fps	0x0
r9	0x0	cpsr	0x40000013
r10	0x200100		
r11	0x0		
r12	0x1fffe8		
sp	0x1ffff8		
lr	0x81fc		
pc	0x821c		

Um ein Fenster immer über den anderen Fenstern anzeigen zu lassen, und so zu verhindern, dass man nach jedem Step das Fenster erneut in der Vordergrund holen muss, kann man in die Titelleiste des Fensters rechtsklicken und anschließend den Menüpunkt **Immer oberhalb** auswählen.



Es gibt auch die Möglichkeit sich noch weitere interessante Informationsfenster anzeigen zu lassen, dies geht über den Menüpunkt View.



Viel Spaß beim Arbeiten mit dem SourceNavigator!