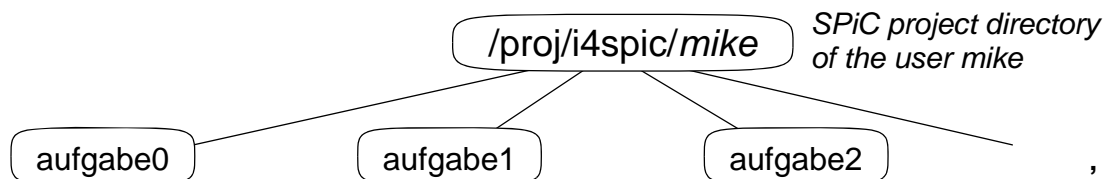


General instructions for the SPiC exercises

- All assignments need to be solved on one's own. There will be no group-submissions.
- Solutions need be submitted prior to the respective deadline.
- A login for the computer science CIP pools is required in order to use the computers. If you do not have a login yet, please visit the CIP administrators during the consultation hours. More information is available at <http://www.cip.informatik.uni-erlangen.de>.
- A project directory will be created for each participant located at `/proj/i4spic/` named with the login name of the user. Project directories will be created for all users that signed up for a tutorial with the WAFFEL system. **Signing up with WAFFEL therefore is mandatory.** The directory tree for the exercises needs to be structured as follows:



- Solutions can be submitted till the deadline by invoking the program
`/proj/i4spic/pub/abgabe aufgabeX` $X = 0 \dots n$
 The program will check the directory structure and file names required according to the assignment and creates an archive containing the submitted files. The program can be invoked repeatedly till the deadline to replace prior submissions. The latest duly submission will be rated.
- If for some good reason you fail to submit your solution on time, a late submission is possible by supplying the **-forced** parameter to the submission program:
`/proj/i4spic/pub/abgabe aufgabeX -force` $X = 0 \dots n$
 Consultation with your tutor is required for the late submission to be considered. A prior duly submission will **not** be replaced by a late submission and can be rated in any case.
- The program is to be placed in a C file named after the program name in the title of the assignment, except the assignment contains differing instructions. For instance, if the program name is **wcount** according to the assignment, the solution should be placed in a file named **wcount.c**.

Aufgabe 0: countdown (8 pts, submission deadline Wed, 10/29/08 8:00 pm)

Implement a C-program **countdown** in a file **countdown.c** for the SPiCboard to get used to the required tools (text editor, compiler, shell). The program should display a countdown from 9 to 0 on the least-significant 7-segment display (*Einer* on the SPiCboard sketch). Each number is displayed for a waiting time implemented by a busy loop with 25.000 passes. After reaching 0, the program will perform a busy loop with 60.000 passes and then restart the countdown.

The program needs to be structured so that the busy loop and the initialization of the I/O ports each are performed in a separate function.

In addition, the source code needs to be commented in a way that illustrates the intention of each single line in the source code, whereby multiple lines that logically belong together can be described by a single comment.

Ensure, that your source code compiles with the required compiler flags, which are printed in the slides of the first tutorial.