General Remarks for the SLP Computer Exercises

- You need a login for the computer-science CIP pool to take part in exercise course. If you do not have a login, one can be created via `https://account.cip.cs.fau.de`.
- Anyone, who registered for the exercises via Waffel, receives a project directory `/proj/i4spic/<login>/`, where `<login>` is a placeholder for your login name. A registration in the system is therefore mandatory to work on the assignments! The project directory is automatically integrated in the SPiC-IDE.
- The structure of the directory for the assignments has to be organized as follows:
  `/proj/i4spic/<login>/aufgabe1`
  `/proj/i4spic/<login>/aufgabe2`
  ...
- The assignments have to be submitted in the SPiC-IDE not later than the deadline.
  Alternatively, they can be submitted via
  `/proj/i4spic/bin/submit aufgabeX`
  (with `X = 1 ... n`). This script copies the files required by the assignment description from the corresponding directory. Before the deadline, any program can be submitted an arbitrary number of times – the most recently submitted version will then be graded after the deadline.
- To check the last (and therefore valid) submission, the SPiC-IDE can be used or via
  `/proj/i4spic/bin/show-submission aufgabeX`
  you can view the last submitted program. To only view differences between the last submission and the current status in the project directory, the option `-d` can be added.
  `/proj/i4spic/bin/show-submission -d aufgabeX`
- The latest date for submission can be seen in the SPiC-IDE or with the call of:
  `/proj/i4spic/bin/get-deadline aufgabeX`
- Grading of a program submitted after the deadline can only be done in **well reasoned and exceptional cases**. You need to address the tutor directly who will then decide individually. An earlier submission before the deadline is *not* overwritten by a late submission. If in doubt, the first one is therefore graded.
- This term, the SPiCsim as well as the SPiCboard serves as a reference for the correction of the assignments. Please make sure that your solution behaves on earch of the platforms exactly as required by the assignment description.
- If not specified further, you need to use the same name for the C source file as the title of the assignment is called. I.e., if the assignment is called *blink*, the program should be created as `blink.c`.
- Further information can be found online:
  `https://sys.cs.fau.de/lehre/ss25/spic/`
- The documentation of the `libspicboard` can also be found there:
  `https://sys.cs.fau.de/lehre/ss25/spic/uebung/spicboard/libapi`

SLP-assignment #3.3: dice

(6 points, no groups)

To get to know the `libspicboard`, program a cube in the file `wuerfel.c`. The program should output a number between 1 and 6 on the 7-segment display when button 0 is pressed. To generate a random number, a variable is to be incremented in a loop. As soon as the button is pressed, a new dice result is output. While the button is pressed, the counter must continue counting to generate the next result. The display should not change while the button is pressed. Only the function `sb_button_getState()` should be used to query the button. Please note that to use the 7-segment display, the interrupts must be activated by calling `sei()` once.

Deadline

Use script in CIP pools: `/proj/i4spic/bin/get-deadline aufgabe3.3 Txx`