

---

# Allgemeine Hinweise zu den SPiC-Übungen

- Die Aufgaben sind teils alleine, teils in Zweier-Gruppen zu bearbeiten, wobei der Lösungsweg und die Programmierung gemeinsam erarbeitet werden sollen. Beachten Sie hierzu auch die Hinweise in den Aufgabenstellungen.
- Sie benötigen einen Login für die CIP-Pools der Informatik, um die Rechner in den Übungsräumen verwenden zu können. Falls Sie noch keinen Login besitzen, suchen Sie bitte die CIP-Betreuer innerhalb deren Sprechstunden auf. Informationen hierzu finden Sie unter <http://wwwcip.cs.fau.de>.
- Alle an der Übung Teilnehmenden erhalten für SPiC jeweils ein spezielles Projektverzeichnis mit dem Namen `/proj/i4spic/<login>/`, wobei `<login>` für den eigenen Login-Namen steht. Die Projektverzeichnisse werden für alle Teilnehmer angelegt, die sich im Waffel-System angemeldet haben. Eine Anmeldung im Waffel-System ist daher zwingend zur Übungsteilnahme erforderlich! Die Projektordner werden auch in die SPiC-IDE eingebunden.
- Der Verzeichnisbaum für die Aufgaben ist folgendermaßen aufzubauen:  
`/proj/i4spic/<login>/aufgabe1`  
`/proj/i4spic/<login>/aufgabe2`  
...
- Die Aufgaben sind bis spätestens zum Abgabetermin durch Aufruf des Programms  
`/proj/i4spic/bin/submit aufgabeX`  
(mit `X = 1 ... n`) abzugeben. Dieses Programm kopiert die in der Aufgabenstellung verlangten Dateien aus dem entsprechendem Verzeichnis. Bis zum Abgabetermin kann ein Programm beliebig oft abgegeben werden – es gilt der letzte, vor dem Abgabetermin vorgenommene Aufruf des Abgabeprogramms. Alternativ kann auch aus der SPiC-IDE abgegeben werden.
- Um seine letzte (und damit gültige) Abgabe zu überprüfen, kann die Abgabe per  
`/proj/i4spic/bin/show-submission aufgabeX`  
betrachtet werden. Um nur die Unterschiede zwischen der letzten Abgabe und dem aktuellen Stand im Projektverzeichnis anzuschauen, kann die Option `-d` übergeben werden.  
`/proj/i4spic/bin/show-submission -d aufgabeX`
- Die Abgabezeitpunkte sind abhängig von der eigenen Tafelübung. Ihren eigenen Abgabezeitpunkt können Sie durch die SPiC-IDE abfragen oder durch Aufruf des Programms  
`/proj/i4spic/bin/get-deadline aufgabeX`
- Eine genaue Beschreibung der SPiC-IDE und ihrer Funktionen ist unter [https://www4.cs.fau.de/Lehre/SS19/V\\_SPIC/SPiCboard/cip.shtml](https://www4.cs.fau.de/Lehre/SS19/V_SPIC/SPiCboard/cip.shtml) zu finden.
- Eine Wertung bei Abgabe nach dem Abgabezeitpunkt kann **nur in begründeten Ausnahmefällen** nach Rücksprache mit Ihrem Übungsleiter erfolgen, der dann entscheidet, ob die verspätete Abgabe noch gewertet wird. Eine frühere, fristgerechte Abgabe wird durch eine verspätete Abgabe *nicht* überschrieben und im Zweifelsfall gewertet.
- Verwenden Sie für den Namen der C-Quelldatei, soweit in der Aufgabenstellung nicht anders angegeben, den Namen des Programms entsprechend dem Titel der jeweiligen Aufgabenstellung. Ist der Titel der Aufgabenstellung also z. B. *blink*, so legen Sie den Quellcode in einer Datei `blink.c` ab.
- Weitere Informationen finden Sie auf den Vorlesungsseiten:  
[https://www4.cs.fau.de/Lehre/SS19/V\\_SPIC/](https://www4.cs.fau.de/Lehre/SS19/V_SPIC/)
- Die Dokumentation der `libspicboard` finden Sie ebenfalls auf der Vorlesungsseite:  
[https://www4.cs.fau.de/Lehre/SS19/V\\_SPIC/SPiCboard/libapi.shtml](https://www4.cs.fau.de/Lehre/SS19/V_SPIC/SPiCboard/libapi.shtml)
- Details zur Entwicklungsumgebung sind hier zu finden:  
[https://www4.cs.fau.de/Lehre/SS19/V\\_SPIC/SPiCboard/cip.shtml](https://www4.cs.fau.de/Lehre/SS19/V_SPIC/SPiCboard/cip.shtml)
- Eine Anleitung um den SPiCsim zu verwenden finden Sie hier:  
[https://www4.cs.fau.de/Lehre/SS19/V\\_SPIC/SPiCboard/spicsim.shtml](https://www4.cs.fau.de/Lehre/SS19/V_SPIC/SPiCboard/spicsim.shtml)

---

# SPiC-Aufgabe #1: blink

**(3 Punkte, keine Gruppen)**

Programmieren Sie zum Kennenlernen der benötigten Programmierwerkzeuge zunächst ein C-Programm `blink` für die AVR-Entwicklungsboards, welches abwechselnd die LEDs `YELLOW0` und `YELLOW1` (siehe SPiCboard-Skizze) an- bzw. ausschaltet. Es soll also immer eine der beiden LEDs aktiviert sein und die jeweils andere deaktiviert, wie es etwa bei einem Warnlicht für Baufahrzeuge der Fall ist. Die LEDs sollen in Zeitintervallen von etwa einer halben Sekunde blinken. Die Realisierung der Wartezeit soll durch eine aktive Warteschleife erfolgen; die zum Erreichen der Wartedauer notwendige Schleifenanzahl soll von Ihnen empirisch ermittelt werden.

Verwenden Sie zur Ansteuerung der LEDs die SPiCboard-Bibliothek und suchen Sie in der Onlinedokumentation der Bibliothek nach den benötigten Funktionen. Speichern Sie den Programmquellcode in einer Datei `blink.c` ab.

## Hinweise

- Im Verzeichnis `/proj/i4spic/pub/aufgabe1/` befindet sich die Datei `blink.elf`. Mit dieser flaschbaren Beispielimplementierung können Sie die gewünschte Verhaltensweise des Programms nachvollziehen.
- Achten Sie darauf, dass bei Ihrer abgegebenen Lösung der Compiler keine Warnungen oder Fehler beim Übersetzungsvorgang ausgibt.

## Abgabezeitpunkt

T01	05.05.2019	18:00:00
T02	05.05.2019	18:00:00
T03	05.05.2019	18:00:00
T04	05.05.2019	18:00:00
T05	06.05.2019	18:00:00
T06	06.05.2019	18:00:00
T07	06.05.2019	18:00:00
T08	07.05.2019	18:00:00
T09	07.05.2019	18:00:00
T10	07.05.2019	18:00:00
T11	07.05.2019	18:00:00
T12	08.05.2019	18:00:00
T13	08.05.2019	18:00:00
T14	08.05.2019	18:00:00
T15	08.05.2019	18:00:00
T16	08.05.2019	18:00:00
T17	09.05.2019	18:00:00
T18	09.05.2019	18:00:00