


9 Versuchsreihe

In der vorletzten Versuchsreihe sollen Sie zwei Regelungen für das Modellhaus entwerfen und testen, bevor Sie dann im letzten Versuch zum Abschluss einmal das Gesamtsystem mit Bluetooth-Anbindung zu einem Android Gerät in Betrieb nehmen. Die notwendigen Techniken hierfür haben Sie bereits mehrfach in den vorherigen Versuchen angewandt. Sollten Sie Probleme bei der Umsetzung haben, lohnt es sich, insbesondere die Versuchsreihen 5-7 noch einmal durch zu schauen.

Kompetenzen Die Studierenden

- können Schaltungen mittels einer Beschreibungssprache entwerfen,
- verstehen den Umgang mit der Hardwarebeschreibungssprache VHDL und deren Zusammenspiel mit der Hardware,
- zeigen Lernbereitschaft und besitzen Offenheit für neue Ideen und Anforderungen,
- kennen die besonderen Randbedingungen des Entwurfs eingebetteter Systeme,
- verstehen den Zusammenhang zwischen Hardware-Konzepten und den Auswirkungen auf die Software bzw. die Anwendung,

9.1 Vorbereitung

Hinweis: Die Lösungen zu den Vorbereitungsaufgaben, insbesondere jene die mit einem  gekennzeichnet sind, müssen bis **spätestens 18 Uhr am Vortag des Laborterminals** per Sciebo (einmal pro Gruppe, Bsp.: für die Montagsgruppe wäre dies Sonntag 18 Uhr) über folgenden Link hochgeladen werden:

Für die Gruppen von Montag (Passwort: Transistor):

<https://uni-siegen.sciebo.de/s/g4Nv14QVgFzk71H/authenticate>

Für die Gruppen von Dienstag (Passwort: Diode):

<https://uni-siegen.sciebo.de/s/cdfkb0cHmtvsmEk/authenticate>

Für die Gruppen von Mittwoch (Passwort: Kondensator):



<https://uni-siegen.sciebo.de/s/64Lje097Gdaw4mB/authenticate>

Dateiname: Gruppenname_ Versuchsreihe


Beispiel: Gruppe 01 und Vorbereitung für Versuch 1: G01_ V1



Wenn der Link unter Sciebo temporär nicht zu erreichen ist, dürfen Sie die Vorbereitung auch an folgende Mail-Adresse verschicken: ferid.mahdi@uni-siegen.de.

Fehlendes Zusenden der Lösungen führt zum Ausschluss vom Labortermin!



1.   Entwerfen Sie als erstes für Ihre Hardware eine Regelung, welche die LED in Abhängigkeit der Helligkeit im Haus regelt (Modul *AshaRegelung*). Bei weniger



als 10 Lux soll die LED zu 100%, bei weniger als 50 Lux zu 50% leuchten und bei weniger als 200 Lux zu 25% leuchten. Bei mehr als 200 Lux soll die LED aus sein.

-  Berechnen Sie die zu den Lux-Werten gehörenden Spannungswerte (umgekehrte Rechnung wie in Versuch 6).
- Bereiten Sie diese Regelaufgabe für die Durchführung so gut wie möglich vor.

2.   In der zweiten Aufgabe sollen Sie die Temperatur im Haus regeln (Modul *AshaRegelung*). Die automatische Regelung soll die Temperaturwerte des Außen- und Innentemperatursensors vergleichen und das Peltier-Element so schalten, dass es im Haus immer 2°C wärmer ist als draußen. Schalten Sie mit dem Peltier beim Heizen (*PeltierDirectionControl*='1') oder Kühlen (*PeltierDirectionControl*='0') immer auch die Lüfter an. Berücksichtigen Sie auch, dass das Peltier nicht beliebig schnell umgepolt werden darf, wenn die Zieltemperatur erreicht ist (also kein ständiger Wechsel zwischen Heizen und Kühlen stattfinden soll). Bereiten Sie diese Regelaufgabe für die Durchführung vor.



Hinweis: Um mit den 2°C arbeiten zu können, muss der Wert erst noch in einen Spannungswert umgerechnet werden. Dies geht am besten, wenn Sie (ähnlich wie in der vorherigen Aufgabe) zwei Berechnungen für bspw. 25°C und 27°C rückwärts rechnen und von den Ergebnissen die Differenz bilden.

-  Wie kann das beschriebene zu schnelle Umschalten des Peltiers verhindert werden?
-  Welchem (dezimalen) Spannungswert entspricht 2°C ?



3.  Im nächsten Versuch wird die Bluetooth-Verbindung zwischen FPGA und einem Android Gerät in Betrieb genommen. Schauen Sie sich dafür schon mal das in moodle hochgeladene Tutorial „Erstellen einer Bluetooth-Verbindung“ an, um einen groben Überblick über die Verwendung von Bluetooth in Android zu bekommen.
Hinweis: Das Tutorial dient zum allgemeinen Verständnis. Wenn Sie es selber einmal durchführen möchten, müssen eventuell leichte Veränderungen vorgenommen werden, da dieses Tutorial noch für die (inzwischen veraltete) Android-Erweiterung von Eclipse geschrieben wurde.
4.  Beantworten Sie die Fragen des 9. Moodle-Selbsttestes. Sollten Sie bei der Beantwortung Probleme haben, müssen Sie ggf. Teile der Vorbereitung nochmal wiederholen.

Tipp: Schauen Sie sich am besten auch schon die Aufgaben der Durchführung an und führen Sie gegebenenfalls selbstständig weitere Vorbereitungen durch, damit Sie in der Laborphase nicht unnötig Zeit verlieren!

9.2 Durchführung

1.   Realisieren Sie die oben beschriebene Helligkeitssteuerung und Temperatursteuerung. In Abbildung 9.1 ist die bereits aus den Versuchen 6 und 7 bekannte FPGA Schalterbelegung dargestellt. Implementieren Sie den *Modus 3: geregelte Aktorsteuerung*, indem Sie die markierte Stelle im Modul *actor* ergänzen. Testen Sie

die Steuerung und lassen Sie das Ergebnis von einem Betreuer abnehmen.

2.   Ergänzen Sie die Module *AshaRegelung* und *actor* so, dass auf der Sieben-segment-Anzeige die Temperaturdifferenz bzw. der Helligkeitswert angezeigt werden, wenn die Anzeige der Sensorwerte (*SW7*) nicht aktiv ist (jeweils die Werte die direkt vom Sensor kommen). Verwenden Sie hierfür die Signale *ControlTempDiffOut* und *ControlLightDiffOut*. Sind beide Regelungen aktiv, soll zwischen den beiden Anzeigen umgeschaltet werden können (entweder durch Drücken eines Buttons oder mit Hilfe eines automatischen Wechsels alle 5 Sekunden).

Tipp: Das höchstwertige Bit von *ControlTempDiffOut* können Sie verwenden um anzuzeigen, ob aktuell die Innentemperatur größer ist oder die Außentemperatur.

FPGA Switch und Button Belegung

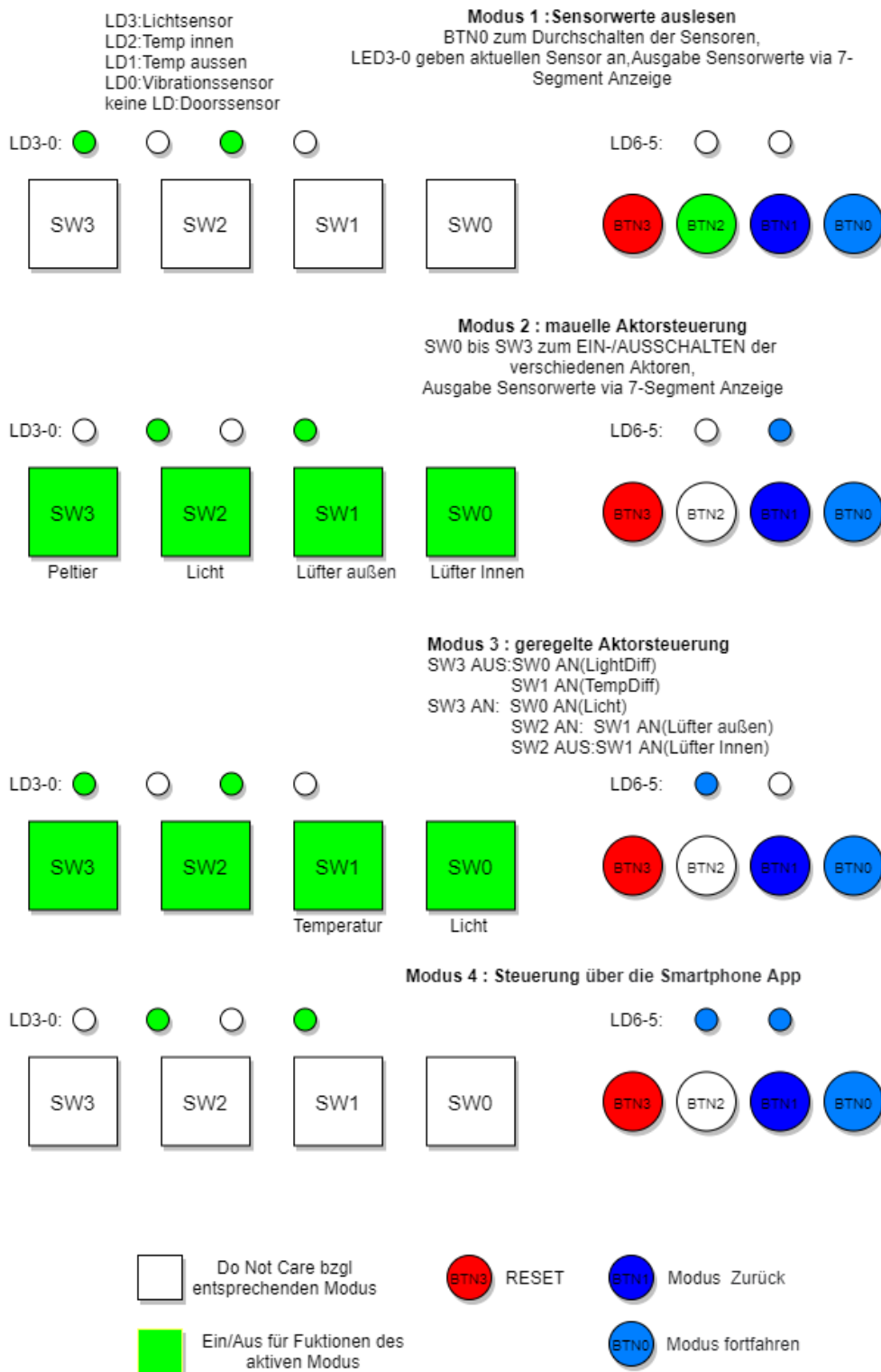


Abbildung 9.1: FPGA-Ansteuerung des Hauses

9.3 Protokollierung

1. ✎ Beschreiben Sie kurz, wie Sie die beiden Regelungsaufgaben gelöst haben.
2. ✎ Kommentieren Sie das Modul *AshaRegelung* ausführlich und drucken Sie den Code fürs Protokoll aus. Verwenden Sie dafür den Texteditor und den Drucker **Brother**.