

## Lehrstuhl Informatik XII

Jürgen Mäter (Informatik IV)

Übungen zur Vorlesung  
„Rechnergestützter Entwurf  
von Mikroelektronik (REM)“  
Sommersemester 2008

Mittwoch, 30.04.2008

## Übungsblatt 4

- Abgabe:** Bis Mittwoch, 07.05.2008, 16:00 Uhr; in der Vorlesung oder in den Übungsgruppen oder per Email an den jeweiligen Übungsgruppenleiter.
- Hinweise:** Gruppenarbeit von bis zu drei Personen aus der gleichen Übungsgruppe ist ausdrücklich erwünscht.  
Bitte vergessen Sie nicht, Ihre Namen, Ihre Matrikelnummern auf die Lösung zu schreiben:  
Übungstermin: Mittwoch, 10-12 Uhr, OH16/E07  
Anwesenheitspflicht in den Übungen!  
Bitte Krankmeldungen an den Übungsgruppenleiter.

### Kriterien für die Scheinvergabe:

- Es gibt 13 Übungszettel
- Insgesamt 50% der Punkte müssen erreicht werden
- Die Ergebnisse müssen vor der Gruppe präsentiert werden können

### Aufgabe 4.1 (16 Punkte)

Modellieren Sie ein Sensor/Alarm-System aus folgenden Komponenten:

- **Ein Temperatursensor**  $t_{\text{sensor}}$ , der alle 5 Sekunden die Temperatur misst. Der Sensor ist über einen FIFO-Kanal A mit einem Rechner verbunden. Die Einträge im FIFO A bestehen aus einer Struktur, die ein Zeichen zur Kennzeichnung des Sensors und einen Gleitkommawert zur Angabe der Meßgröße enthält. Der Temperatursensor kennzeichnet seine Einträge im FIFO A mit einem ‚T‘. Bei einer Abweichung von mehr als 0,1 Grad gegenüber dem vorherigen Wert sowie nach dem Lesen des allerersten Wertes schreibt der Sensor eine Nachricht in den FIFO-Kanal. Die gemessenen Werte werden über eine Datei temperatures.txt simuliert, die auf der Übungswebseite bereitgestellt wird. Die Datei enthält pro Zeile einen Messwert im Gleitkommaformat.
- **Ein Feuchtigkeitssensor**  $f_{\text{sensor}}$ , der alle 10 Sekunden die Feuchtigkeit misst. Die gemessenen Werte werden über eine Datei humidity.txt simuliert, die auf der Übungswebseite bereitgestellt wird. Die Datei enthält pro Zeile einen Messwert (in %) im

(Bitte wenden)

Gleitkommaformat. Der Sensor ist ebenfalls über den FIFO-Kanal mit einem Rechner verbunden. Der Feuchtigkeitssensor kennzeichnet seine Einträge im FIFO mit dem Zeichen ‚F‘.

- **Ein Rechner** `rechner`, der alle Einträge in dem FIFO-Kanal A entgegen nimmt. Bei jedem Erhalt eines Eintrags wird ein Tripel bestehend aus der aktuellen Simulationszeit, der letzten bekannten Temperatur und der letzten bekannten Feuchtigkeit in einen zweiten FIFO-Kanal B geschrieben, der mit einer Speichereinheit verbunden ist. Das Schreiben startet erst, wenn Temperatur *und* Feuchtigkeit bekannt sind. Wenn die Temperatur außerhalb eines Bereiches von  $0 < T < 30$  oder die Feuchtigkeit außerhalb eines Bereiches von  $20 < F < 95$  % liegt, schreibt der Rechner (über `cout`) eine Warnung. Nach dem Lesen eines Eintrags im FIFO-Kanal A und der entsprechenden Verarbeitung kann der Rechner erst nach 3 Sekunden wieder den nächsten Eintrag aus dem Kanal A lesen.
- **Eine Speichereinheit** `logger`, die vom Kanal B liest und die alle Einträge speichert (dies wird durch eine Ausgabe über `cout` simuliert). Die Speichereinheit kann den jeweils nächsten Eintrag im Kanal erst 2 Sekunden nach dem Lesen des vorherigen Eintrags entgegen nehmen.

Die Kanäle sind unter Verwendung der vordefinierten `fifo`-Klassen und der zugehörigen Interfaces zu modellieren.

Simulieren Sie insgesamt 200 Sekunden. Die Eingabe-Dateien enthalten dafür ausreichend viele Messwerte. Prüfen Sie, ob durch gleichzeitige Schreibvorgänge beim Kanal A Daten verloren gehen.

## Aufgabe 4.2 (4 Punkte)

- Ersetzen Sie das Einlesen von Daten durch die Sensoren selbst durch eine **Testumgebung** `tester`. `tester` liest Daten aus einer Datei `events.txt`. Die Datei enthält pro Zeile ein Tripel aus einer Zeitangabe (Sekunden in Gleitkommadarstellung), dem Typ des Sensors (‚T‘ für Temperatur, ‚F‘ für Feuchte) und dem neuen Wert für diesen Sensor. `tester` besitzt zwei Ports für `sc_signal`-Kanäle `t` und `f`, die mit den Sensoren verbunden sind. In den Sensoren ist das Lesen von Dateien zu ersetzen durch das Lesen von diesen Kanälen. Simulieren Sie insgesamt 200 Sekunden.
- Beschreiben Sie kurz in Textform, was Sie ändern müssten, damit der Tester auch die Ergebnisse automatisiert überprüft!

## Anmerkung:

- Die Abgabe der Lösungen von Programmieraufgaben **muss** per email ([Juergen.Maeter@udo.edu](mailto:Juergen.Maeter@udo.edu)) an den Übungsgruppenleiter erfolgen.
- Einzureichen sind alle Quelldateien in kompilierfähiger Version sowie die Console-Ausgaben in einer Text-Datei. Es sollen **KEINE** ausführbaren / fertig kompilierten Dateien abgegeben werden.
- Bitte fassen Sie die Lösungen nach Aufgaben getrennt in einem ZIP-Ordner zusammen.