

Praktikum Rechnerstrukturen– Aufgabenblatt 2

Die folgenden Aufgaben sollen in 2er Gruppen gemeinsam an einem PC bearbeitet werden. Das Kopieren von fertigen Lösungen, die von Personen außerhalb der Arbeitsgruppe erstellt wurden wird als Täuschungsversuch gewertet und kann zum Ausschluss vom Praktikum führen.

Die Lösungen sind im Praktikum zu präsentieren und direkt danach per moodle abzugeben. Die Bearbeitungsdauer beträgt 2 Praktikumstermine.

Einleitung

In diesem Aufgabenblatt werden Sie Schaltnetze mit verschiedenen Funktionen in Hades aufbauen und Simulieren. Sie sollen dafür Ihre Kenntnisse im Schaltnetzentwurf verwenden. Die hier betrachteten Schaltnetze treten üblicher Weise als Bauteile komplexerer Systeme auf.

Hinweis: Die Bearbeitungsdauer ist so bemessen, dass erwartet wird, dass Sie zum 2. Termin die vorbereitenden Arbeiten wie Schaltnetzminimierung, welche Hades nicht erfordern, bereits vorbereitet haben.

Aufgaben


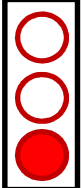
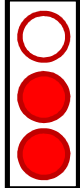
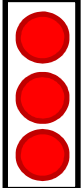
2-1. Ein 1-Bit 4:1 Multiplexer

- Implementieren Sie einen 4:1 Mux analog dem in der Vorlesung behandelten 2:1 Mux in Hades auf *eine* Ihnen geeignet erscheinende Art (als Kaskade von bestehenden SN oder als zweistufiges SN über strukturierten SN-Entwurf).
- Ergänzen Sie das Schaltnetz um geeignete Komponenten für Eingabe und Ausgabe und testen Sie seine korrekte Funktion.
- Demonstrieren und erläutern Sie ihre Lösung im Praktikum. Es ist keine weitere Dokumentation im Bericht notwendig.

2-2. Codeumsetzer: Thermometerskala

Für Anzeigen wird in vielen Geräten eine sog. Thermometerdarstellung verwendet. Dabei wird der Wert durch einen mehr oder weniger weit ausgefüllten Balken visualisiert. Eine einfache Implementierung hiervon ist eine Reihe von Leuchtdioden, von denen je nach Eingabewert mehr oder weniger eingeschaltet werden.

Eine lineare Thermometeranzeige mit 3 Dioden für den Wertebereich 0-3 hat folgende Anzeige:

Wert	00b	01b	10b	11b
Ausgangs-codierung	000b	001	011b	111b
Anzeige				

Ihre Aufgabe ist es, einen Codeumsetzer zu implementieren, der eine gegebene 4-Bit Zahl im Binärsystem in eine Anzeige auf der 6-Bit Thermometerskala umsetzt.

Die Eigenschaften des Codeumsetzters:

- Die Dioden werden durch die Signale $Y=(y_5, \dots, y_0)$ angesteuert.
- Die Binärzahl (Eingangsvektor) ist gegeben durch $X=(x_3, \dots, x_0)$.
- Die Anzeige soll nicht wie oben linear sein, sondern wie folgt abbilden:

X	0	1,2,3,4,5	6	7,8	9	A,B,C,D,E	F
Y	000000b	000001b	000011b	000111b	001111b	011111b	111111b

Bitte bearbeiten Sie zur Lösung dieser Aufgabe folgende Teilaufgaben:

- Geben Sie die Boolesche Funktion an, die den Codeumsetzer formal beschreibt.
- Für jede der Teilfunktionen f_i der y_i geben Sie bitte $ON(f_i)$ an.
- Minimieren sie die Schaltfunktionen der f_i mit einem Ihnen sinnvoll erscheinenden Verfahren und geben sie die Minimalpolymome für y_i an.
- Implementieren Sie den Codeumsetzer in Hades, ergänzen Sie sinnvolle Komponenten für Ein- und Ausgabe.
- Überprüfen Sie Ihre Implementierung auf Korrektheit.

2-3. Folgeprimzahl

Als Übung für die Implementierung partieller Boolesche Funktionen soll ein SN entworfen werden, dass für eine Primzahl, die in vorzeichenloser Binärdarstellung mit drei Bit darstellbar ist, die jeweils nächste (binär dreistellige) Primzahl ausgibt. Die Ausgabe für die Sieben soll die Zwei sein. Für eine Eingabe, die keine Primzahl ist, ist die Ausgabe undefiniert.

- Geben Sie die partielle Boolesche Funktion an, diese Aufgabe löst.
- Für jede der Teilfunktionen f_i der y_i geben Sie bitte $ON(f_i)$ und $DC(f_i)$ an.
- Minimieren sie die Schaltfunktionen der f_i mit einem Ihnen sinnvoll erscheinenden Verfahren und geben sie die Minimalpolymome für y_i an.
- Implementieren Sie die Funktion mit einem Schaltnetz in Hades, ergänzen Sie sinnvolle Komponenten für Ein- und Ausgabe.
- Überprüfen Sie Ihre Implementierung auf Korrektheit.

Abgabe / Präsentation im Praktikum

Alle Aufgaben sind im Praktikum (i.a. an einem der Laborrechner) zu präsentieren. Die Aufgaben eines Aufgabenzettels sollen in der angegebenen Reihenfolge bearbeitet und einzeln präsentiert werden. Gelöste Aufgaben *sollen* erst präsentiert werden und dann die nächste Aufgabe bearbeitet werden. Die Bearbeitung eines neuen Aufgabenzettels *darf* erst begonnen werden, wenn der vorhergehende abgegeben und präsentiert ist.

Die Lösungen der Aufgaben sind zu dokumentieren. D.h. alle erstellten Simulationsschaltungen sind mit sprechenden Namen zu versehen. Wenn vorbereitende Arbeiten (z.B. das Ermitteln einer Schaltfunktion oder ähnlicher Entwurfsschritte) zur Aufgabenstellung gehören, sind diese schriftlich knapp festzuhalten und elektronisch abzugeben. Dabei sind ggf. die erstellten Hades-Simulation mit dem Dateinamen zu referenzieren. Handschriftliche eingescannte Lösungen sind erlaubt.

Die erstellten Lösungen müssen auf den Laborrechnern fehlerfrei laufen. Falls Sie Lösungen auf eigenen Laptops/Notebooks erstellt haben, müssen diese trotzdem auf den Laborrechnern fehlerfrei laufen. Präsentationen auf dem eigenen Laptop/Notebook erfolgen auf eigene Gefahr.

Hochladen in moodle

Die präsentierten Lösungen sind unverändert gruppenweise in moodle hochzuladen. Je Gruppe bitte nur eine Abgabe gemäß der unten angegebenen Form.

- Die Hades-Schaltungen sind *am selben Kalendertag* hochzuladen, an dem die letzte Aufgabe des Aufgabenblattes im Praktikum präsentiert wurde.
- Der komplette Praktikumsbericht (d.h. inklusive der zuvor bereits abgegeben Hades-Dateien) zu einem abgeschlossenen Aufgabenblatt ist *spätestens am darauffolgenden Kalendertag* abzugeben. Erfolgt dies am selben Kalendertag, darf die erste Abgabe entfallen.

Die Abgaben sollen in folgender Form geschehen:

- Je Abgabe eine Archivdatei (.zip oder .rar) mit folgendem Inhalt:
 - Alle Dateien, die Teil der Lösung sind mit selbsterklärenden Namen. Ggf. bitte Verzeichnisse verwenden, die die Lösungen von Teilaufgaben voneinander trennen.
 - Bei umfangreicheren Abgaben muss eine Datei `readme.txt` im Wurzelverzeichnis des Archivs vorhanden sein, die ggf. notwendige Hinweise zur Verwendung der Lösung enthält.
 - Als Dokumentformate für Berichte sind nur .txt, .doc, .docx, .pdf und .jpg zulässig.
 - Eine Textdatei mit Namen `gruppe.txt` im Wurzelverzeichnis des Archivs, die die Namen der Gruppenmitglieder mit Matrikelnummer zeilenweise auflistet.
 - Die Archivdatei muss einen Dateinamen haben, der eindeutig das Fach, die Aufgabe und die Gruppe enthält. Folgendes Format ist zu verwenden:
SS2015-RS_Aufg<x>_<Name>_<Vorname>.zip (bzw. .rar). Ein Beispiel:
SS2015-RS_Aufg2_Blaurock_Ole.zip.
Dabei ist es egal von welchem Gruppenmitglied der Name verwendet wird.
- Pro Gruppe ist nur eine Lösung in moodle hochzuladen.
 - Sollte eine Gruppe mehrere Lösungen abgeben, wird eine nach dem Zufallsprinzip ausgewählt und die anderen Lösungen ignoriert.
- Lösungen, die nicht den formalen Vorgaben entsprechen, werden nicht berücksichtigt.
- Nicht vergessen, die Lösungen in moodle als endgültig abgegeben zu markieren.