



7. Übungsblatt

Lösen Sie die folgenden Aufgaben:

- i) Führen Sie *alle* besprochene Änderungen an Ihrer Komponente `Imm` für die Immediate-Werte aus.
- ii) Bauen Sie den Datenpfad der CPU. Verwenden Sie dazu die Architekturdarstellung von Ripes. Im Prinzip erstellen Sie eine Strukturbeschreibung und verwenden die schon erstellten Komponenten.
- iii) Implementieren Sie einen einfachen Kontrollpfad mit Hilfe eines Moore-Automaten wie besprochen. Überlegen Sie sich die notwendigen Zustände für Ihre Architektur (z.B. `Fetch`, `Execute`, `WriteRegs` und `UpdatePC`). Erzeugen Sie Kontrollsignale um R-Type und I-Type Instruktionen¹ ausführen zu können. Entwerfen Sie ein kleines Testprogramm. Ein Register kann gelöscht werden indem man den Inhalt des 0-Registers mit dem Immediate 0 addiert und in das zu löschende Register schreibt (I-Type). Man kann ein Register auch löschen indem man es mit sich selbst XOR-verknüpft (R-Type). Die notwendigen Instruktionen können Sie leicht von Hand erzeugen und ab Speicherstelle 0 in Ihre Speicherkomponente eintragen. Vermutlich haben Sie diese als Array von Words implementiert, dann kann man das Programm einfach initialen Wert des Arrays schreiben verwenden.

```
-- Init RAM
signal ram : ram_t := (
    0 => "0000000000000000000000000000000100",
    1 => "0000000000000000000000000000000011",
    others => "10101010101010101010101010101010");
```

¹<https://github.com/jameslzhu/riscv-card/blob/master/riscv-card.pdf>

Wenn Sie Ihr Programm ab Speicherstelle 0 ablegen, dann muss natürlich auch der PC nach einem Reset den Wert 0 enthalten.

Besprechung und Abnahme am 14. Dezember 2023