

5. Übungsblatt

Lösen Sie die folgenden Aufgaben:

- i) Führen Sie evtl. besprochene Änderungen an Ihrer ALU aus. Verbessern Sie unbedingt die Testbench! Jede Funktion Ihrer ALU (addieren, subtrahieren, etc.) sollte mit mindestens 20 (möglichst zufällig gewählten) Werten getestet werden. Bereiten Sie *diesmal* Ihre Verbesserungen so vor, dass sie vorgeführt und abgenommen werden können.
- ii) Machen Sie Ihre ALU generisch. Implementieren Sie eine 4-Bit breite Variante und testen Sie die Funktionalität mit den Schiebeschaltern und LEDs auf Ihrem FPGA-Board. Lassen Sie sich eine schematische Darstellung Ihrer ALU generieren und testen Sie, ob Ihre ALU garantiert keine Latches enthält.
- iii) Implementieren Sie den Programmzähler PC unserer CPU. Welche Breite muss der PC in Ihrer RISC-V-Implementierung haben? Können Sie eine Begründung angeben? Dabei macht es Sinn die drei Teilkomponenten des Programmzählers aus Abbildung 1 in einer Komponente zu implementieren.
Hinweis: Denken Sie nochmal darüber nach, wie Sie Ihren Speicher adressieren (Wort/Byte). Welche Auswirkungen hat das für Ihre Implementierung des Programmzählers?
- iv) Entwerfen Sie eine Testbench für den PC und automatisieren Sie den Buildprozess mit **make** oder einem geeigneten Skript. Verwenden Sie auf jeden Fall auch **report** und **assert** für Ihre Testbench, um die Tests automatisieren zu können! Bereiten Sie alles so vor, dass die Funktion Ihrer Testbench leicht vorgeführt werden kann.
- v) Implementieren Sie eine Teilfunktionalität des Befehlsdekoders unserer CPU. Zum aktuellen Zeitpunkt müssen nur R-Type Befehle dekodiert werden. Diese haben immer die Form $R_i = R_j \text{ op } R_k$. Bestimmen Sie dazu die Indices der beteiligten Register aus der Instruktion. Es bietet sich an, dass Sie Ihre Implementation so aufbauen, dass wir diese später leicht durch andere Befehlsformate erweitern können (z.B. Sprünge).
Hinweis: Spätestens nun sollten Sie sich auf die Suche nach geeigneter Literatur über RISC-V begeben (Buch!).

Besprechung und Abnahme am 3. Dezember 2025

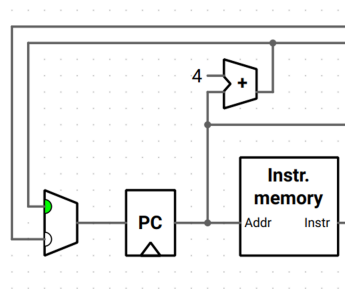


Abbildung 1: Ausschnitt Ripes - Single Cycle CPU