

# Architektur

## Übungen Computerarchitektur

## 1 | Architektur

### 1.1 Stack-Architektur

Auswertung des Ausdrucks  $\frac{a+bc}{a+dc-e}$  unter Verwendung eines stack-basierten Prozessors.

- Schreibe Pseudo-Code der Berechnung.
- Wie viele direkte und indirekte Speicherreferenzen sind bei einer unendlichen Stackgröße ?
- Wie viele direkte und indirekte Speicherreferenzen sind bei einer Stackgröße von 2 erforderlich?

*arc/stack-01*

### 1.2 Stack-Architektur

Auswertung des Ausdrucks  $\frac{(a+b)^2}{\pi} * (a + b + c)$  unter Verwendung eines stack-basierten Prozessors.

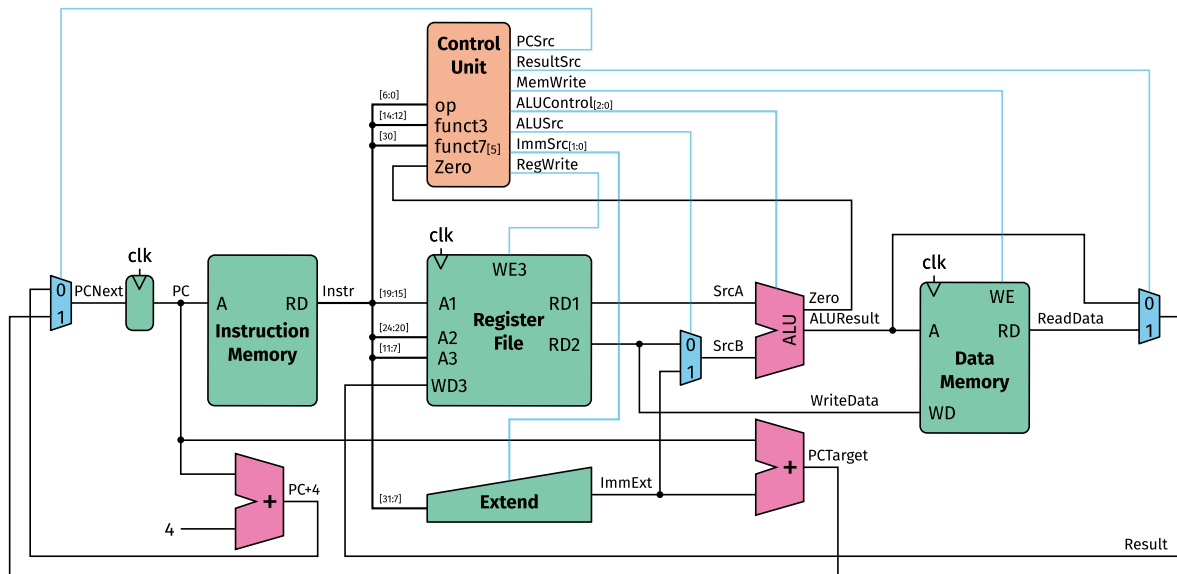
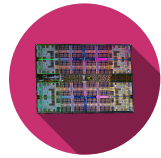
- Schreibe Pseudo-Code der Berechnung.
- Wie viele direkte und indirekte Speicherreferenzen sind bei einer Stackgröße von 4 erforderlich?
- Wie viele direkte und indirekte Speicherreferenzen sind bei einer Stackgröße von 3 erforderlich?
- Wie viele direkte und indirekte Speicherreferenzen sind bei einer Stackgröße von 2 erforderlich?

*arc/stack-02*

## 2 | Single-Cycle RISC-V

### 2.1 Single-Cycle-Prozessorbetrieb

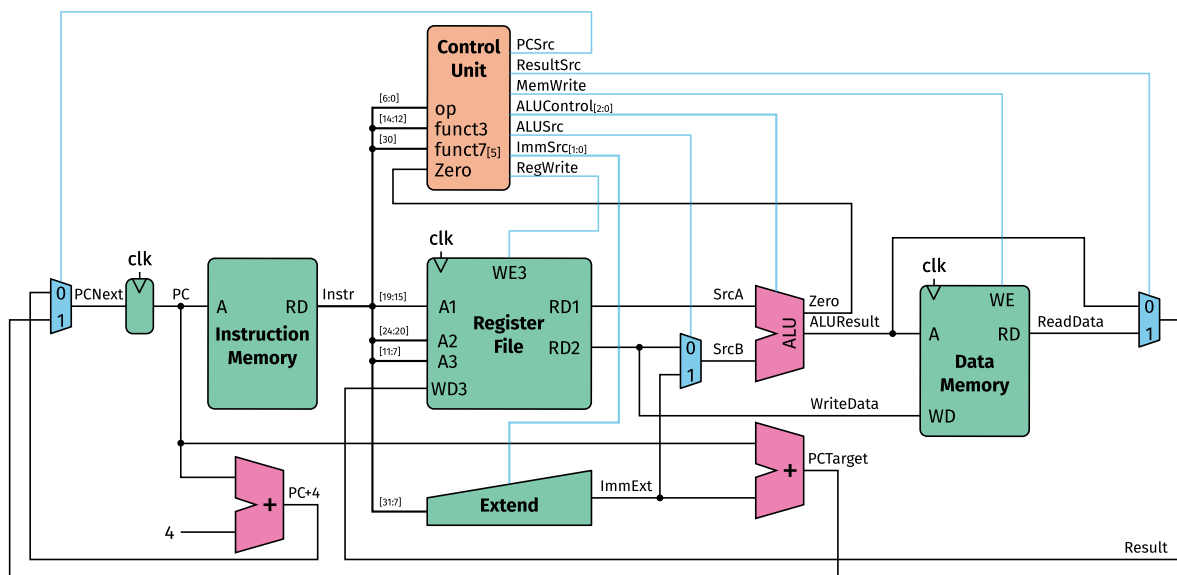
Bestimmen Sie die Werte der Steuersignale und die Teile des Datenpfads, die bei der Ausführung einer **and**-Anweisung verwendet werden. Zeichnen Sie direkt auf dem Bild das Innenleben des Prozessors.



arc/scr-01

## 2.2 Einzelzyklus mit Anweisung **jal** verlängern

Zeigen Sie, wie der gegebene RISC-V-Einzyklusprozessor so geändert werden kann, dass er den Sprung- und Verknüpfungsbefehl **jal** unterstützt. **jal** schreibt **PC+4** nach **rd** und ändert den PC auf die Sprungzieladresse **PC+imm**.



arc/scr-02

## 2.3 Einzelzyklus-Prozessorleistung

Ein mit einem 7-nm-CMOS-Fertigungsprozess hergestellter Einzyklus Prozessor weist die folgenden Zeitmerkmale auf.



Element	Parameter	Delay(ps)
Register clk-to-Q	$t_{pcq}$	40
Register Setup	$t_{setup}$	50
Multiplexer	$t_{mux}$	30
AND-OR Gate	$t_{AND\_OR}$	20
ALU	$t_{ALU}$	120
Decoder (Control Unit)	$t_{dec}$	25
Extend Unit	$t_{ext}$	35
Memory Read	$t_{mem}$	200
Register File Read	$t_{RFread}$	100
Register File Setup	$t_{RFSetup}$	60

Das Programm für den SPECINT2000-Benchmark enthält 100 Milliarden Anweisungen. Berechnen Sie die Berechnungszeit des Benchmarks für diesen Einzyklus Prozessor.

*arc/scr-03*

### 3 | Multi-Cycle RISC-V

#### 3.1 Mehrzyklus-Prozessorleistung

Das Programm für den SPECINT2000-Benchmark besteht aus ungefähr 25% loads, 10% stores, 11% branches, 2% jumps und 52% R- oder I-Typ ALU-Befehlen.

Ermitteln Sie den durchschnittlichen CPI für diesen Benchmark für den von uns entwickelten Multi-Cycle-Prozessor.

*arc/mcr-01*

#### 3.2 Multi-Zyklus-Prozessorleistung

Ein mit einem 7-nm-CMOS-Fertigungsprozess hergestellter Multi-Zyklus Prozessor weist die folgenden Zeitmerkmale auf.

Element	Parameter	Delay(ps)
Register clk-to-Q	$t_{pc}$	40
Register Setup	$t_{setup}$	50
Multiplexer	$t_{mux}$	30
AND-OR Gate	$t_{AND\_OR}$	20
ALU	$t_{ALU}$	120
Decoder (Control Unit)	$t_{dec}$	25
Extend Unit	$t_{ext}$	35
Memory Read	$t_{mem}$	200
Register File Read	$t_{RFread}$	100
Register File Setup	$t_{RFSetup}$	60



Das Programm für den SPECINT2000-Benchmark enthält 100 Milliarden Anweisungen. Nutzen sie die  $CPI_{avg}$  der vorherigen Aufgabe.

Berechnen Sie die Berechnungszeit des Benchmarks für diesen Multi-Zyklus Prozessor.

*arc/mcr-02*