

Design eines Redis Cache  
Uni-Projekt: Chip Design & Verilog



Philipp Hecht

Luca Pinnekamp

Luca Schmid

17. Februar 2026

## **Zusammenfassung**

Implement a Redis-inspired cache that works in conjunction with the CPU. The goal is to create a fast cache for storing key-value pairs. Basic CRUD (Create, Read, Update, Delete) implementation

## **Inhaltsverzeichnis**

<b>1 Einleitung / Idee</b>	<b>2</b>
<b>2 Learnings</b>	<b>2</b>
<b>3 Ausblick / Zusätzliche Funktionalitäten</b>	<b>2</b>
<b>4 Vivado Setup (Mac)</b>	<b>2</b>

## 1 Einleitung / Idee

Die ursprüngliche Idee dieses Projekts ist der Entwurf und die Implementierung eines kompakten, synthetisierbaren Key-Value-Stores, inspiriert von Redis, auf RTL-Ebene (für FPGAs oder ASICs).

Das Ziel ist es, grundlegende Speicheroperationen direkt in Hardware abzubilden, um eine hohe Performance und geringe Latenz zu erreichen. Die ursprünglich geplanten Kernfunktionen sind:

- **Einfügen von Schlüssel-Wert Paaren (Key-Value Insertion)**
- **Abrufen von Werten anhand von Schlüsseln (Value Retrieval)**
- **Löschen von Werten anhand von Schlüsseln (Key Deletion)**
- **Auflisten von Schlüsseln (Key Listing)**
- **Automatische Ablaufzeit (TTL - Time-to-Live)**

Die Motivation liegt darin, die Effizienz von Key-Value-Speichern durch Hardwarebeschleunigung zu untersuchen und eine Schnittstelle bereitzustellen, die ähnlich wie Software-Caches funktioniert, aber die Vorteile dedizierter Hardware nutzt.

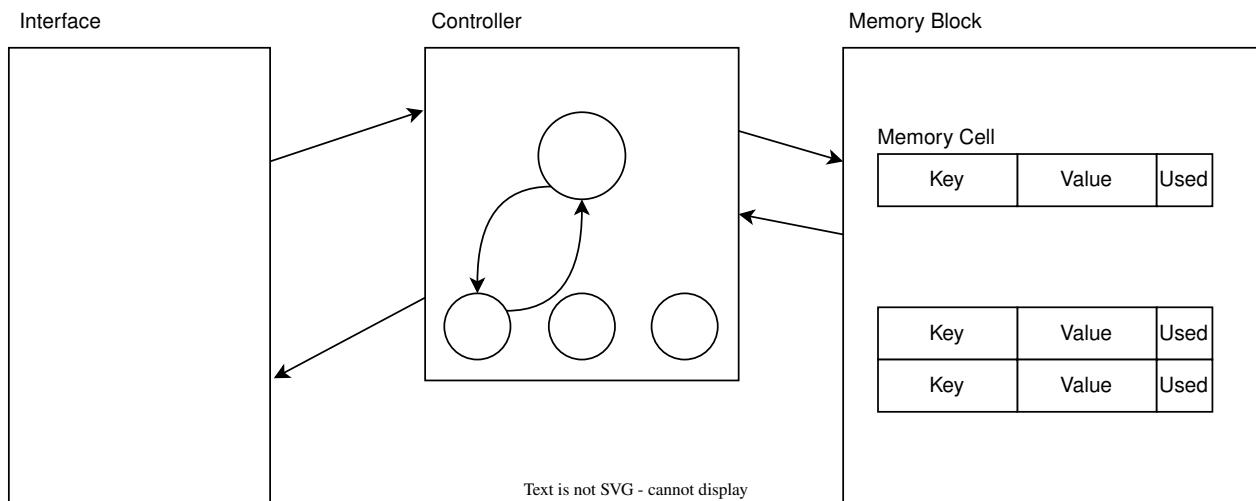


Abbildung 1: Speicherblöcke

## 2 Learnings

## 3 Ausblick / Zusätzliche Funktionalitäten

## 4 Vivado Setup (Mac)