

Kolloquium zum Master-Praktikum INF_MA_PR
**Implementierung eines
Omega-Netzwerkes in VHDL**

Matthias Häsing <matthias.haasing@tu-dresden.de>

Dresden, 20.11.2014



Gliederung

- 1. Motivation / Aufgabenstellung**
- 2. Theorie des Omega-Netzwerks**
- 3. Designentscheidungen für Entwurf**
- 4. Optimierungen des Entwurfs**
- 5. Ergebnisse**
- 6. Zusammenfassung und Ausblick**

1 Motivation

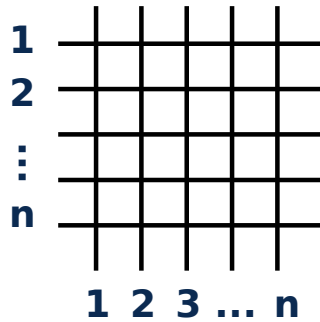
N-zu-N-Verbindungen sind teuer

Gleich, ob in Rechnernetzen, Parallelrechnern oder on-chip-Netzwerken - die Kommunikation zwischen Knotenpunkten

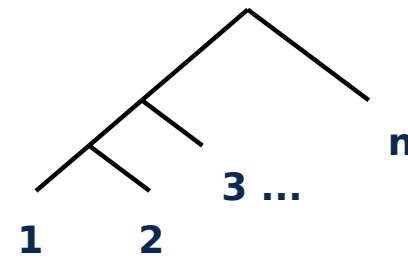
- entwickelt sich für gewöhnlich zu einem **Flaschenhals**,
- ist **teuer** in Bezug auf **Kommunikationszeit oder**
- **teuer** bezüglich **Technologie, Chipfläche, etc.**

Effiziente Vernetzung hat gravierende Bedeutung für **Systemleistung**.

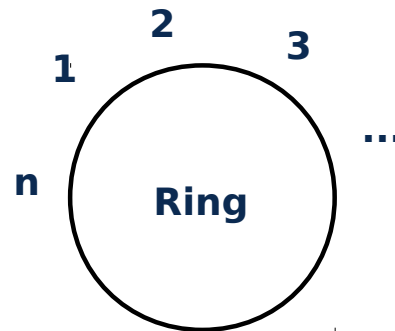
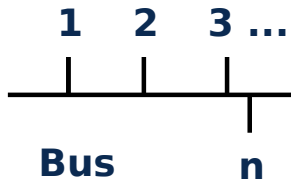
Geläufige Netzwerktopologien



Kreuzschienenverteiler



Bäume



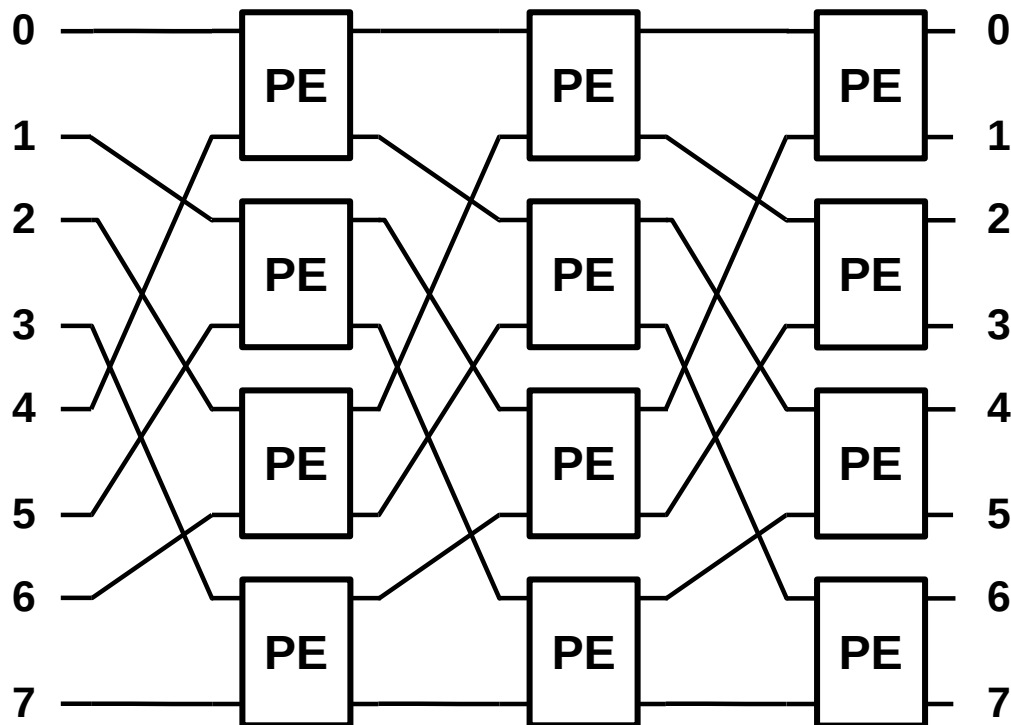
Omega-Netzwerk ?

Aufgabenstellung

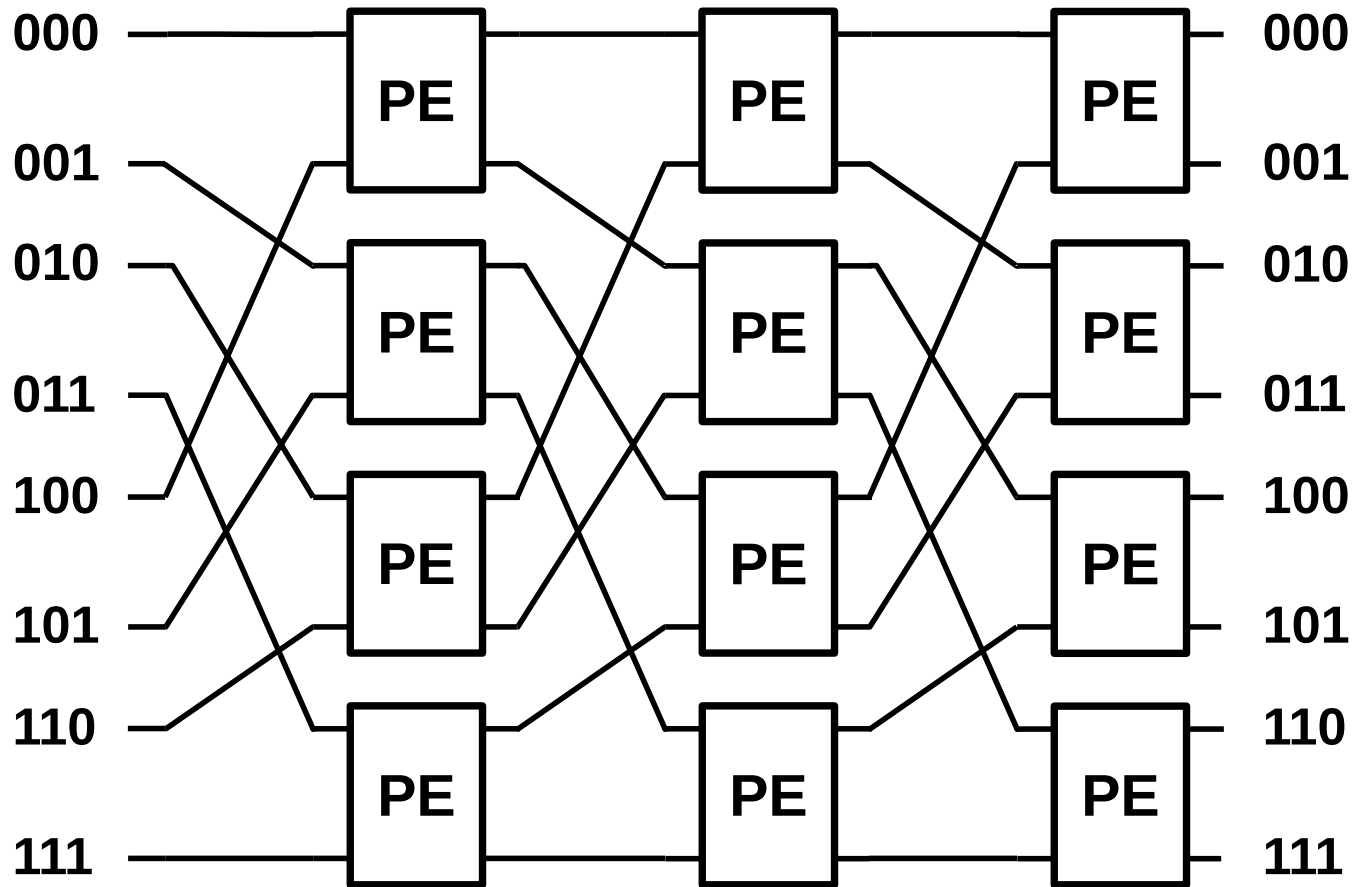
Implementierung eines Omega-Netzwerkes in VHDL

- Suche von Optimierungsmöglichkeiten
- Suche von Einsatzmöglichkeiten

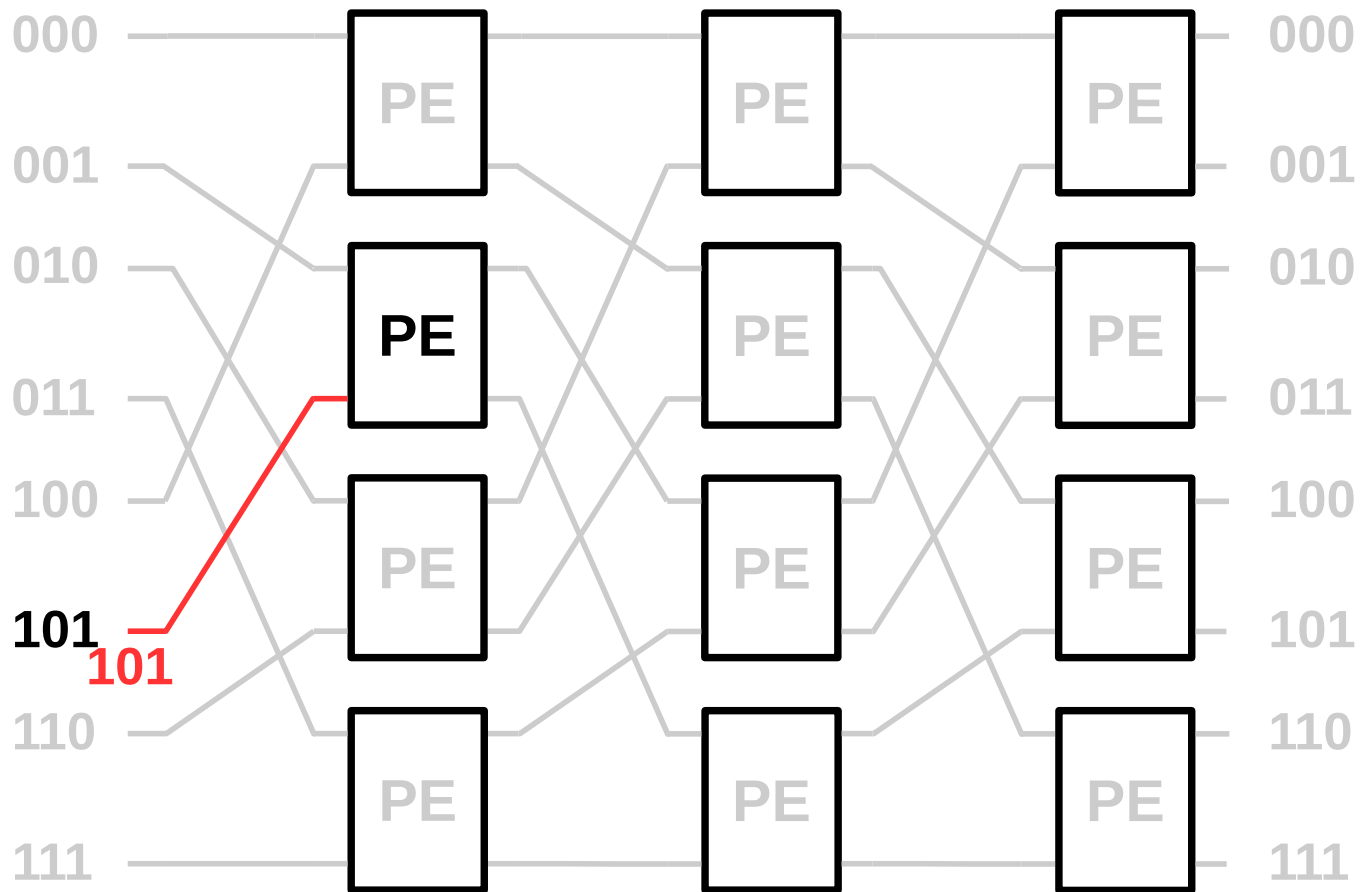
2 Das Omega-Netzwerk



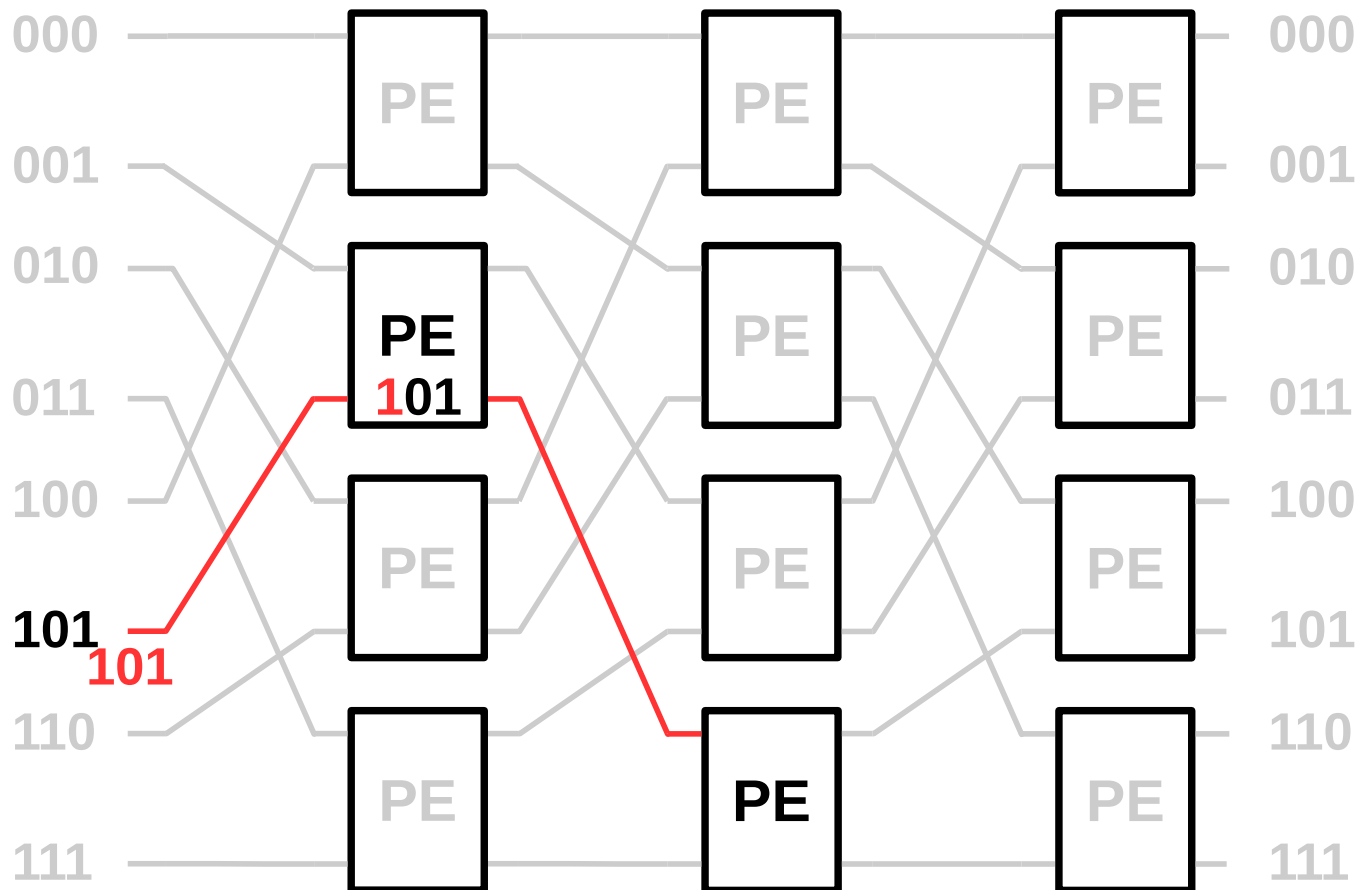
8x8 Omega-Netzwerk



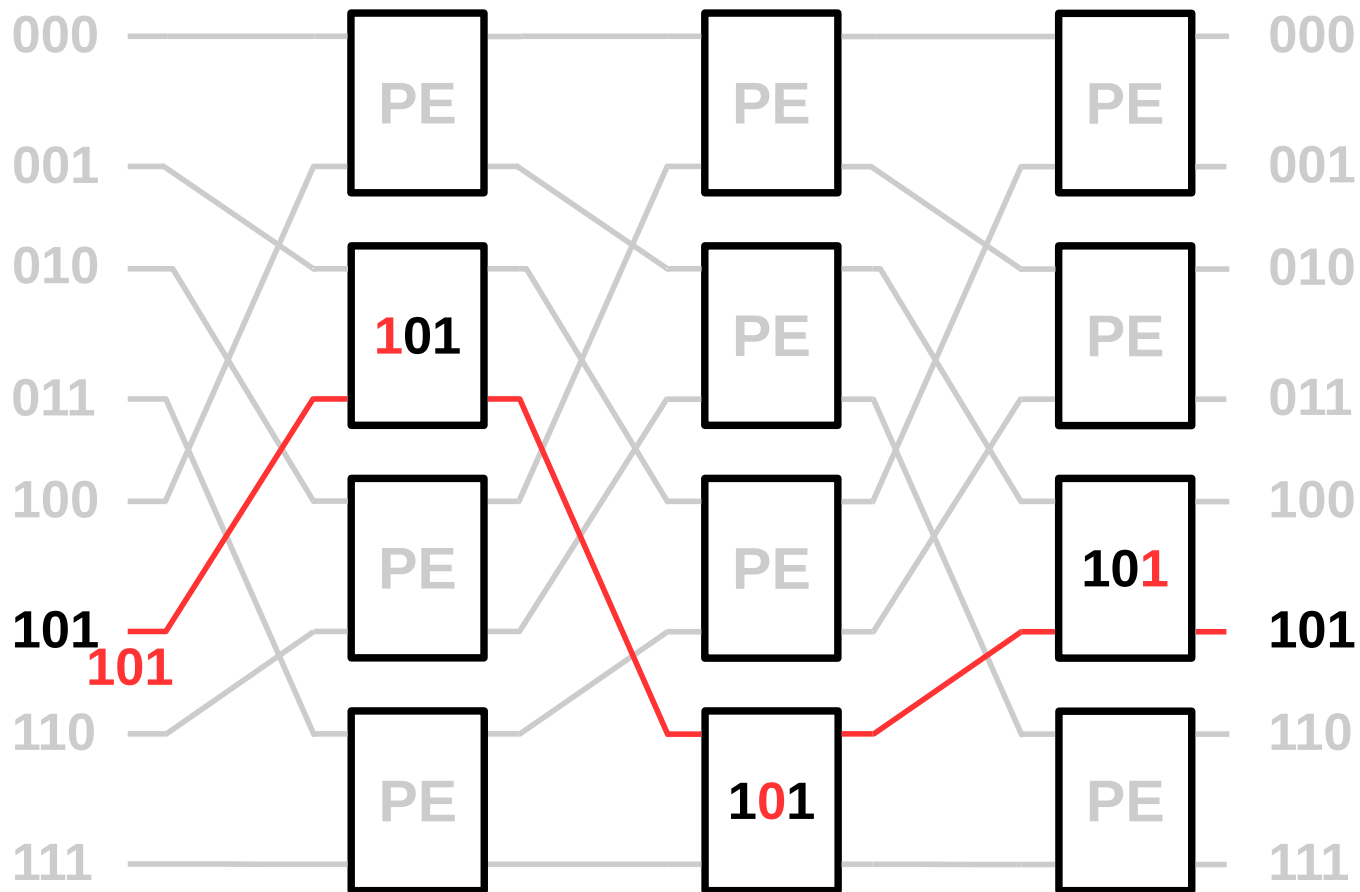
Beispiel 1: 101 → 101



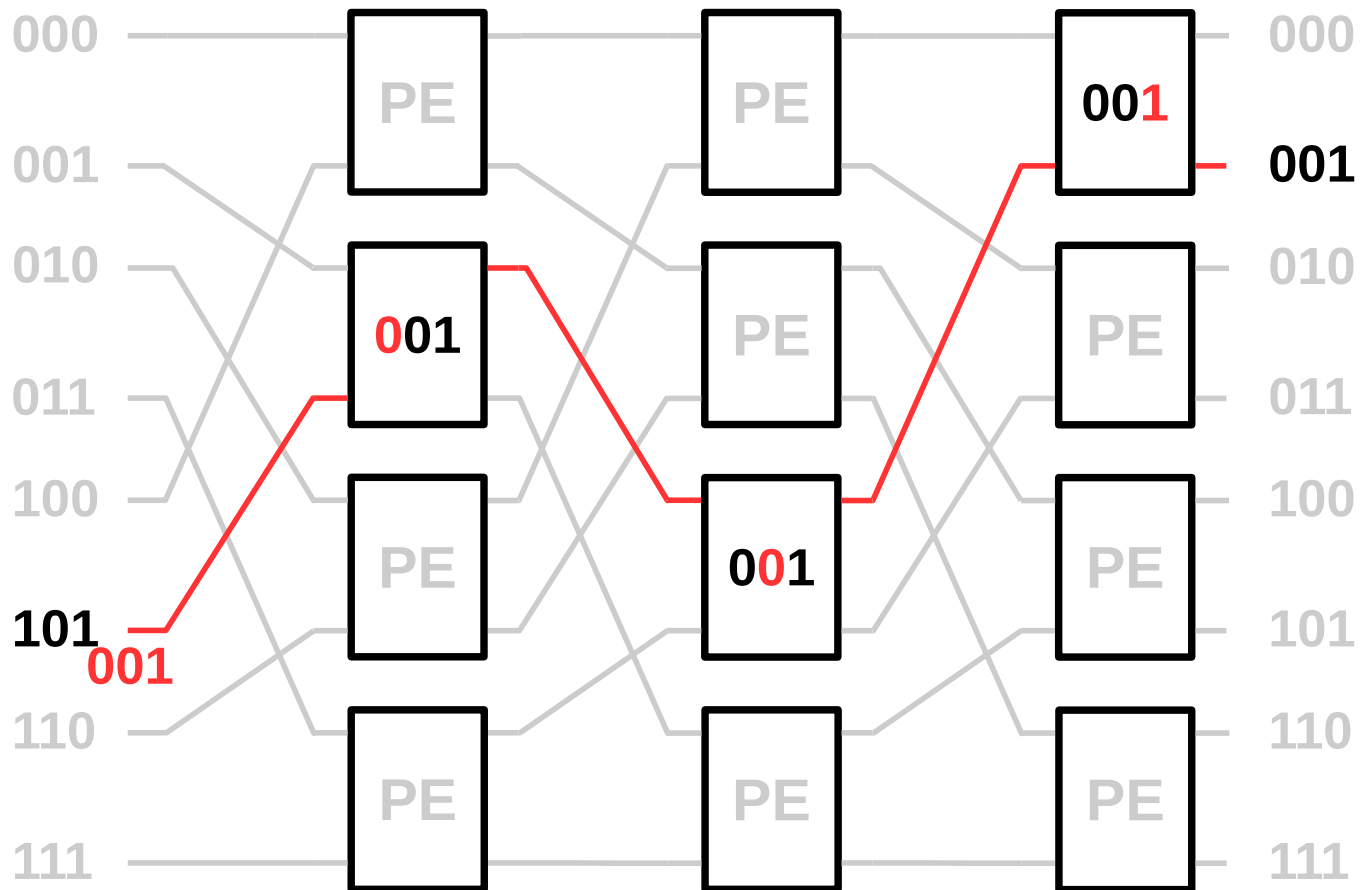
Beispiel 1: 101 → 101



Beispiel 1: 101 → 101

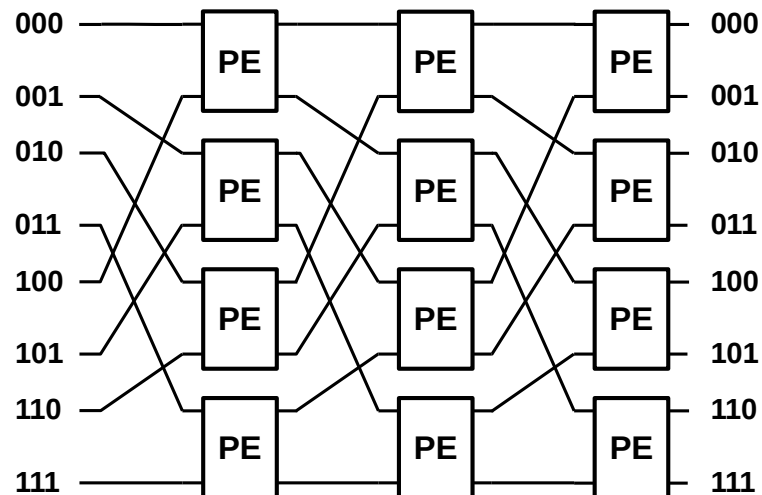


Beispiel 2: 101 → 001



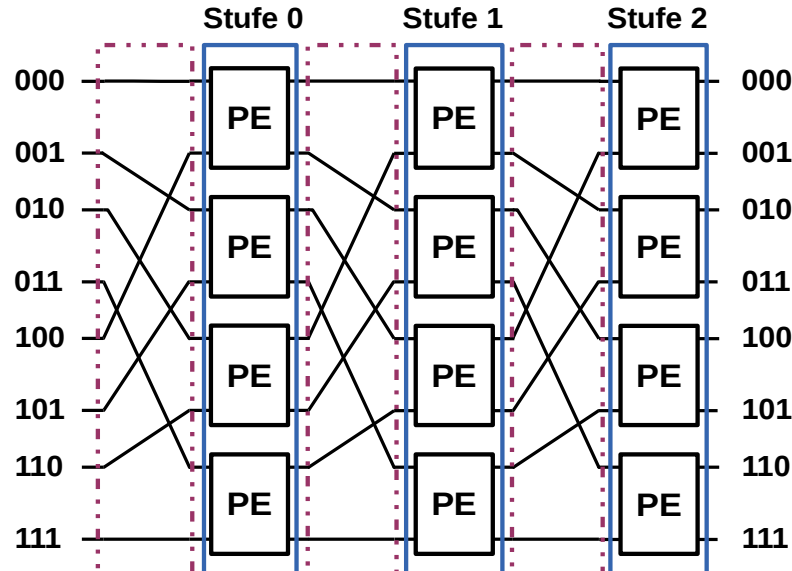
Haupteigenschaften des Omega-Netzwerkes

- Gleichförmiger Aufbau in n identischen Stufen
- Verdrahtung durch „Perfect Shuffle“
- Einfaches Routing anhand Zieladresse
- Logarithmische Tiefe: n Stufen bedienen 2^n Eingänge

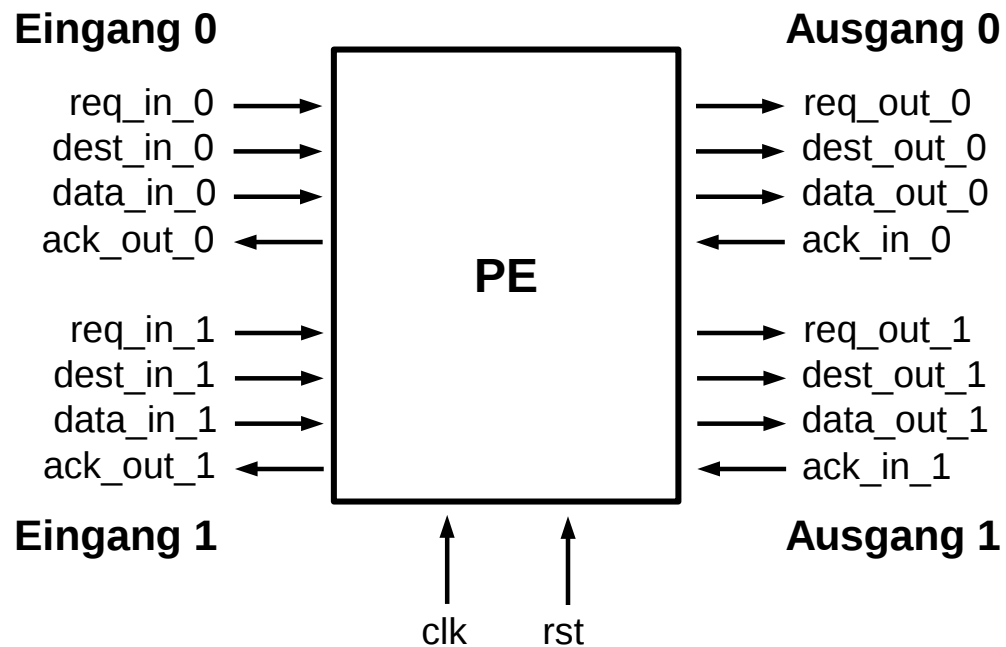


3 Designentscheidungen

- Hierarchischer Aufbau des Designs in n Stufen mit je 2^{n-1} PEs
- Perfect Shuffle-Verdrahtung als (generisches) Modul

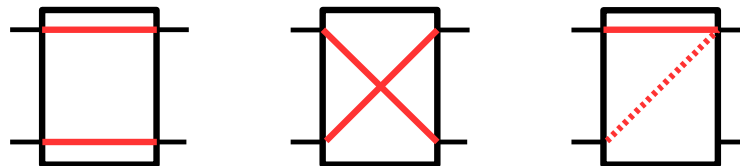


- Schnittstelle mit Request und Acknowledge
- Bei konkurrierenden Anfragen: Round-Robin-Arbitrierung

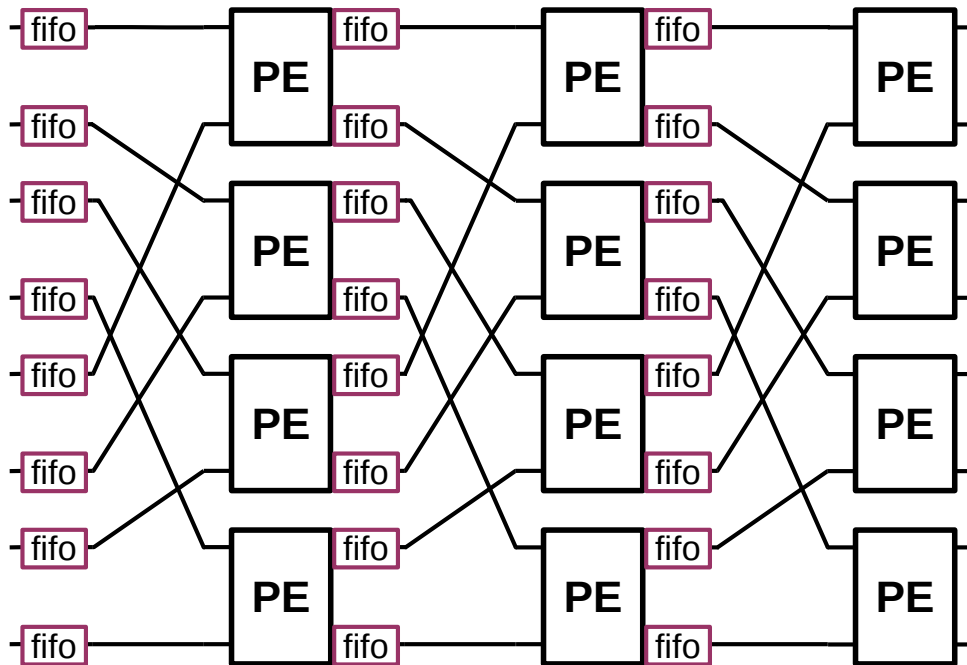


4 Optimierungen des Entwurfs

- Durchgehend generisches Design bezüglich n und Datenwortbreite
- Eliminierung von Wartetakten: Eingehender Request nach n Takten am Ausgang anliegend (bei freiem Pfad)
- Gleichzeitige Nutzung beider Kommunikationspfade im PE, falls keine Kollision vorliegt



- Pufferung durch minimale FIFO (fifo_glue)
- Frei halten des 2. Kommunikationspfads im PE, falls Ausgang belegt
→ Dadurch erhöhter Durchsatz möglich



5 Ergebnisse

Test des Netzwerks sowie „Messungen“ durch Simulation.

Für Test eigene Sender- und Empfangsmodule, die an jeweils einem Ein- bzw. Ausgang Nachrichten absenden bzw. erwarten.

- Nachrichten werden korrekt übertragen
- Nach Optimierung entspricht Latenz für einzelne Nachrichten, bzw. bei freiem Pfad dem Erwartungswert (n Takte, bis Request propagiert ist für ungepuffertes Netzwerk)
- Im ungünstigsten Fall (n Sender an einen Empfänger) werden die Nachrichten fair serialisiert

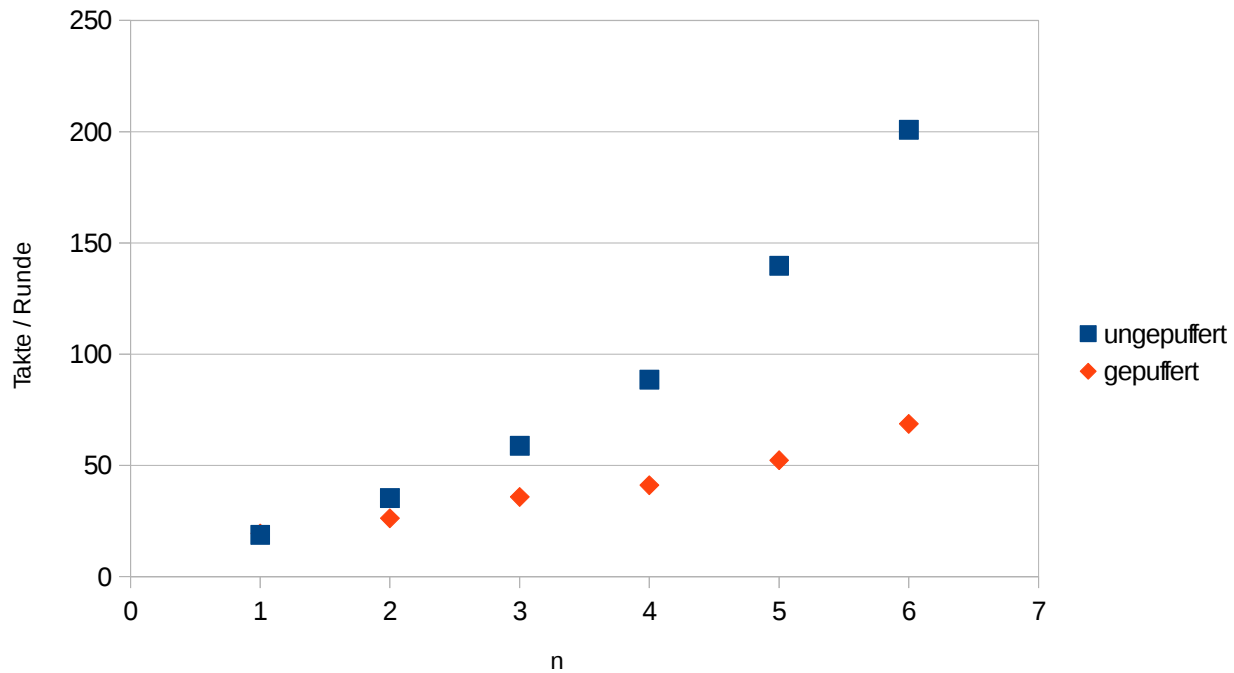
Die Generierung von realistischen Testdaten bzw. Kommunikationsmustern ist aufwändig bzw. abhängig von Einsatzgebiet.

Daher Versuch, Leistungsfähigkeit und Skalierung mit „einfachen“ Testdaten zu untersuchen:

- Sender senden fortlaufend Nachrichten und inkrementieren die Empfängeradresse.
- Durch geeignete Startwerte werden Konflikte erzeugt
- Auswertung der Übertragungsdauer (in Takten) für feste Anzahl von Nachrichten bzw. **Runden von Nachrichten** (Eine Nachricht pro Eingang).

Laufzeit für ausgelastetes Netzwerk

2^n Sender senden an 2^n Empfänger; **1 Runde = 2^n Nachrichten**



Ressourcenbedarf (geschätzt)

Virtex 6 ML605; Datenwortbreite 32 bit;

Slice Registers		n	1	2	3
(93120)	ungepuffert		32	118	340
	gepuffert		42	160	500
Slice Luts					
(46560)	ungepuffert		56	212	569
	gepuffert		63	94	594

6 Zusammenfassung und Ausblick

- Eigenes Design mittels Simulation getestet.
- Generisches Design
 - Datenwortbreite
 - Tiefe des Netzwerks (n)
- Pufferung bringt für Testdaten großen Laufzeitvorteil

Zu untersuchen: Einfluss **größerer Puffer** und **asymmetrischer Puffer**
Davor allerdings Entscheidung nötig, ob Pufferung fest eingebaut.
Dann Verzicht auf Acknowledge-Signal möglich.

Ausblick:

- Anpassung auf konkretes Anwendungsgebiet (z.B. Zugriff auch verteilten Speicher)
- Dann Messungen mit realistischen Testdaten möglich
- Vergleich mit anderen Verbindungsnetzwerken bezüglich
 - Übertragungsgeschwindigkeit und
 - Ressourcenverbrauch

Literatur

Lawrie, D.H., "Access and Alignment of Data in an Array Processor," Computers, IEEE Transactions on , vol.C-24, no.12, pp.1145,1155, Dec. 1975



»Wissen schafft Brücken.«