

AESS

*Architekturen eingebetteter
Systeme für die Signalverarbeitung*



Fraunhofer
HHI



Harte Fakten – FPGAs aus der Architekturperspektive

Workshop "FPGAs in (High Performance) Computing... Quo Vadis?"

13.3.2026

12. März 1984

Wie Alles begann

- Ross H. Freeman (Gründer von Xilinx) meldet ein Patent an, was den Aufbau rekonfigurierbarer Logik beschreibt.
- **Patent US 4870302: Configurable electrical circuit having configurable logic elements and configurable interconnects.**
- Angemeldet am **12. März 1984**, veröffentlicht am 26. September 1989

Zwei Varianten sog. L.E. / Logic Elements

- Universeller Logikblock
- LUT-basierte Architektur

- Konfigurierbares Verbindungsnetzwerk



U.S. Patent 4,870,302

Example of a 3x3 Logic Element CLA with 12 I/O pads & 3 types of L. E.'s

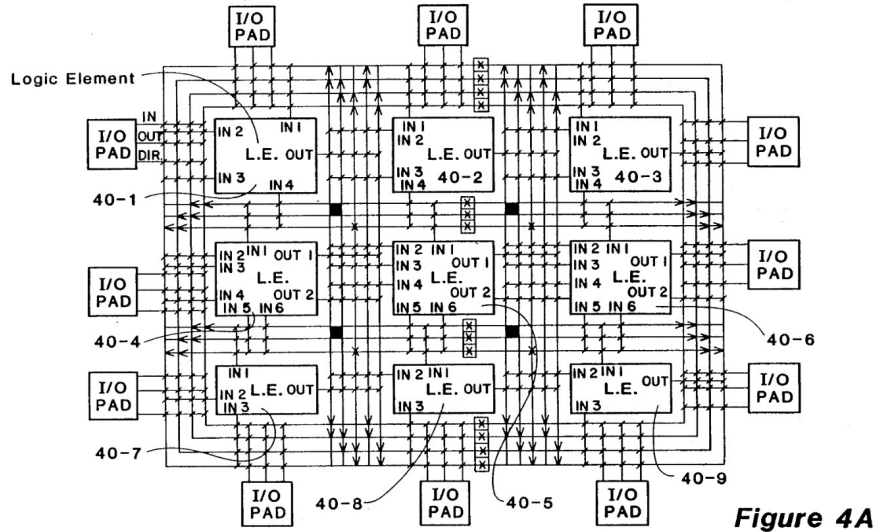


Figure 4A

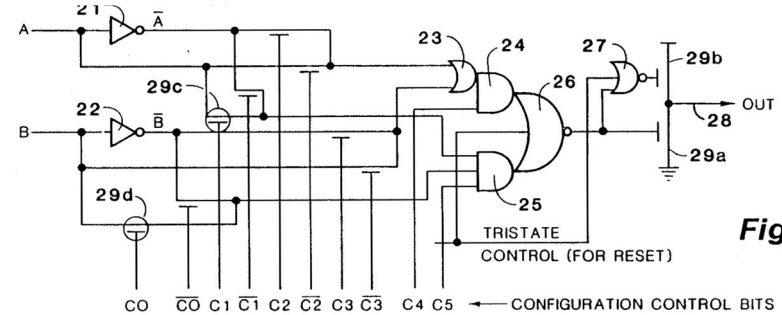


Figure 2

U.S. Patent

Sep. 26, 1989

Sheet 4 of 15

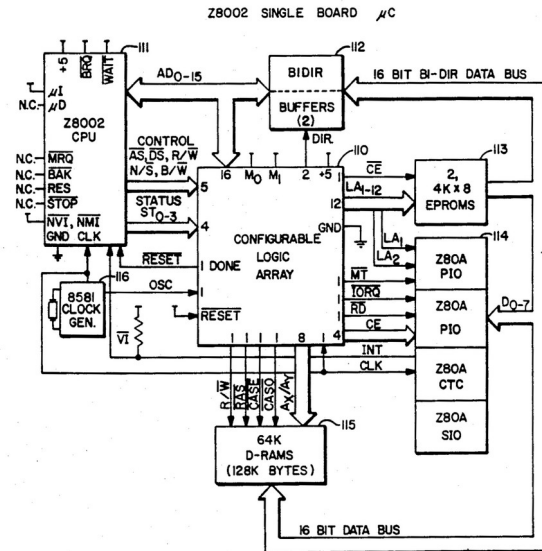
4,870,302

U.S. Patent

Sep. 26, 1989

Sheet 15 of 15

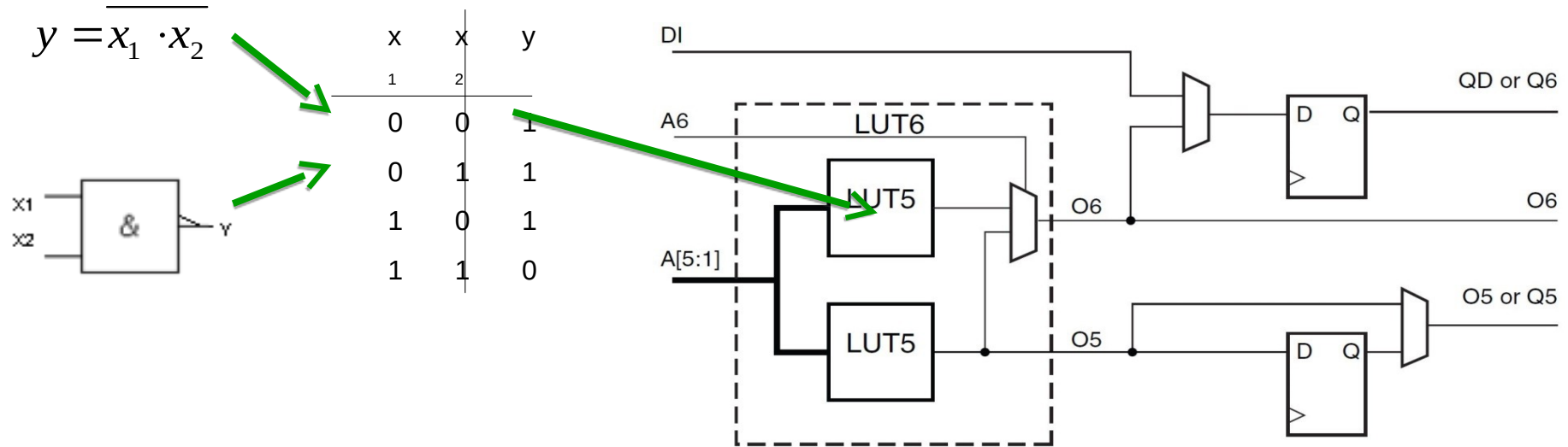
4,870,302



FPGA Funktionsweise

Was ist eine LUT ?

FPGAs bilden kombinatorische Logik über Look-Up-Tables ab (Speicherblöcke)



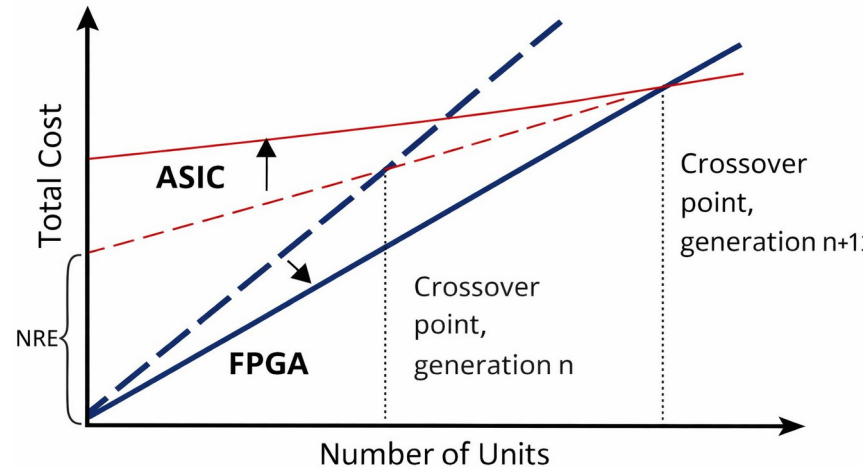
Ein FPGA ist damit ein Array von Look Up Tables

Bildquelle: Xilinx Spartan Family Reference Manual

Follow the Money

Kommerzialisierung der FPGA Technologie

- Hohe Fertigungskosten für ASICs machten programmierbare Logik attraktiv, wie z.B. PAL, PLD, etc.
- XC2064 von Xilinx mit **64 CLBs / LE** -
erstes kommerziell verfügbares FPGA, vorgestellt im Jahr 1985.

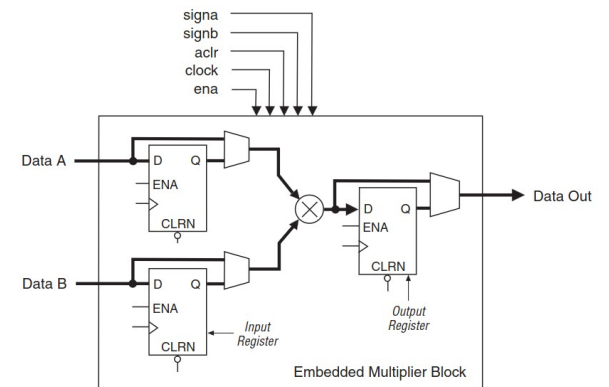


Historisch

42 Jahre oder wie es weiter ging

- Steigende Transistorzahlen (Moore's Law)
- Integration von **Block RAM, DSP-Slices / Multiplizierer, PLLs**
- FPGAs werden produktive Systemkomponenten
- Einsatz in Telekommunikation, Netzwerktechnik, Embedded Systems
- Vom reinen Logikbaustein zur heterogenen Rechenplattform
- Integration kompletter Prozessoren
- Partielle Rekonfiguration
- **High-Speed I/O**
 - Transceiver IPs, Speichercontroller, PCIe
- FPGAs als SoC Plattform

Figure 4-2. Multiplier Block Architecture



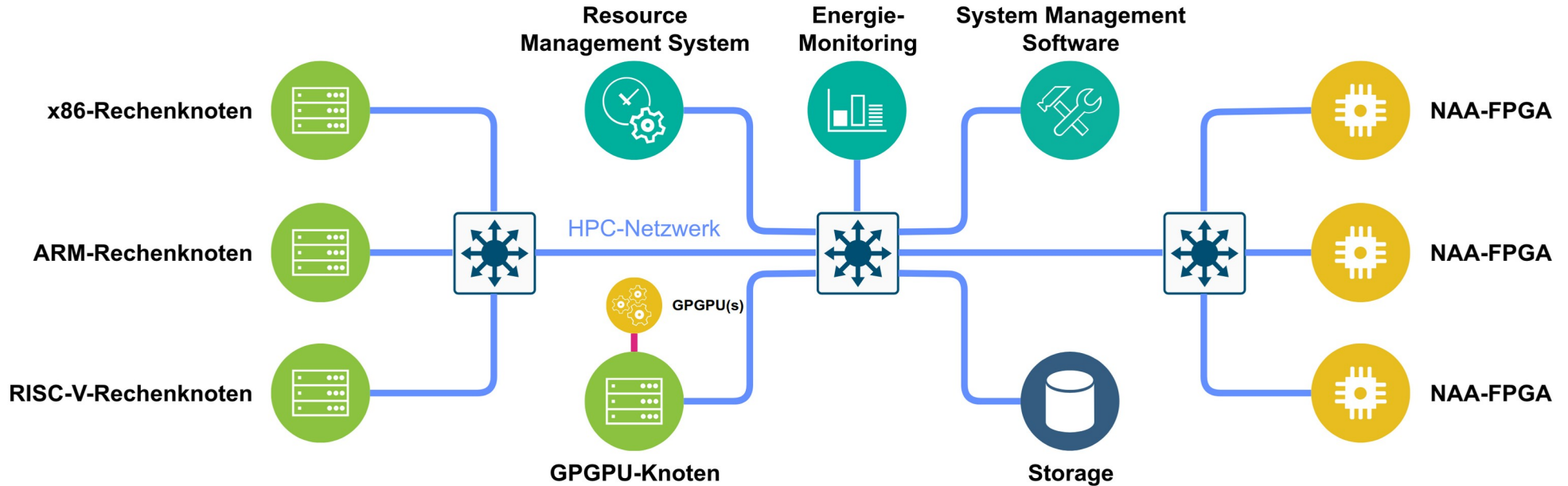
State of the Art / Agilix 7 FPGA 10 nm

- Logic Elements (LE) 2,692.760
- Adaptive Logic Modules (ALM) **912.800**
- Adaptive Logic Module (ALM) Registers **3.651.200**
- Fabric and I/O Phase-Locked Loops (PLLs) 28
- Maximum Embedded Memory **287 Mb**
- Digital Signal Processing (DSP) Blocks **8.528**
- Digital Signal Processing (DSP) Format
Fixed Point (hard IP), Floating Point (hard IP), **Multiply, Multiply and Accumulate, Variable Precision**
- Hard Processor System (HPS) **Quad-core 64 bit Arm* Cortex*-A53**
- External Memory Interfaces (EMIF) DDR4, QDR IV
- Transceiver Protocol Hard IP PCIe Gen4, **10/25/50/100/200/400G Ethernet**

NAAICE

Das NAAICE Projekt

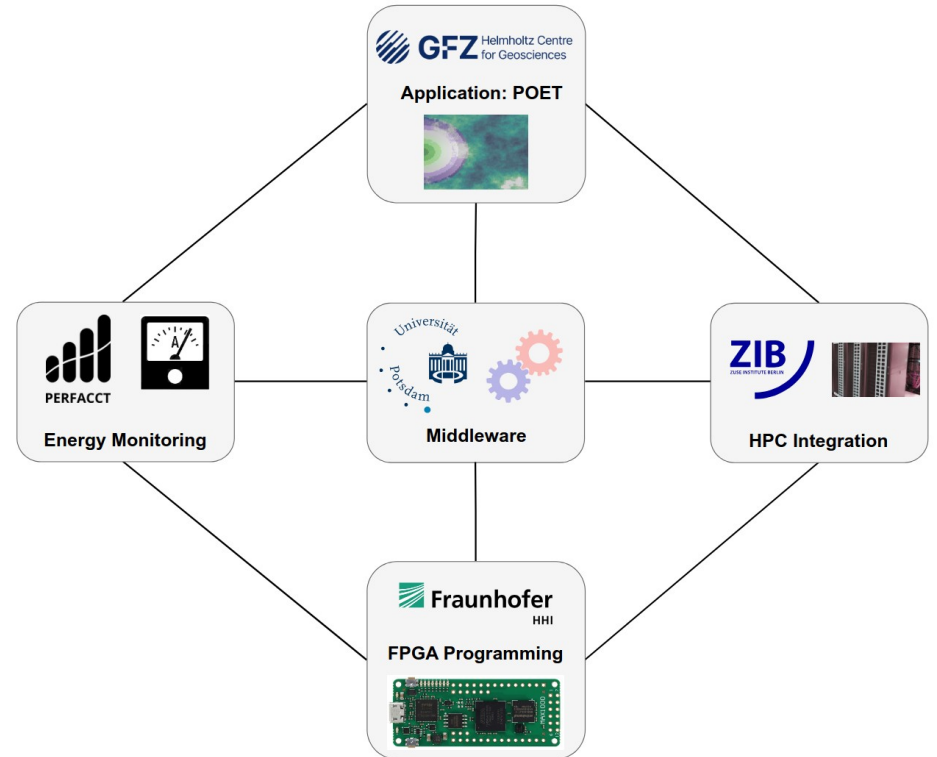
HPC-Rechenzentrum



NAAICE

Das NAAICE Projekt

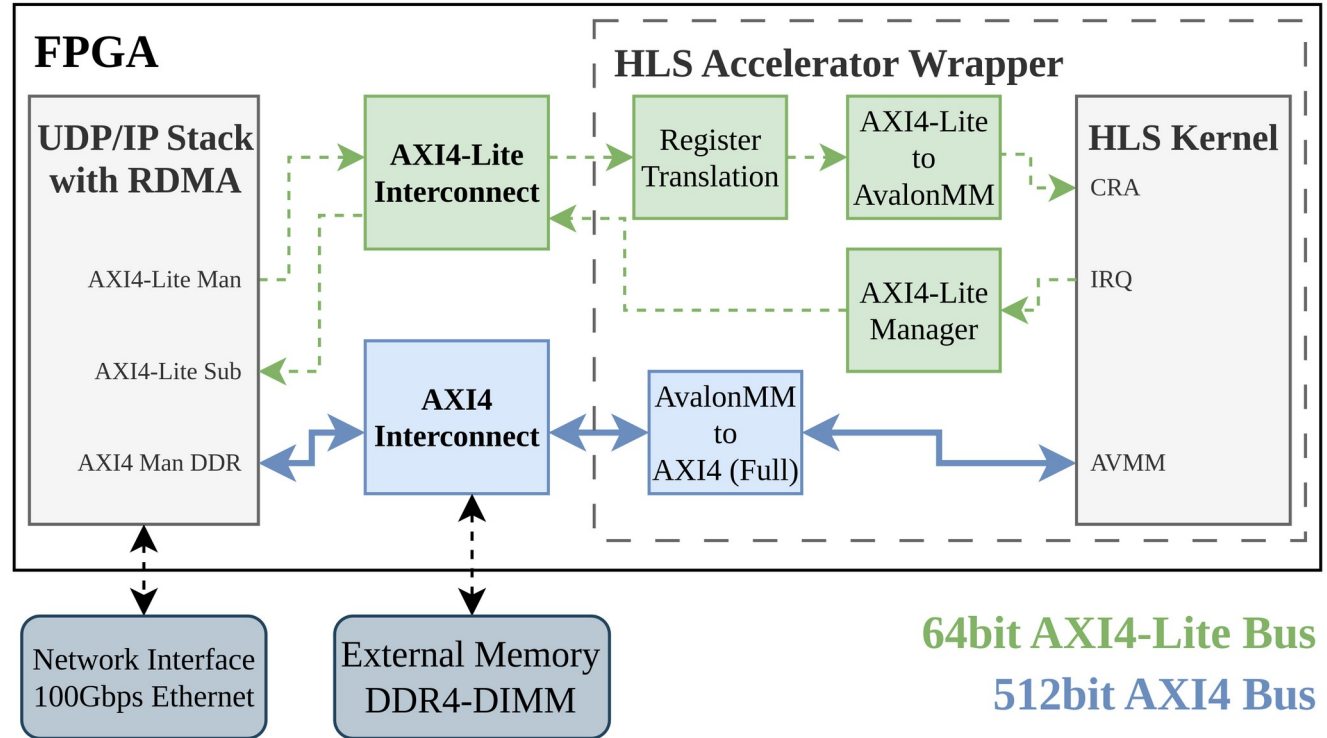
- **Netzanbindung** des FPGAs über das Hochgeschwindigkeitsnetzwerk RoCEv2 (**HHI**)
- Ansteuerung des FPGAs über eine **RPC** basierte **Kommunikationsmiddleware** (**UP**, **ZIB**)
- **Integration** in die bestehende Software-Infrastruktur des **HPC-Rechenzentrums** (**ZIB**)
- Erstellung eines **Profiling-Frameworks** (**PERFACCT**), das auch die Leistungsaufnahme des FPGA-Beschleunigers ermitteln kann
- **Evaluierung** der Konzepte anhand der **Anwendung POET** (**GFZ**, **UP**) aus den Geowissenschaften



NAAICE

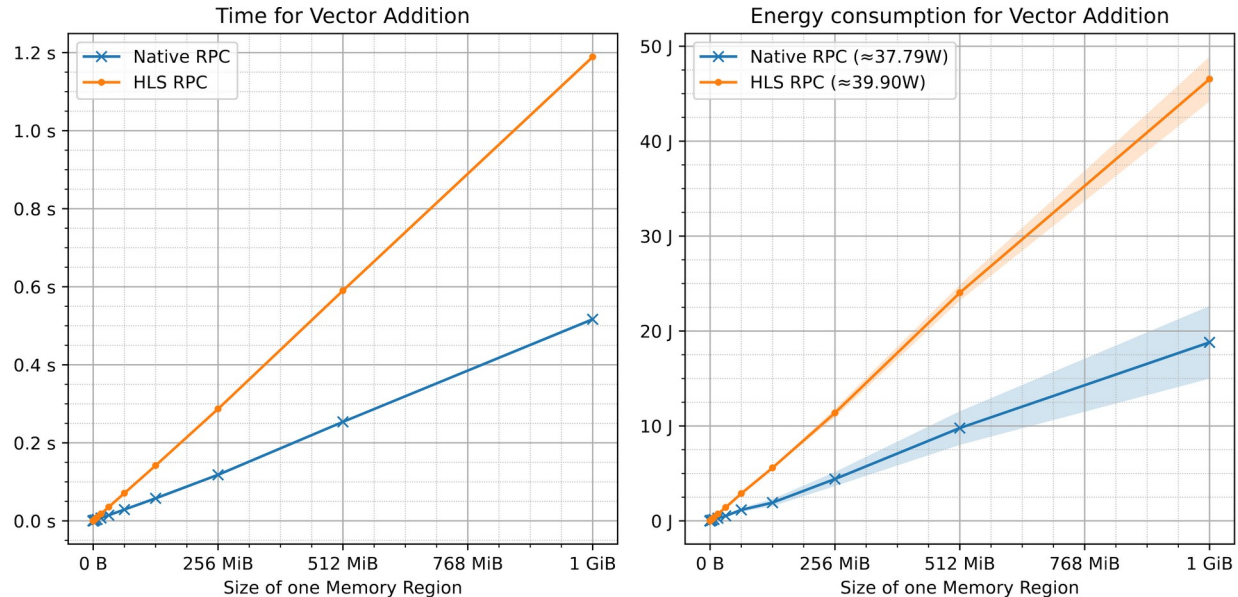
NAAICE Accelerator Konzept

- Nallatech IA-840f Board
- Altera Agilex 7- AGF027
912.800 ALMs / 13.272 BRAM
- DDR4-2660 Memory Interface
- Konfiguration und Setup via Netzwerk
- Powermonitoring inkl. MQTT Kommunikation
- NAA Stack
7.187 ALMs ($\approx 7.4\%$)
224 BRAM ($\approx 1.7\%$)
- 340 Mhz Clock



NAAICE Experiment HLS vs. HDL

- Vektoraddition als einfaches Experiment mittels OneAPI, HLS bzw. HDL Implementierung

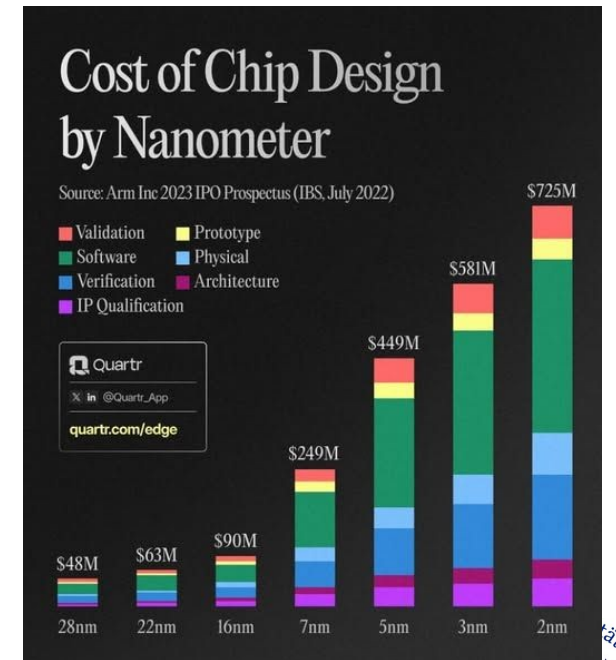


S. Christgau et al., "On the Usability and Energy Efficiency of High-Level Synthesis for FPGA-based Network-Attached Accelerators," 2025 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Milano, Italy, 2025

Fazit FPGAs

Fazit FPGA Technologie

- Mit **steigenden Technologiekosten** wird es auch weiterhin FPGAs als “ASIC Ersatz” geben
- Designflows für FPGAs werden sich daher immer an den Design Flows für ASICs orientieren
- HDLs, wie VHDL oder Verilog und deren Workflows dominieren hier den Markt und werden es weiterhin tun
- Es gibt keinen wirklichen Grund für FPGA Hersteller Nischenmärkte zu bedienen, wenn die Hauptmärkte ertragreich sind



Fazit HPC Kontext

Fazit FPGA Technologie im HPC Einsatz

- HDL-basierte Designflows sind im **HPC-Umfeld für Anwender** inakzeptabel
- Designflows, die zur **Designzeit** das jeweilige **Synthesetool** verwenden, sind eher ungeeignet, da
 - Steigende Laufzeiten mit steigender Komplexität der Designs / FPGAs
 - HDL/HLS “Tuning” erfordert HW Expertenwissen
 - Fehlersuche / Debugging erfordert Spezialwissen
 - **Ergebnis in der Regel nicht vorhersehbar bzw. nicht deterministisch**
- **Kompromisloses Erreichen höchster Durchsätze !!**
- Design Gap hinsichtlich der effizienten “Programmierung” von FPGAs im **HPC Umfeld !**

Quo Vadis

Quo Vadis FPGA im HPC Umfeld

- **Design Space Exploration** / bzw. Performance Modelling Teil eines Workflows
 - **Welche Plattform** ist für mein gegebenes Rechenproblem geeignet ?
 - **Welche Performance** werde ich erreichen vs. GPU vs. CPU ?
- **Overlay Architekturen**
 - Vorteile hinsichtlich Determinismus
 - Ermöglichen Design Space Exploration
 - Design-Zeiten könnten reduziert werden, da eine bestehende Architektur verwendet wird
- **Bibliotheksfunktionen**
 - Höchster Optimierungsgrad / Beste Performance
 - Stetig zu erweitern

Vielen Dank !