Paderborn
Center for
Parallel
Computing

Low-Latency Multi-FPGA Inference using FINN

Author: Bjarne Wintermann
Bjarne.wintermann@uni-paderborn.de

ENERGY EFFICIENT AI
BY DNN APPROXIMATION
FOR FPGAs
https://eki-project.tech

MULTIFPGA + LOW LATENCY

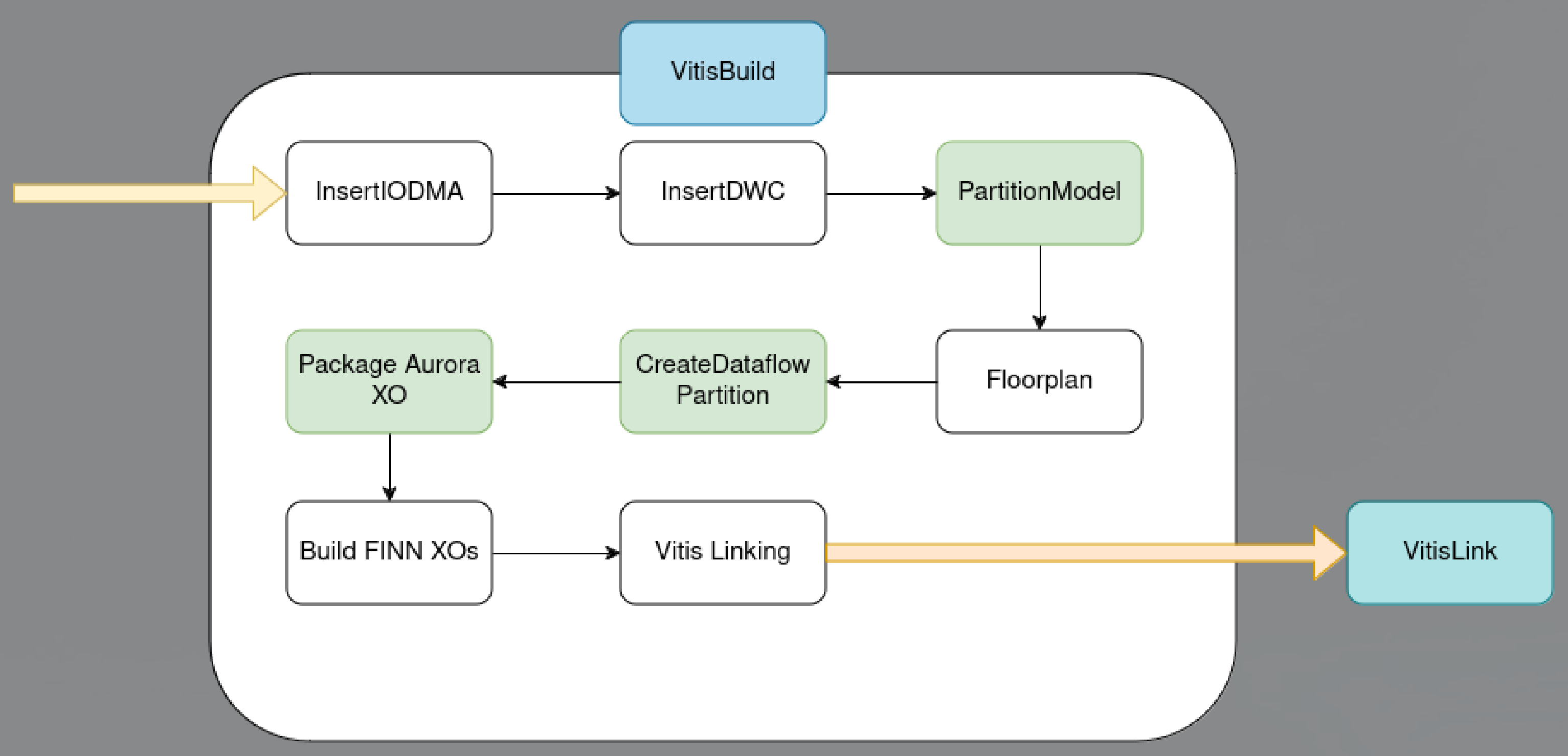
- FINN already enables low latency on single FPGAs
- We want scaling for larger networks
  - Resnet50 as a first benchmark
- Solutions exist for networking – but we want low latency, direct connection

- Solution: Using the Aurora 64B/66B IP Core for direct serial communication
  - Very efficient protocol
  - Very low overhead
  - Direct serial connection instead of routing via for example UDP
  - Openly available from AMD

- Combined with automatic partitioning tooling for optimal distribution of load
  - Avoids congestion during Synthesis and P&R

- Tight integration: Simply pass your partitioning parameters to the default FINN Build process
- FINN does everything else!
  - Scheduling and distribution of bitstreams via SLURM

THE BUILD PROCESS

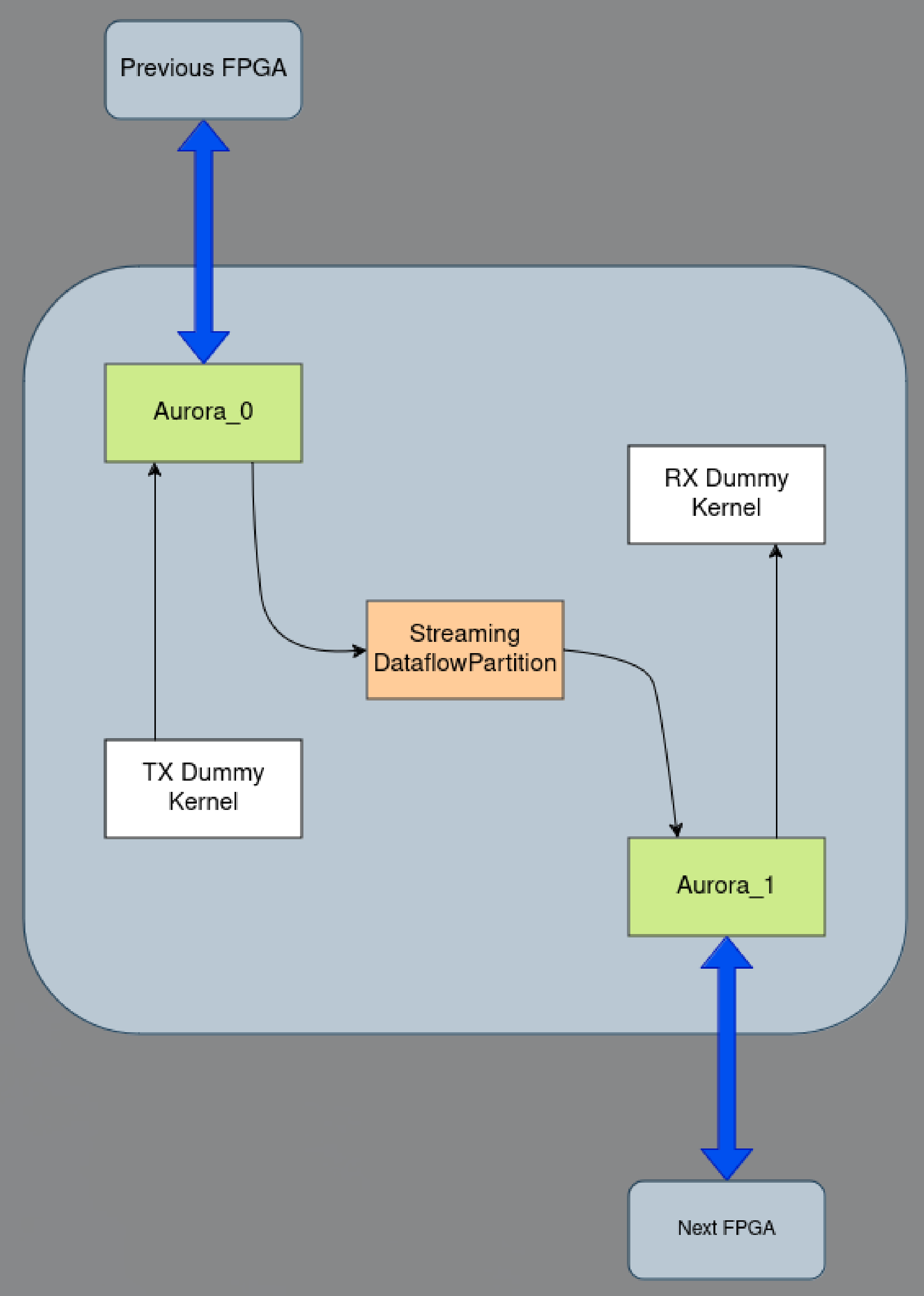


- This is all executed in the Synthesize-Bitfile Build Step of FINN
- Green Transformations are only executed for Multi-FPGA Usage
- This results in one StreamingDataflowPartition per Device, just as in a Single-FPGA Flow

THE HARDWARE

- For the Multi-FPGA Application we use the following hardware:
- 48x Alveo U280 FPGAs, with 2x QSFP 100 GBit each
  - ○ Distributed in 16 nodes with 3 FPGAs each
- Connected via an CALIENT S320 Optical Circuit Switch
  - ○ Programmable via simple scripts at the start of a SLURM Job

THE DESIGN



- This is how one of the interconnected FPGAs looks like
- One Aurora Core for communication with the other FPGA on both directions – this forms a line connection, essentially expanding the Dataflow Paradigm over multiple devices!
- The input and output FPGAs only require one Aurora Core
- The Aurora Cores can either be connected like this to form a line
  - Or you can insert two StreamingDataflowPartitions and connected back using the open Aurora line in the other direction

OUTLOOK

- Multiple topologies
- Customizable partitioning optimization functions
- A custom protocol for encoding multiple layer connections in one serial connection
- Integration with our new high-performance C++ driver

FINN C++ Driver

REFERENCES

- [1] The Aurora 64B/66B IP Core https://www.xilinx.com/products/intellectual-property/aurora64b66b.html

The eki Project is supported by:

Zukunft
Umwelt
Gesellschaft

Bundesministerium
für Umwelt, Naturschutz
und Reaktorsicherheit

PC² – Shaping the Future of HPC