PCIe-Wishbone IP core

Wesley W. Terpstra

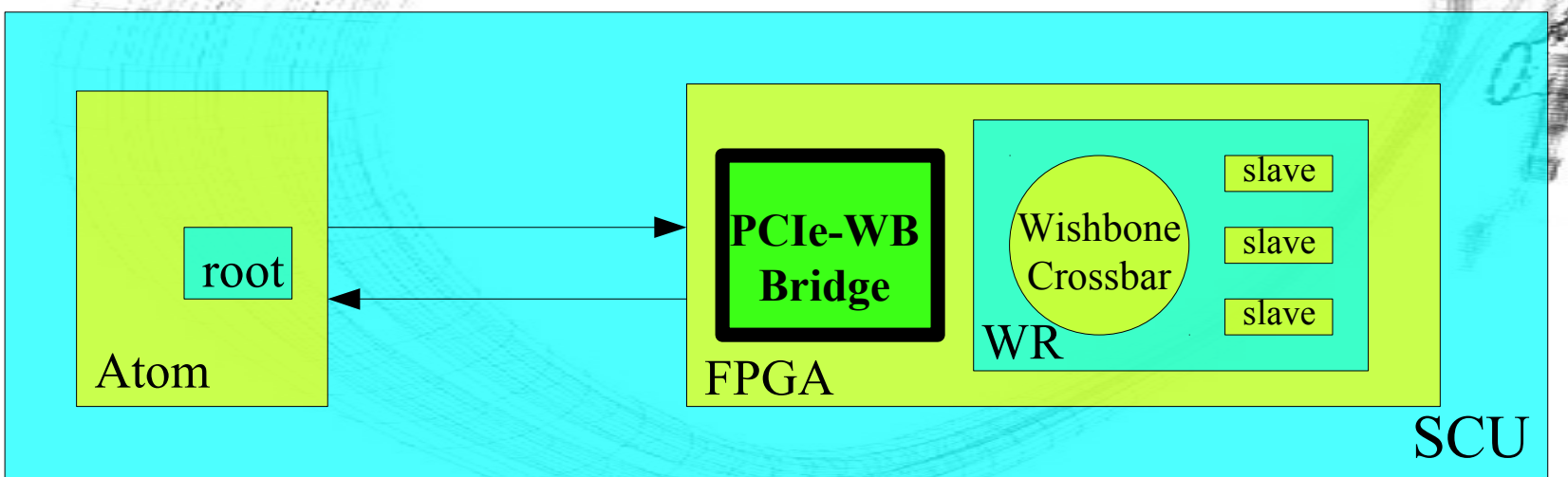
from BEL

http://bel.gsi.de

GSI FAIR

Why we built it

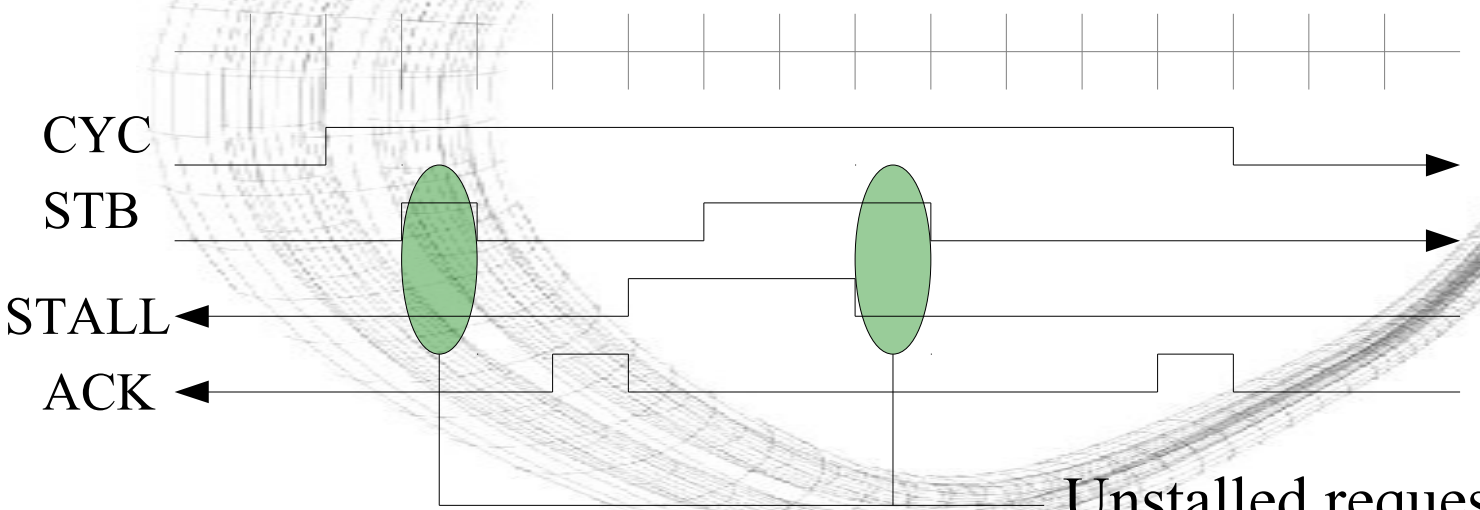
- Wishbone is used in the White Rabbit (WR) project with CERN
- The SCU connects the Atom processor to FPGA via PCIe
- The frontend / FESA group need access to WR registers



What is Wishbone?

- Wishbone: synchronous parallel bus
- Simple master-slave connections only
- In the White Rabbit project: 32-bit @ 125MHz

➔ Maximum throughput: 500 MB/s

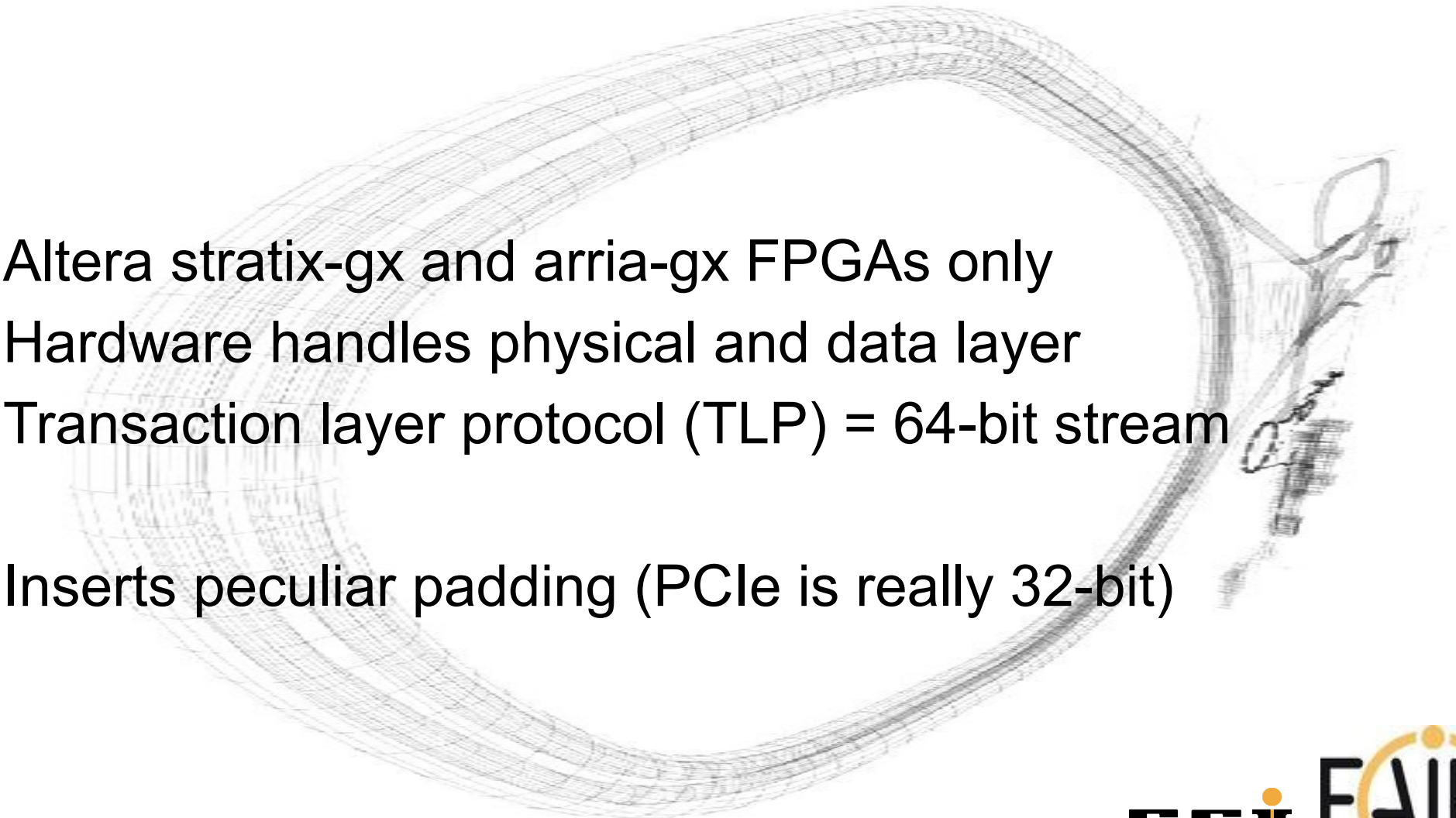


Compared to PCIe

- PCIe: asynchronous serial bus, multiple lanes
- Switched fabric of communicating peers
- v1: 2.5GHz 8b10b v2: 5GHz 8b10b v3: 8Ghz 128b130b
- Our bridge is v1.1 so 250MB/s per lane

- Layered protocol
  - Physical: recovers clock and decodes bitstream
  - Data: flow control, ordering, retransmission
  - Transaction (TLP): read/writes of memory ranges

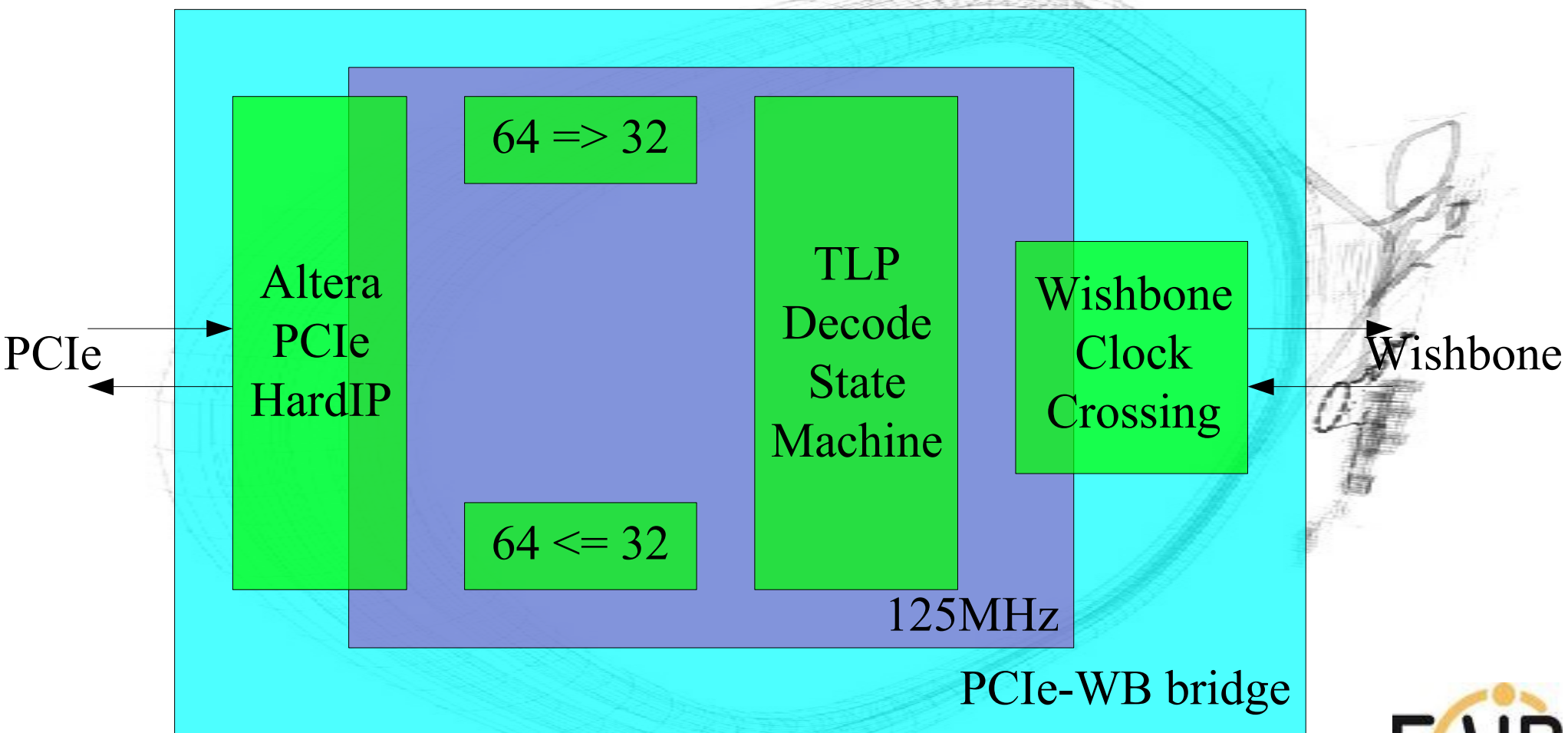
Altera PCIe HardIP core



- Altera stratix-gx and arria-gx FPGAs only
- Hardware handles physical and data layer
- Transaction layer protocol (TLP) = 64-bit stream

- Inserts peculiar padding (PCIe is really 32-bit)

Bridge: TLP decoder to Wishbone



Interface as seen by Atom

- Two BARs (memory mapped regions):
  - BAR0 (128B) = control cycle line and address bits 64-17
  - BAR1 (64kB) = memory mapped access for bits 15-0

- BAR0 provides error status register and SDB address
- Can add additional logic (burst transfer registers?) to BAR0

- Kernel PCIe driver maps WB to Etherbone character device
- Re-uses mature user-space Etherbone library

Limitations of our implementation

PCIe has significant fixed overhead per TLP and ...
Current core is a PCIe slave only and CPUs never burst
To achieve full throughput => become a bursting master

Processes 32-bits at 125MHz – at most 2x v1.1 lanes
=> Restricts maximum throughput to 500MB/s

Master only on PCIe side; WB cannot initiate transfers

Questions?