Unified Communication Framework

Dominic Gaisbauer, Yunpeng Bai, Igor Konorov, Dmytro Levit, Dominik Steffen

d.gaisbauer@tum.de

Technische Universität München

Institute for Hadronic Structure and
Fundamental Symmetries

Padova, June 6th, 2016
Status Quo of DAQ Topologies

- One link for trigger and timing messages
- One link for slow-control messages
- Multiple data links
- Underlying principle is similar in many experiments, e.g. BELLE II, COMPASS, LHC experiments, …

- Why not combine slow control, trigger, timing, and data in a single link?
Unified Communication Framework (UCF)

- Originates from the SODA time distribution system developed for the PANDA experiment
- Single high-speed serial link for data, slow control, trigger, and timing information
- Up to 64 different communication channels (e.g. JTAG, I2C, SPI, …)
- Fixed latency for one channel
- Priority handling for all channels
- Self recoverable after connection losses
- Independent from physical layer
UCF – Example Topologies

- Point-to-Point topology:
  - Single or multiple 1:1 connections
  - Frontend - data concentrator applications, …
  - Bidirectional on all channels
UCF – Example Topologies

- Star-like topology:
  - Single 1:n connections
  - Experiments with low data rates, …
  - Time distribution systems
  - Slaves share link in time division manner
  - Bidirectional on all channels
UCF – Example Topologies

- Hybrid topology:
  - Combination of point-to-point and star-like topologies
  - System-in-a-system applications
  - Bidirectional on all channels
UCF – Low Layer Protocol

- Backbone of UCF
- Handles communication and initialization
- 8b/10b encoding scheme
- 10b K-characters for control and synchronization
- Protocol frames consist always of sequence of several characters:
  - Start of frame
  - Type of the message (either specific destination or broadcast)
  - Protocol identifier
  - Payload
  - Remainder defining the valid bytes in the last transmission
  - End of frame

<table>
<thead>
<tr>
<th>SOF</th>
<th>TYPE</th>
<th>ID</th>
<th>PAYLOAD</th>
<th>REM</th>
<th>EOF</th>
</tr>
</thead>
</table>
UCF – Initialization

• Fixed phase synchronization by sequence of two defined K-characters (x”BCDC”)
• Synchronization character will be send for specific time to let the slaves synchronize
• Attached slaves are scanned by sending an initialization frame containing different DNAs and waiting for response
• DNA is the serial number of an FPGA
• Unique ID and IP assignment for all connection parties
UCF – Priority Handling

- All 64 communication channels have different priorities
- Protocol 0 has the highest and then it cascades down to the protocol 63 which has the lowest priority
- Frames with higher priority are inserted into lower priority frames
- Maintains fixed latency for the timing channel
UCF – User Interface and Configuration

- All channels are addressed via the standardized ARM AMBA AXI4 interface
- Leads to easy interfacing with other IP-Cores
- Configuration of all parameters within one file:
  - Link speed
  - Topology
  - Device type (Spartan6, Virtex6, Artix7)
  - ...

```plaintext
package ConfigUCF_ufc is
  constant strDeviceType : string := "Virtex6";
  constant stdActivateP2P : std_logic := '1';
  constant stdCardPurpose : std_logic := '0';
  constant intTransceivers : integer := 1;
  constant intInterfaces : integer := 1;
  constant intCylcesMin : integer := 8;
  -- Data width settings for User
  constant intDataWidthUsr : integer := 32;
  constant intByteWidthUsr : integer := 4;
  constant intSerials : integer := 1;
  type dna_type is array (integer range <>) of std_logic_vector(63 downto 0);
  constant DNAs : dna_type(intSerials downto 0) := (x"0000000000000000")
end package;
```
UCF – Tests and Measurements

- Point-to-Point topology with 1 slave and 1 master
- Virtex 6 as slave and master
- 2.5 Gbit/s link speed
- Recovered clock jitter ($\sigma$) of 23 ps
- Requires 2 % slice register and 4 % slice LUT utilization on a Virtex 6 LX130T
UCF – Tests and Measurements

• Star-like topology with 12 slaves and 1 master
• Spartan 6 FPGA as slave and Virtex 6 as master
• 1.25 Gbit/s link speed
• Switching time of 16 µs (includes character transmission and synchronization)
• Long term stability test with 99 % link utilization over two weeks
• Forwarding of JTAG possible with up to 100 kHz JTAG frequency
• IPBus over UCF

<table>
<thead>
<tr>
<th>Transmission Time [µs]</th>
<th>Efficiency [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>25000</td>
<td>99,93</td>
</tr>
<tr>
<td>10000</td>
<td>99,84</td>
</tr>
<tr>
<td>1000</td>
<td>98,42</td>
</tr>
<tr>
<td>500</td>
<td>96,90</td>
</tr>
<tr>
<td>100</td>
<td>86,20</td>
</tr>
</tbody>
</table>
UCF – PENeLOPE

- **Precision Experiment on Neutron Lifetime Operating with Proton Extraction**
- 500 Mbit/s
- Slaves read out in Round-Robin manner
- Distribution of random trigger and global clock with determined latency
- IPBus connection
UCF – Belle II Pixel Detector

- Point-to-Point topology with 1 DHC and 5 DHE
- 22 GB/s data rate of the detector
- 6.5 Gbit/s link speed
- IPBus, data and trigger distribution
- Tested complete readout chain from simulated detector data over DHE and DHC to PC
UCF – Conclusion

- IP-Core providing unified communication of up to 64 channels via a single optical link
- Fixed latency for one channel
- 23 ps recovered clock jitter
- Standardized ARM AMBA AXI4 interface for user
- Multiple 1:n and 1:1 connections possible
- Typically 98 to 99 % link utilization efficiency for star-like topologies (16 µs switching time)
- JTAG with 100 kHz frequency
- IPBus
- Tested set-ups for Belle II and PENeLOPE successfully in the lab
UCF – Outlook

- Additional CRC check integration to UCF
- Integration of UCF in the NA64 experiment in 2016
- Beam test with the Belle II set-up at the end of 2016
- Integration of the PENeLOPE set-up into the experiment at the end of 2017
- Integration of UCF in the COMPASS experiment in 2016/2017
- Will be published as an open source project after commissioning during the Belle II beam test

- Poster by Igor Konorov about the *Intelligent FPGA based Event Builder and Data Acquisition System for the COMPASS experiment* on Thursday 9th of June
- Poster by Dmytro Levit about the *Intelligent FPGA Data Acquisition Framework* on Thursday 9th of June
Thank you for your attention