Complete MRF Timing System on MicroTCA.4

MicroTCA.4 Workshop for Industry and Research

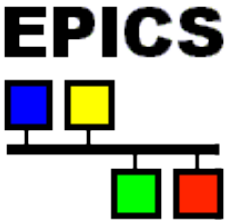
DESY, December 2018

Jukka Pietarinen

Micro-Research Finland Oy

MRF Timing System Development

- The MRF Timing system originally developed for the Swiss Light Source (SLS), Paul-Scherrer Institut in 1999
- First commercially available timing system for particle accelerators
- Current users include
  - SLS, PSI
  - Diamond Light Source Ltd., U.K.
  - SSRF, Shanghai, China
  - Australian Synchrotron
  - ALBA, Spain (Tango)
  - Elettra, Trieste, Italy (Tango)
  - BEPCII, Insitute for High Energy Physics, Beijing, China
  - LCLS, Stanford Linear Accelerator Center, USA
  - KEK, Japan
  - FERMI, Trieste, Italy (Tango)
  - TLS, Taiwan
  - NSLS II, Brookhaven, USA
  - PAL-XFEL, Pohang, South Korea
  - LANSCE upgrade, Los Alamos, USA
  - FRIB, Michigan, USA
  - SwissFEL, PSI
  - STFC, Daresbury
  - ESS, Sweden
- Most of the sites listed above are using EPICS



jukka.pietarinen@mrf.fi

What do you need a timing system for?

- Phase and frequency locked clocks at different locations
- Trigger something at different locations at exactly the same time
- Make something happen in sequence with predefined time intervals
- Synchronize the local time at different locations with high precision
- Timestamp events at different locations to analyze what happened first

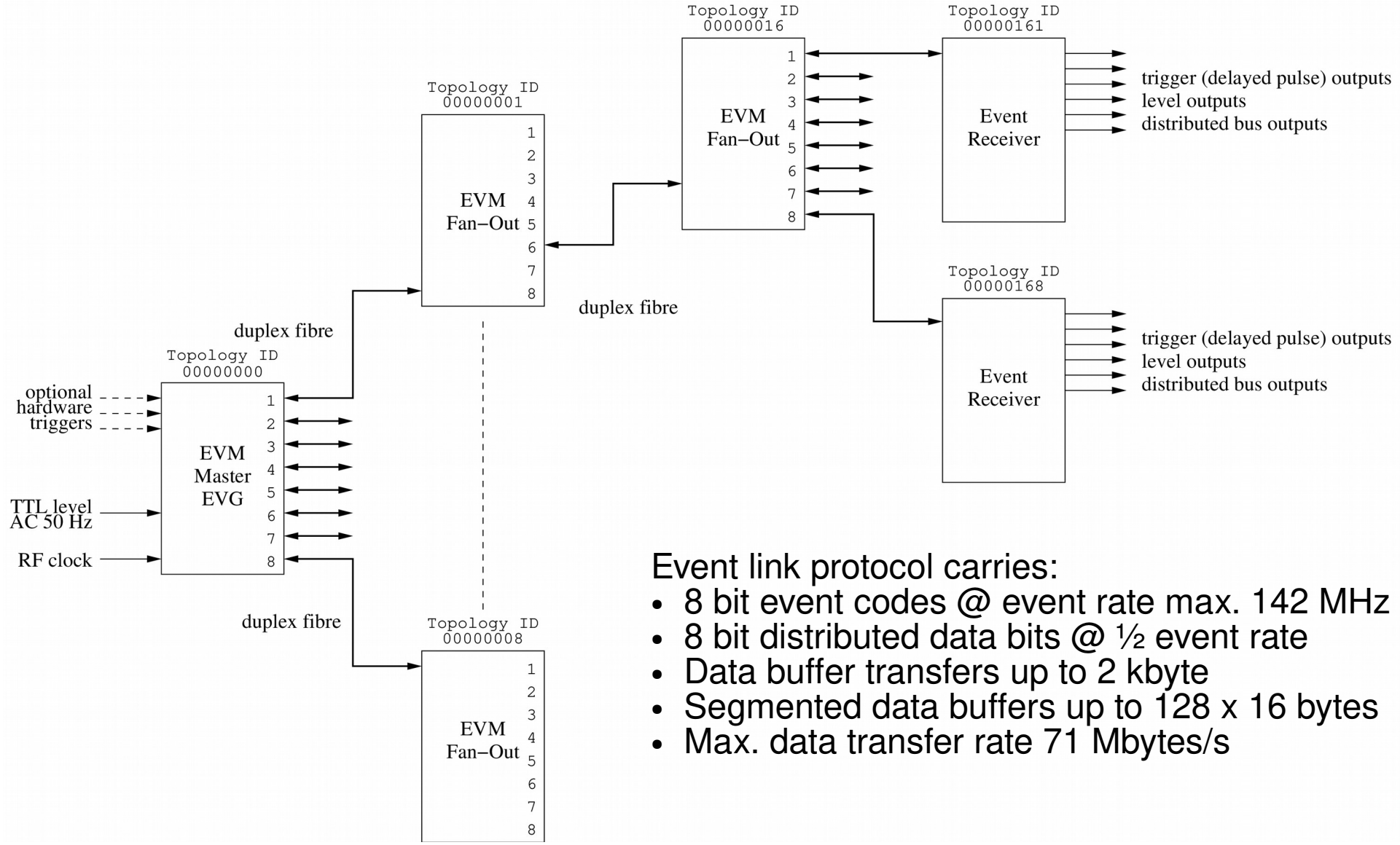
Operating Principle of the MRF Timing System

- Event based system
  - 256 Event codes, one event code/cycle (7 ns - 20 ns or 50 MHz to 142 MHz)
- In addition to events there are eight signals (distributed bus) that are updated at half the rate (14 ns - 40 ns or 25 MHz to 71 MHz)
- Synchronous data transfers
- Transmission medium: optical fiber
  - typically multi-mode, OM3 or better up to 550 m
  - single-mode, up to 10 km
- Line code or “event stream” based on 8B10B encoding

Event Clock

- Base clock of MRF timing system
- Typically driven from external source e.g. accelerator RF signal
- Has to be in range 50 MHz to 142 MHz (7 ns to 20 ns cycle)
- Has to be provided to the Timing System Master, all other nodes regenerate this clock from the event stream received through the fiber
- The Timing Master has an input divider to divide the provided RF clock by 1, 2, 3, ..., 12, 14, 15, 16, ..., 32.

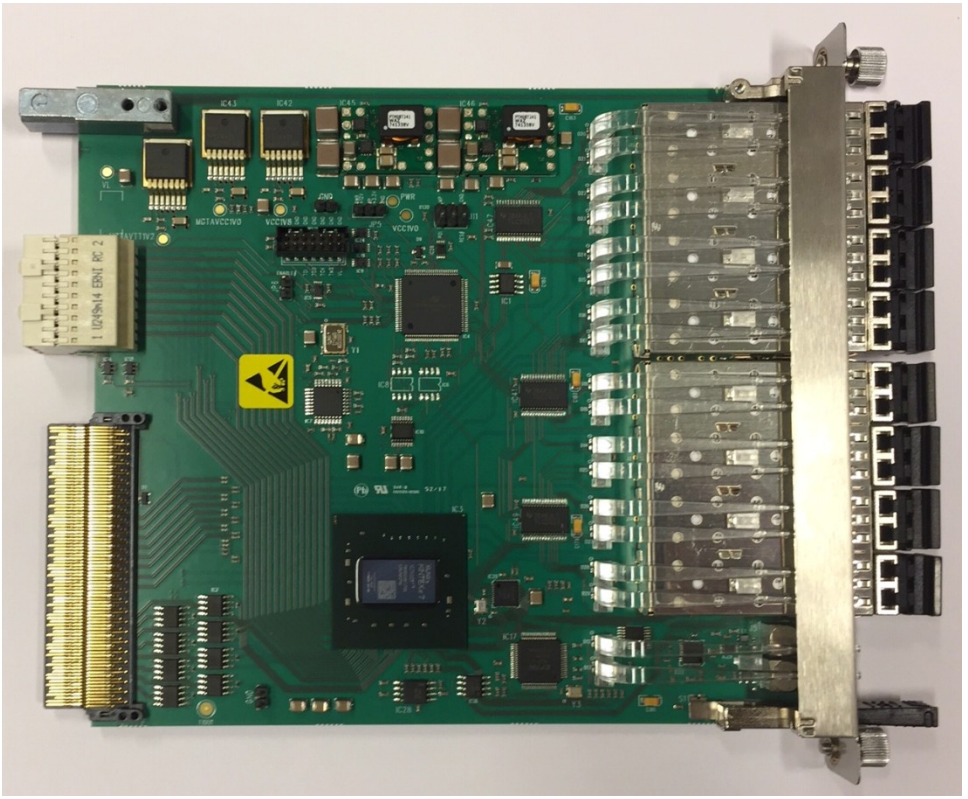
Typical System Layout - Delay Compensation (DC)



Event link protocol carries:

- 8 bit event codes @ event rate max. 142 MHz
- 8 bit distributed data bits @ ½ event rate
- Data buffer transfers up to 2 kbyte
- Segmented data buffers up to 128 x 16 bytes
- Max. data transfer rate 71 Mbytes/s

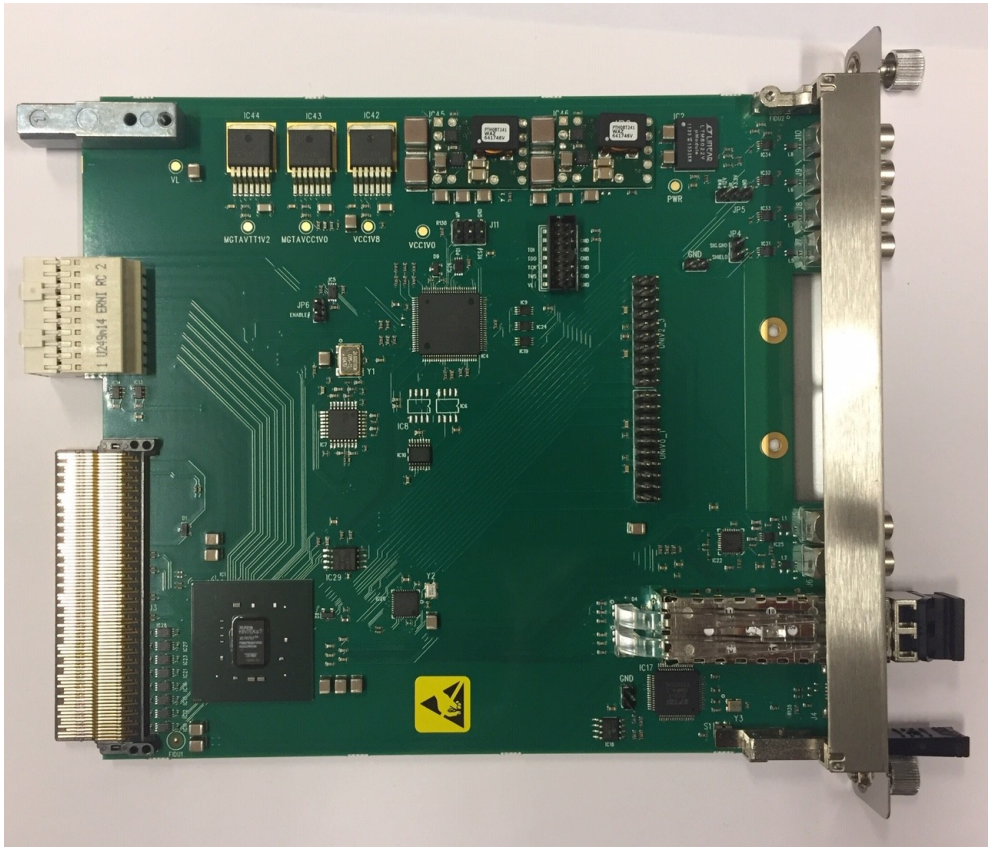
mTCA-EVM-300

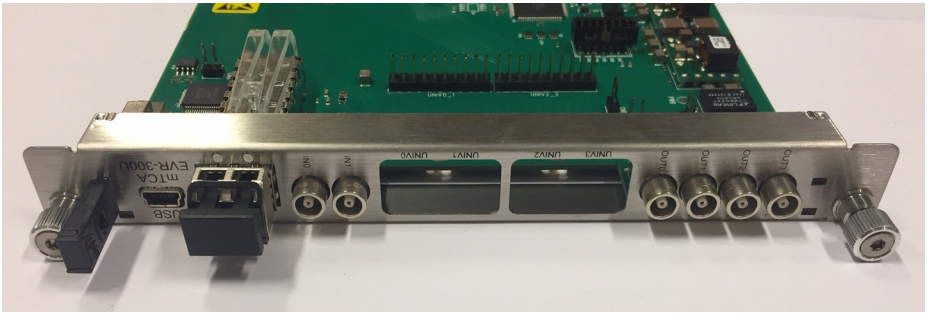


- Event Generator (EVG)
  - 2 sequencers with maskable events
    - Max. 2047 events/sequence
    - 32 bit timestamp
  - 8 multiplexed counters
  - data buffer up to 2k bytes
  - segmented data buffer
    - 127 segments, 16 bytes each
- 7-Way Fan-Out/Concentrator
  - +4 backplane ports
- Two Event Receivers (EVR)
  - 8 internal pulse outputs
  - one sequencer
- Event rate conversion (with some data buffer related limitations)
- Front panel input phase monitoring/select features
- Distributed bus phase selection
- RF input monitoring

mTCA-EVR-300U

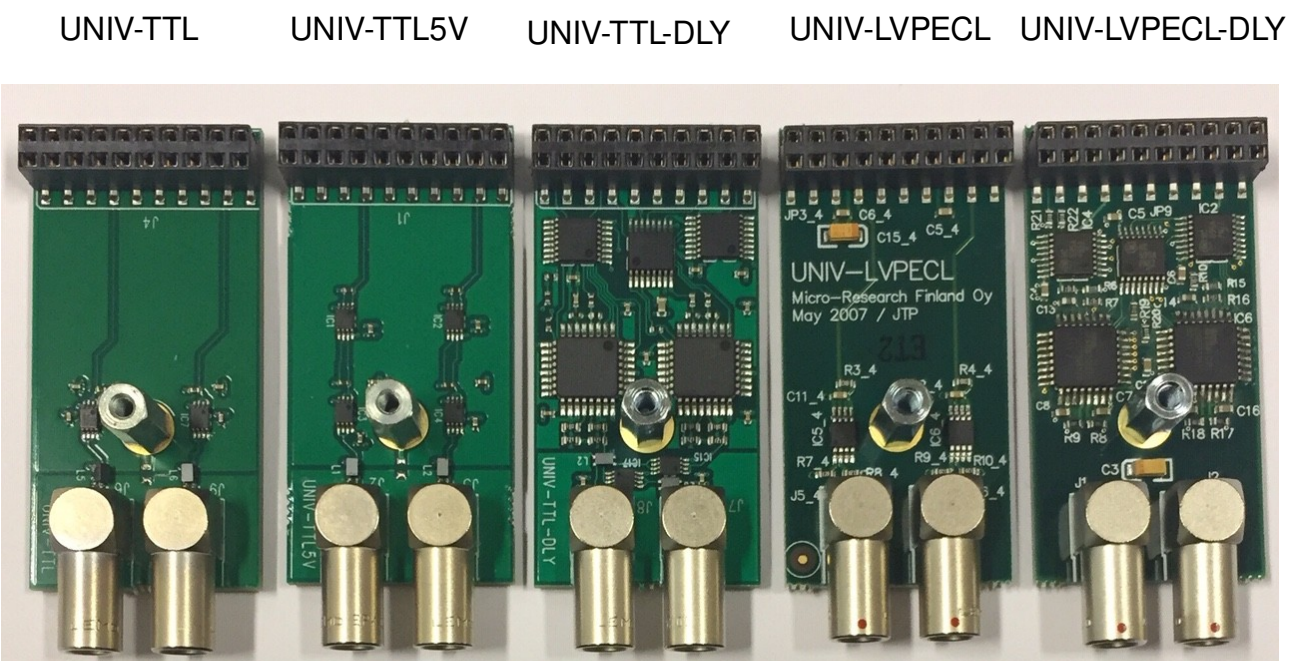
- Event Receiver (EVR)
- Two Universal I/O slots
  - both compatible with -DLY modules
- Four LVTTL outputs
- Two TTL inputs for external triggers
- Backplane triggers
- Can drive TCLKA/TCLKB clocks with GTX logic
- RTM interface, no RTM designed yet





Universal I/O Modules

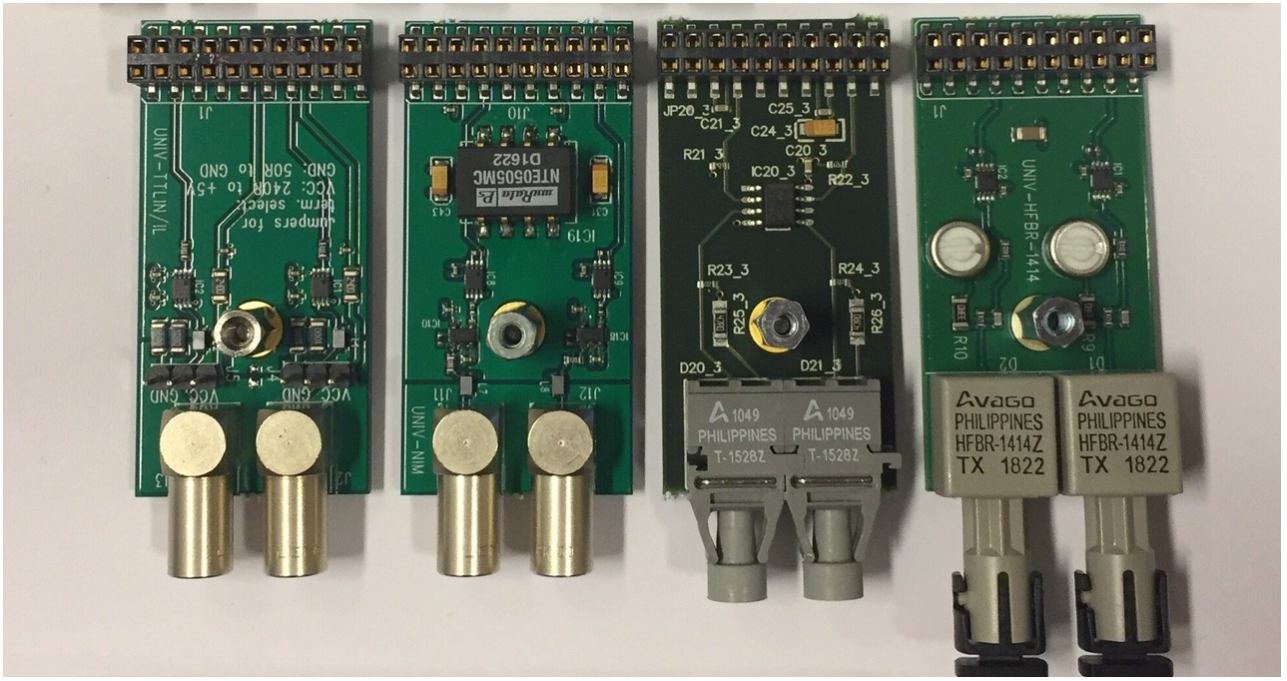
- UNIV-TTL
  - LVTTL output
- UNIV-TTL5V
  - 5V TTL output
- UNIV-TTL-DLY
  - LVTTL output
  - Delay tuning 1024 steps of ~9 ns
- UNIV-LVPECL
  - differential LVPECL output
- UNIV-LVPECL-DLY
  - differential LVPECL output
  - Delay tuning 1024 steps of ~9 ns



Universal I/O Modules

- UNIV-TTLIN
  - TTL input
  - configurable term.
    - 240 ohm to +5V
    - 50 ohm to GND
- UNIV-NIM
  - NIM output
- UNIV-HFBR-1528
  - Optical Output
- UNIV-HFBR-1414
  - Optical Output

UNIV-TTLIN/IL UNIV-NIM UNIV-HFBR-1528 UNIV-HFBR-1414



Open Source Event Receiver - Introduction

- What is the Open Source Event Receiver
  - Basic building block required to build devices receiving the MRF Timing protocol including but not limited to Delay Compensation capability
- What is it not
  - it is not a replacement firmware for current MRF products
  - it is not a complete Event Receiver with MRF product compatible register map
    - No bus/register interface
    - No pulse generators
    - etc.
- Available on GitHub https://github.com/jpietari/mrf-openevr

Open Source Event Receiver - Requirements

- Hardware
  - Xilinx Kintex-7 based FPGA with GTX transceivers **ONLY!**
  - Zynq 7Z030 is Kintex-7 based
  - SFP Transceiver
  - Reference clock for GTX
  - Example design built for
    - Avnet PicoZed 7Z030
    - Avnet PicoZed FMC Carrier Card V2
- Software
  - Xilinx Vivado 2017.4 (Free WebPack version is sufficient)
- Xilinx programming cable
  - e.g. Platform Cable USB II

Avnet PicoZed FMC carrier with 7Z030 SOM

- Avnet PicoZed AES-Z7PZ-7Z030-SOM-G
- Avnet PicoZed FMC carrier AES-PZCC-FMC-V2-G
  - Zynq 7Z030 incorporates
    - Kintex-based FPGA core
    - Four GTX transceivers
    - Dual-core ARM Cortex-A9
- This kit has everything from the hardware point of view to be used as an event receiver