MRF Timing System Overview

Timing Workshop EPICS Meeting Spring 2019

Jukka Pietarinen

Micro-Research Finland Oy

Micro-Research Finland Oy - Company Introduction

- MRF founded in 1985
- Hardware development (and firmware/software)
- History of
    - VME bus products
    - Data acquisition
    - Fibre Optic data transmission
    - Embedded microcontroller systems
        - Motor control/Control system for a lab printing machine
    - Sensor to analyse printing quality for paper
- Current focus on timing systems for particle acceleratos

MRF Experience and Capabilities

- Fibre Optics
  - TAXI chips (125 Mbit/s)
  - Gigabit transceivers
  - Multigigabit transceivers embedded in FPGAs
  - Especially time deterministic applications and low jitter
- FPGAs
  - Xilinx (since the first XC2064)
  - Lattice ECP2M/ECP3
  - VHDL
  - Limited experience Altera
- HW form factors
  - VME / VME64x
  - CompactPCI / PXI
  - PMC
  - CompactRIO
  - PCI Express
  - PXI Express
  - MicroTCA.4

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
- Future users
  - ESS, Sweden
- Most of the sites listed above are using EPICS

EPICS
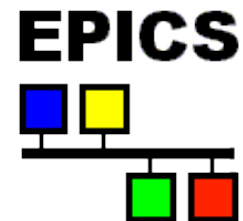
MRF Timing System History

- Functionality based on the APS, Argonne timing system which came from Fermilab
- Redesigned for SLS → Series 100
- Improved performance for Diamond → Series 200
- Timing signals needed for synchronisation of subsystems are applied to Event Generator (EVG) or generated by EVG
- Timing information is converted to 8-bit event codes and disbtributed to Event Receivers (EVR) as an optical signal
- Event clock rate determines timing resolution:
  - Minimum clock rate 50 MHz, 20 ns resolution
  - Maximum clock rate 125 MHz, 8 ns resolution
- 8-bit distributed bus running in parallel and independent of timing events allows distribution of eight signals updated with the event clock rate
- Further improved performance (7 ns resolution) and Delay Compensation for SwissFEL → DC

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

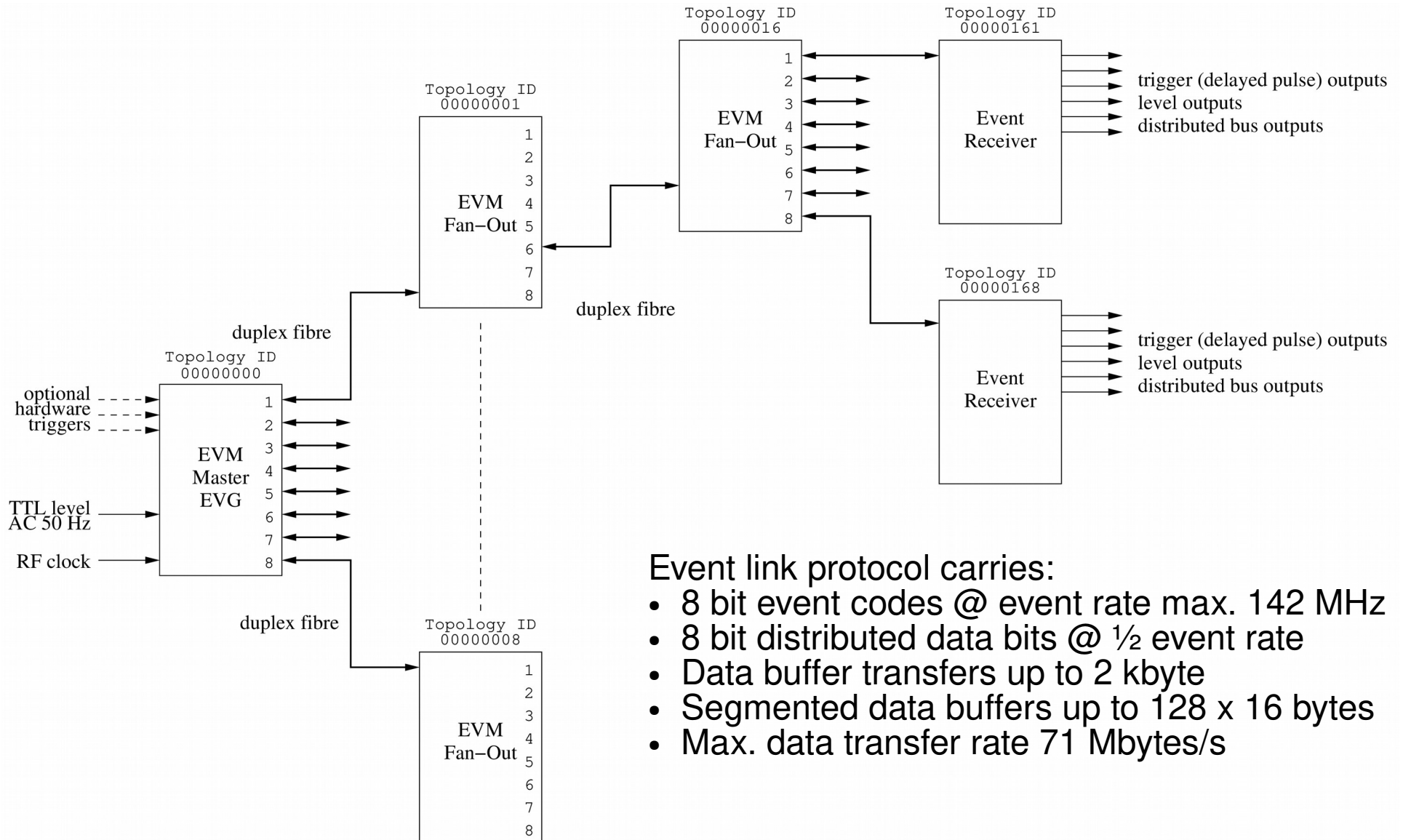 - single-mode, up to 10 km

- 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.

Acronyms etc. timing jargon

- EVG - Event Generator, produces event stream

- EVR - Event Receiver, decodes event stream and performs different hardware & software functions

- Fan-Out - Splits event stream to several ports

- Concentrator - Combines events from several sources into one port

- EVM - Event Master, EVG and Fan-Out/Concentrator

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

EVG Basic Building Blocks

- Trigger Events

- Multiplexed Counters

- Sequencers

- Event Analyzer

- Software Event

- Time distribution logic

- Configurable Size Data buffer

- Segmented Data buffer

Trigger Event

- A Trigger Event has one event code assigned to it

- Upon a trigger this event code is sent out

- Sources for triggers:

  - External signal

  - Multiplexed Counter edge
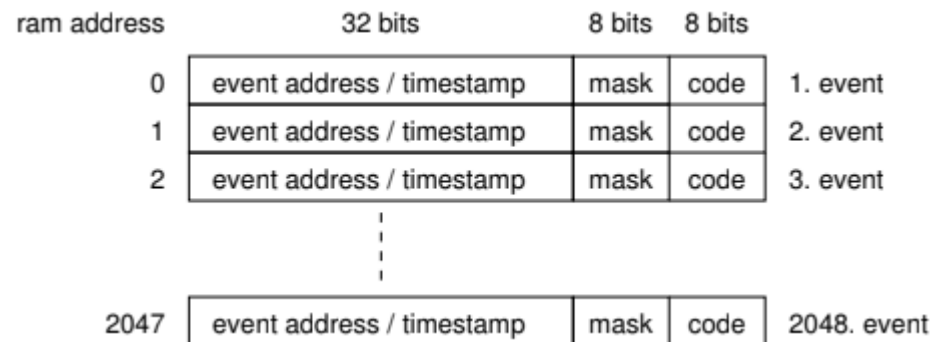
  - AC main synchronization logic

Multiplexed Counter

- 32 bit counters running at the event clock rate

- Can produce frequencies from event clock rate/$(2^{32} - 1)$ to event clock rate/2

- Counters can be reset simultaneously by software

- Configurable polarity
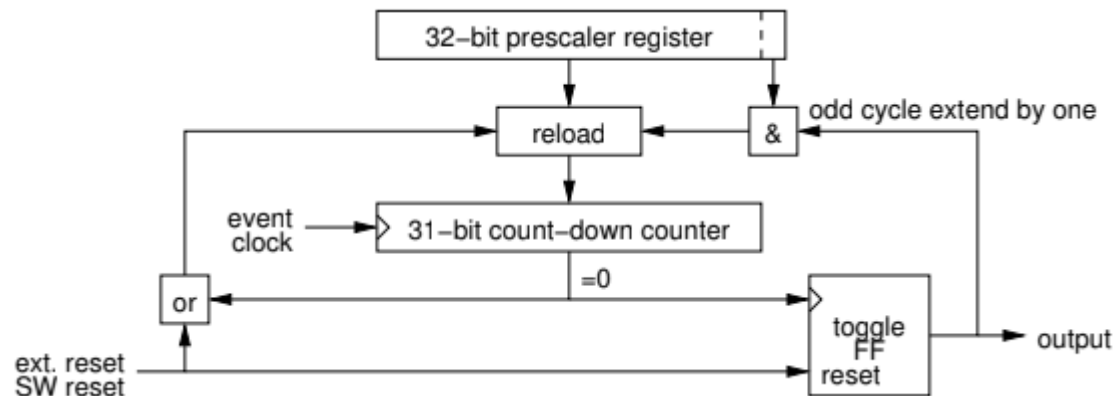
Sequencer

- A sequencer sends out events in predefined order with predefined timing

- 2k dual-port memory for 32-bit timestamp, event code and mask

- Operating modes: single, normal (trigger) and recycle

ram address	32 bits	8 bits	8 bits	
0	event address / timestamp	mask	code	1. event
1	event address / timestamp	mask	code	2. event
2	event address / timestamp	mask	code	3. event
2047	event address / timestamp	mask	code	2048. event

Sequencer triggering

- Multiplexed counter

- AC Synchronization logic

- External trigger

- Software trigger
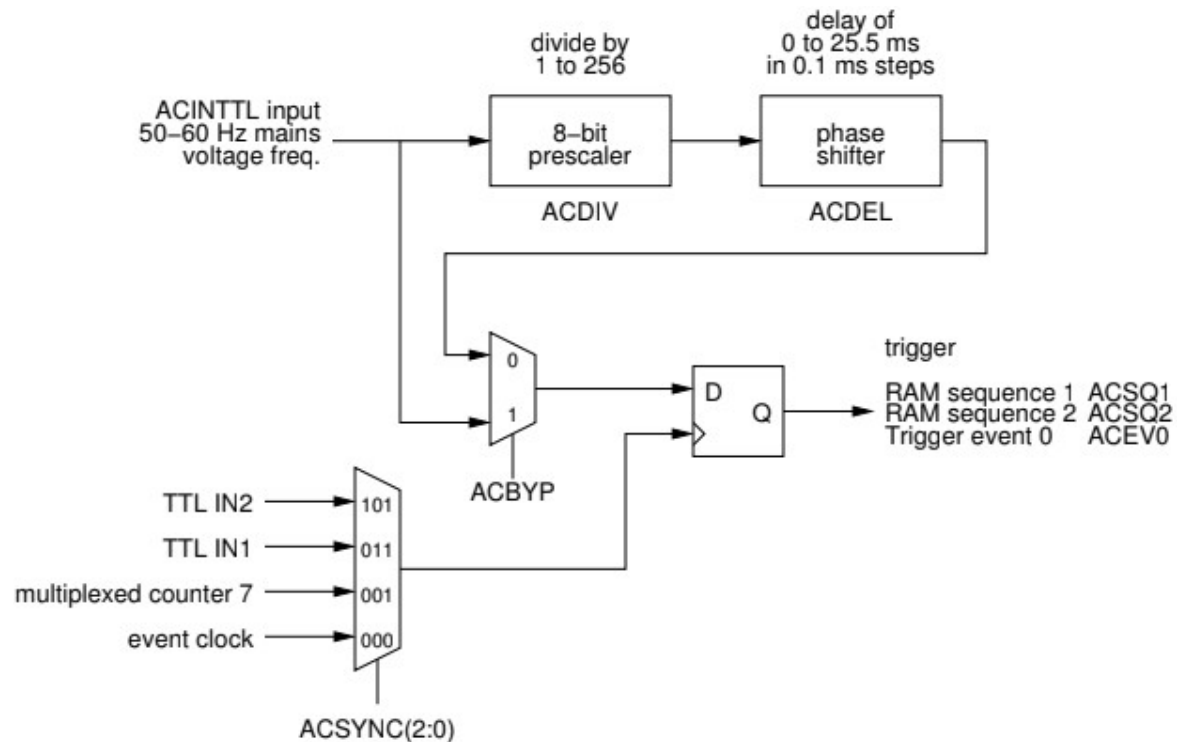
Sequencer operating modes

- Single mode

  – Plays back sequence once until end sequence event

- Normal mode

  – Plays back sequence and waits for next trigger after end sequence event

- Recycle mode

  – Plays back sequence and immediately restarts sequence from beginning after end sequence event

Event Analyzer

- Tools to record all events sent out by the EVG

- 64-bit counter counting event clock cycles

- 2k memory to store event codes, 64-timestamp and distributed bus state

Sequencer and Event Analyser Example

address	timestamp	event	
0	0	0x20	1. Event
1	85168*66	0x2a	2. Event
2	94537*66	0x24	3. Event
3	94538*66	0x25	4. Event
4	94821*66	0x2c	5. Event
5	282002*66	0x30	6. Event
6	283895*66	0x3c	7. Event
7	284000*66	0x7f	End sequence

- 264/4 = 66, 528 us cycles
- Line sync. divider 10
- 50 Hz applied to line sync. Input
- Trigger event enabled to send 0x11 on seq. trigger

Event Analyser with 64-bit time counter

Analyser time (s)	Offset (us)	code
0,190072274	-1,07	11
0,190073347	0,00	20
0,235073191	44999,84	2a
0,240023448	49950,10	24
0,240023977	49950,63	25
0,240173504	50100,16	2c
0,339073511	149000,16	30
0,340073707	150000,36	3c
0,390076629	-1,07	11
0,390077702	0,00	20
0,435077546	44999,84	2a
0,440027803	49950,10	24
0,440028332	49950,63	25
0,440177859	50100,16	2c
0,539077866	149000,16	30
0,540078062	150000,36	3c

Time Distribution Logic

- Support for 32 bit Seconds register

- 32-bit sub second register - e.g. 1 us precision can be used

- Seconds value is send out with event codes

- Sub-second clock is send out as an event or on the distributed bus

- 1 PPS and faster clock e.g. 1 MHz have to be supplied externally and these two clock must be rising edge aligned

Configurable Size Data Buffer

- 2k byte dual-port memory

- Transfer size 4 bytes to 2k in 4 byte increments

- Controlled by software

Segmented Data Buffer

- Introduced with Delay Compensation hardware

- 2k byte dual-port memory

- 128 16-byte segments

- Can send a single or multiple segments at a time

- Controlled by software

- Last segment #127 is reserved for delay compensation

  - delay compensation data

  - Topology ID

EVR Basic Building Blocks

- Event Mapping RAM

- Pulse Generators

- Prescalers

- Timestamping logic

- Event FIFO

- Event Log

- Data buffer

- Segmented Data buffer

Event Mapping RAM

- 256 x 128 bit dual-port RAM

- Each event code has 128 bits assigned and these bits define what functions are to be performed

Map bit	Default event code	Function
127		Save event in FIFO
126		Latch timestamp
125		Led Event
124		Forward event
123	0x79	Stop log
122		Log event
101	0x7A	Heartbeat
100	0x7B	Reset Prescalers
99	0x7D	Timestamp reset event
98	0x7C	Timestamp clock event
97	0x71	Timestamp Seconds '1'
96	0x70	Timestamp Seconds '0'
95 - 64		Trigger pulse generator 31 - 0
63 - 32		Set pulse generator 31 - 0 output high
31 - 0		Reset pulse generator 31 - 0 output low

Pulse Generator
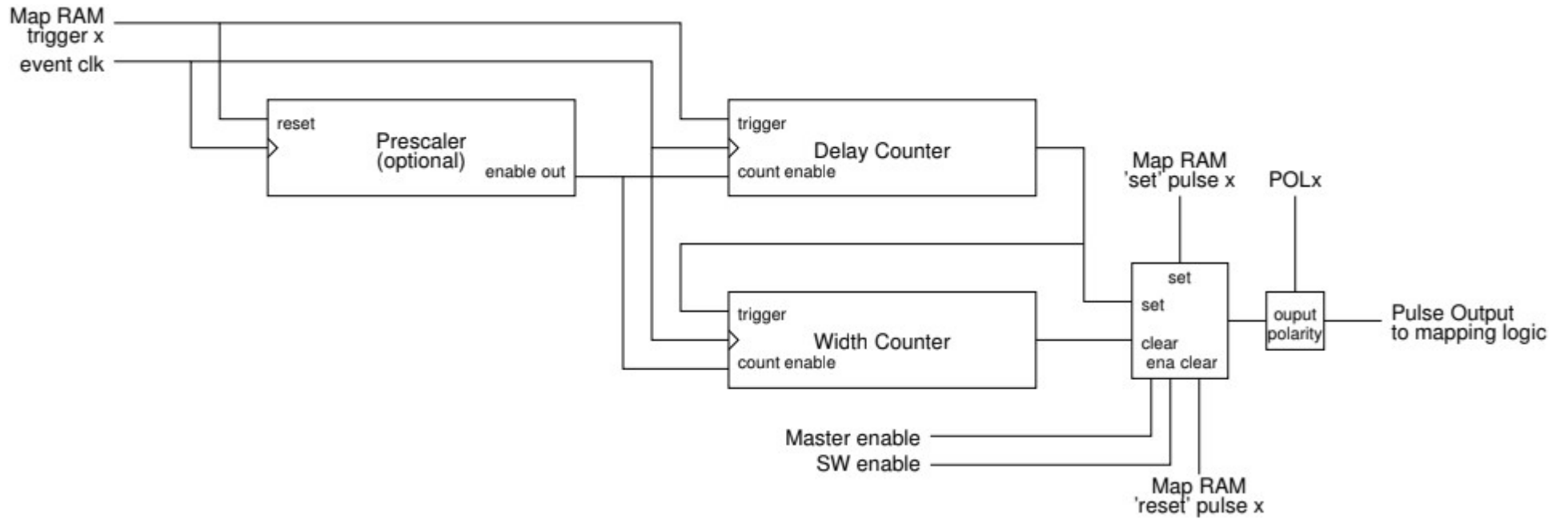
Prescalers

- Programmable counter that can produce different frequencies

- All prescalers are reset by one event code, thus all prescalers on all EVRs can be synchronized

- Can be output directly on I/O or used to trigger pulse generators

- Recent change: addition of phase control

    - When the 'reset prescalers' event is received the prescaler counter value is reloaded from the phase offset register

Timestamping Logic and Event FIFO
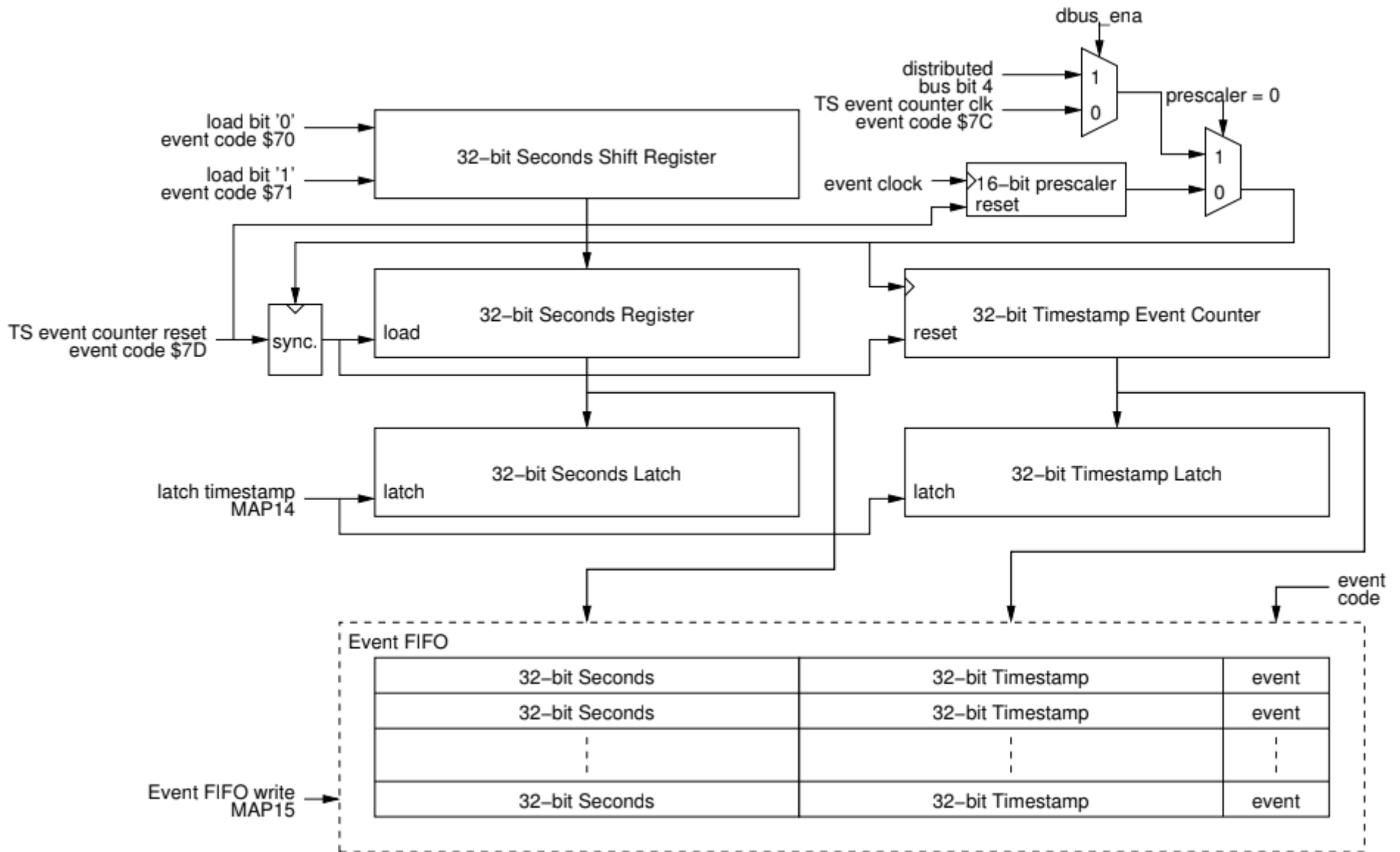
Event Log

- Similar to Event FIFO, but with ring buffer

- Event FIFO stops when full

- Event Log rolls over and overwrites old events

- Can be used for diagnostics: stop log event e.g. after beam loss, read out log events

Data Buffer

- Receive Data Buffer send by EVG

- Data buffer has to be armed to receive anything

- Data buffer is automatically disable after a received buffer

- Data buffers can be lost if buffer is not read and re-armed before EVG sends next buffer

- Software interrupt upon reception

Segmented Data Buffer

- Receive Segments send by EVG

- Segmented Data buffer is enabled all the time

- Flags for each segment: receive complete, checksum mismatch, overflow flag

- Each segment has a received data bytes counter

- Software interrupt configurable for each segment independently

GTX Outputs

- 40 bits/event clock cycle resolution (200 ps @ 125 MHz event clock)

- Operating modes:
  - pulse
  - pattern
  - frequency
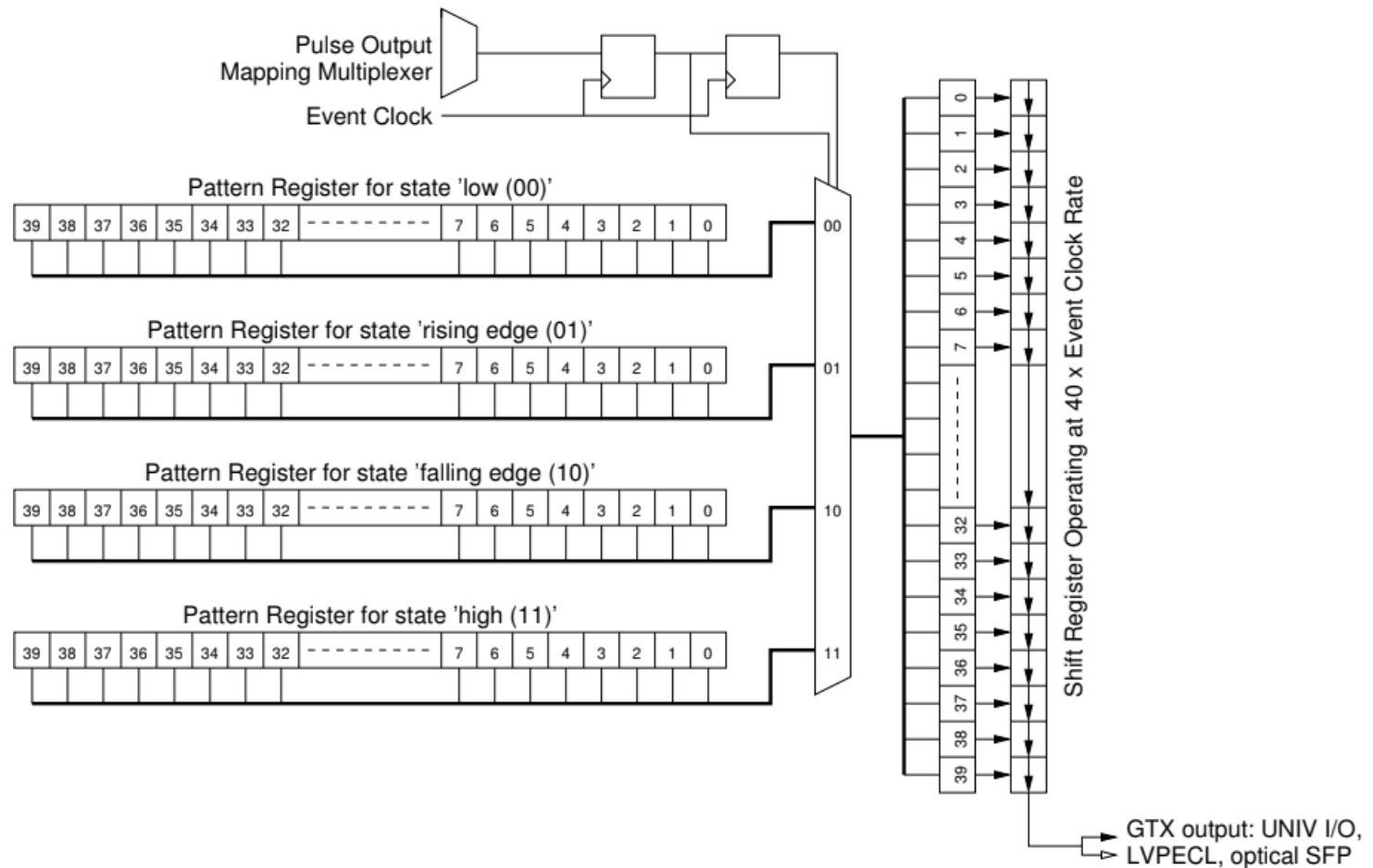
Flip-Flop Outputs (recent addition)

- Output stage that is using two pulse generators to control a set/reset flip-flop

- Flip-flop 0

  - high level on pulse generator 0 sets output high

  - high level on pulse generator 1 resets output low

- Flip-flop 1

  - high level on pulse generator 2 sets output high

  - high level on pulse generator 3 resets output low

- Etc.

- If both PGs are high → output low

EVM, EVG, EVR

To make it even more confusing...

- The difference between Event Generator and Event Receiver is not that clear anymore as it used to be

- EVG features have been added to EVRs and vice versa

Event Master

- Introduced with the latest series i.e. Delay Compensation (DC) capable hardware

- Can operate as the System Master (Top level) or Fan-Out (lower level)

- Consists of

  - one EVG

  - Fan-Out / Concentrator

  - Two "internal" EVRs

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

EVM configured as System Master
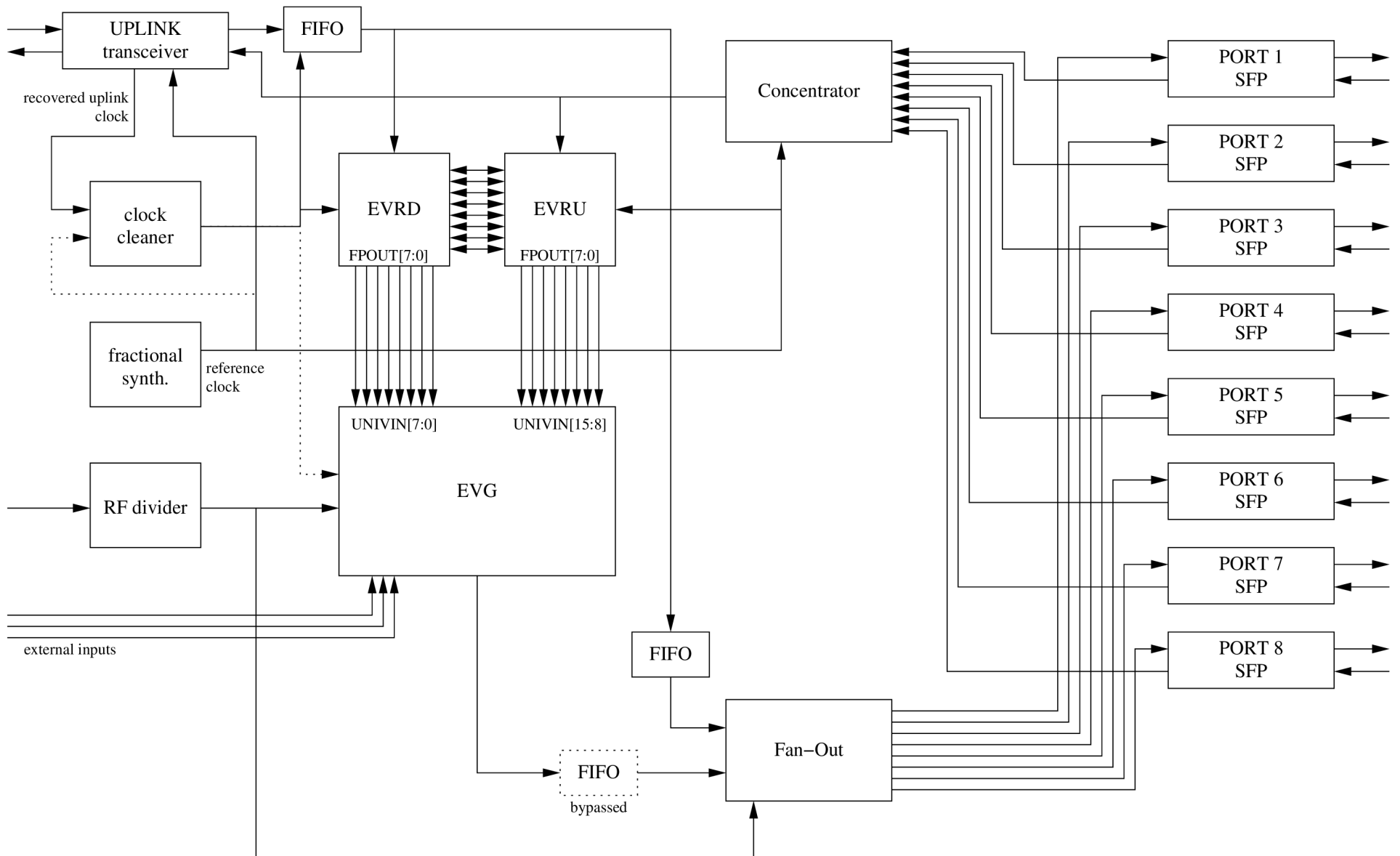
EVM configured as Fan-Out
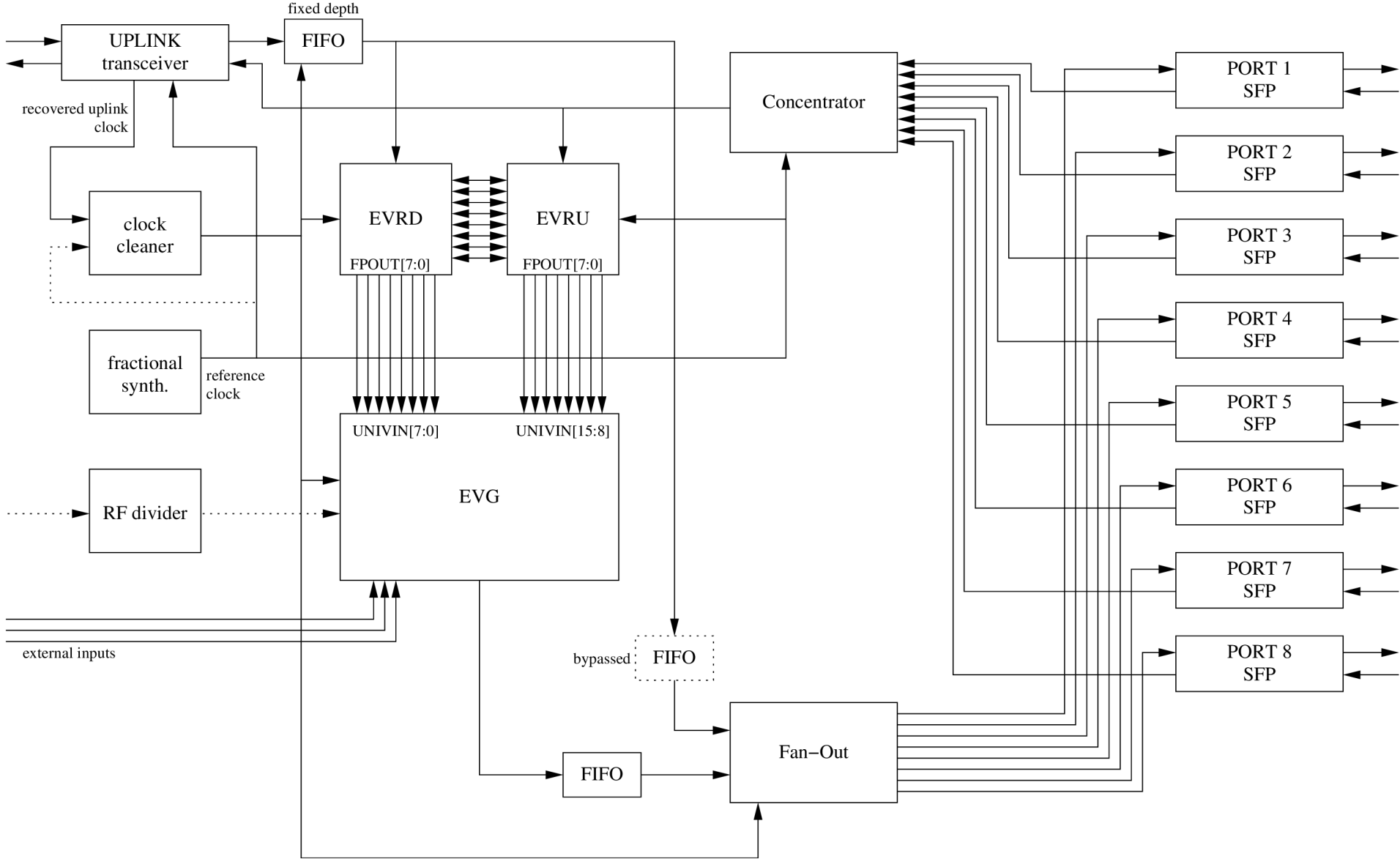
EVR Simplified Block Diagram
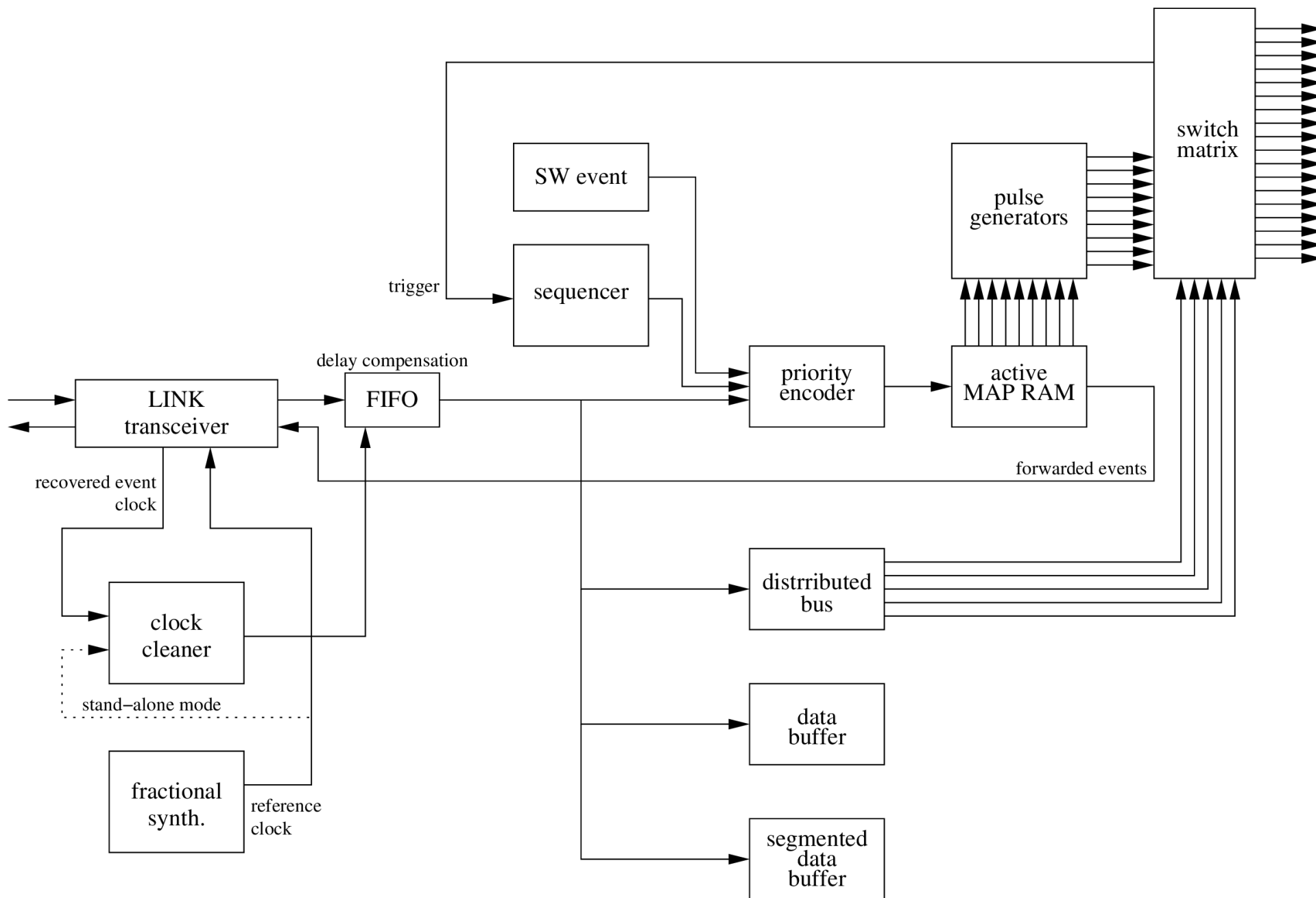
Topology of Simple System mainly for testing purposes without EVG and Delay Compensation
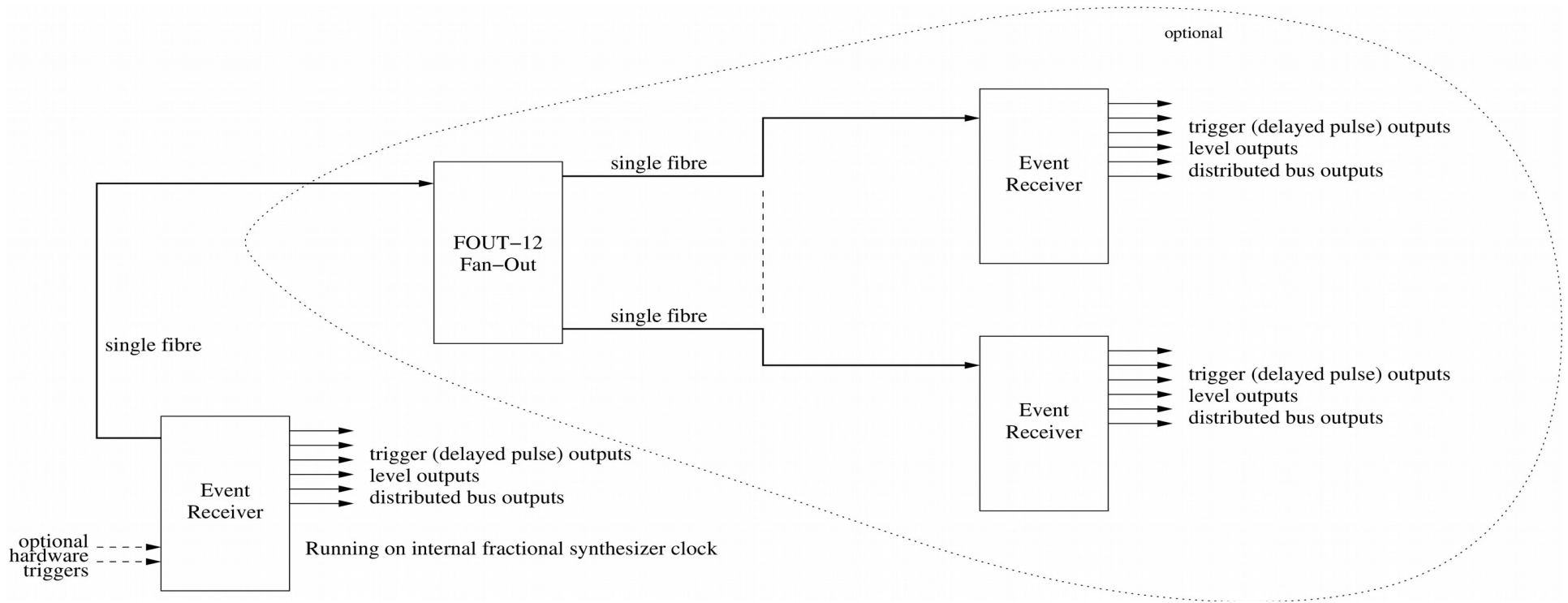
MRF Timing System Delay Compensation Hardware

Timing Workshop EPICS Meeting Fall 2018

Jukka Pietarinen

Micro-Research Finland Oy

VME-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
- 8-Way Fan-Out/Concentrator
- 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
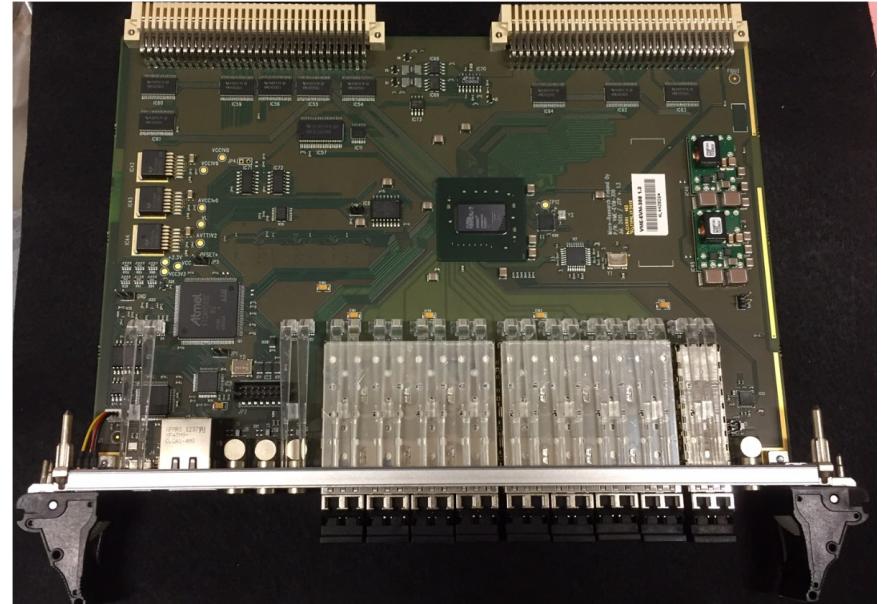
VME-EVR-300

- Event Receiver (EVR)
- Four Universal I/O slots
  - all compatible with -DLY modules
  - one slot driven by GTX logic
- two differential CML outputs driven by GTX logic
- two TTL inputs for external triggers
- transition board interface for additional eight Universal I/O modules (no support for -DLY modules)

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
- 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.4 Trigger Bus Backplane I/O
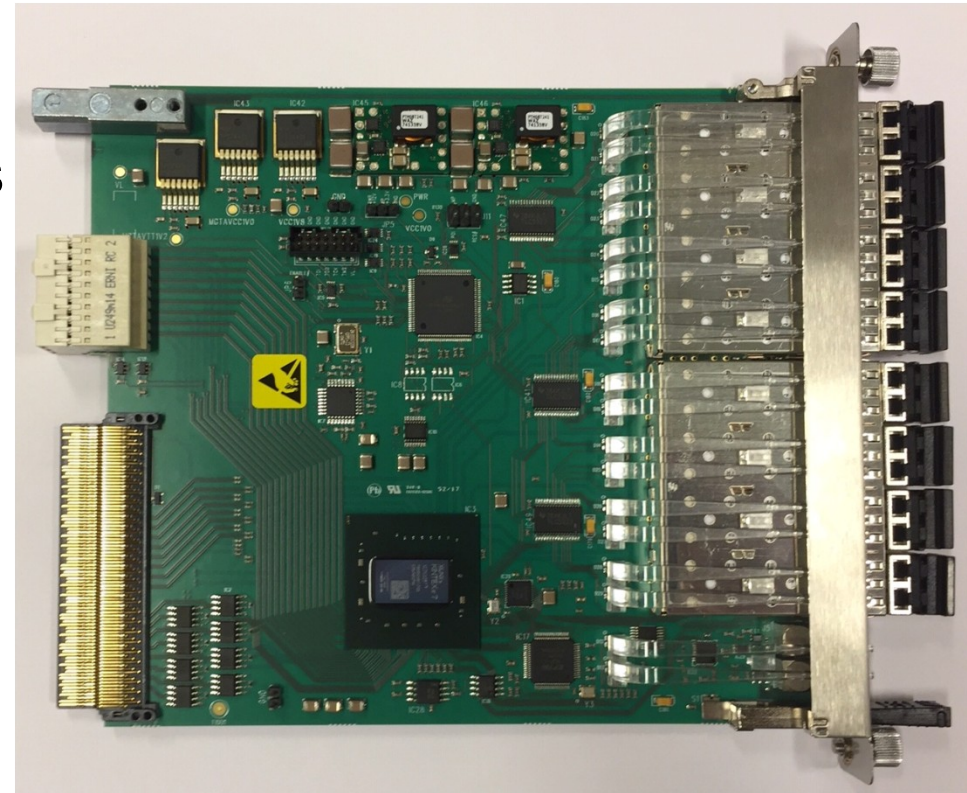
mTCA-EVR-300U

- 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
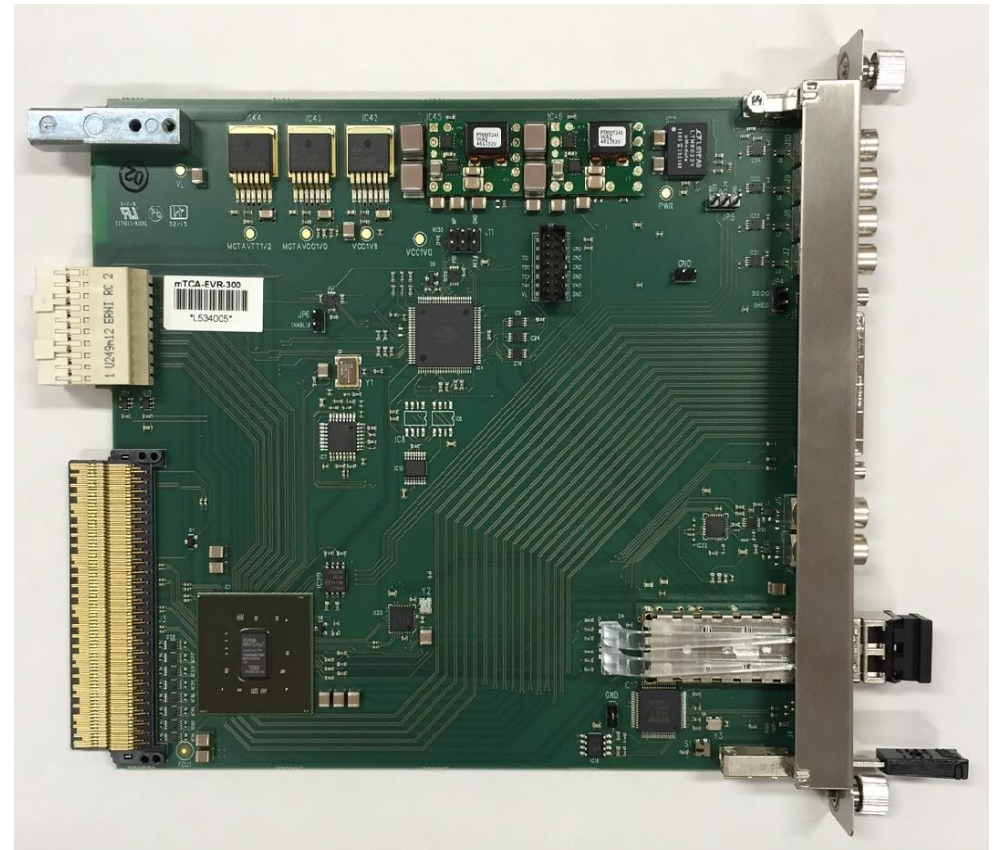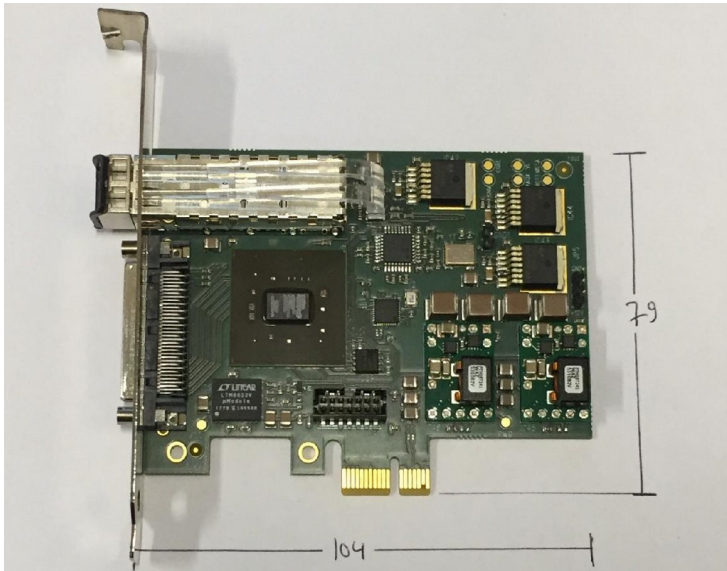- External triggers from backplane (Trigger Bus), Universal I/O and RTM
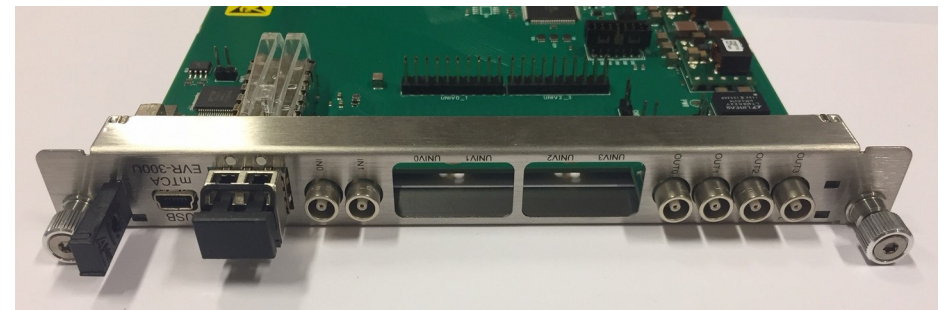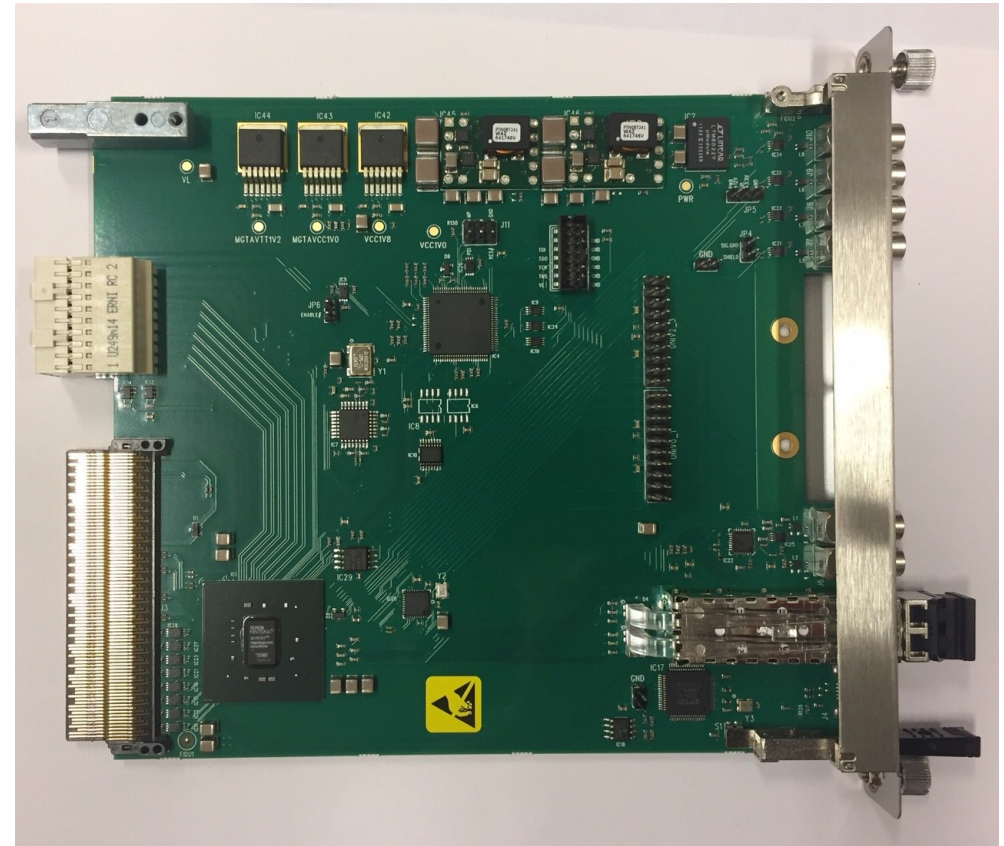
mTCA-EVRTM-300 in development

- Rear Transition Module (RTM) for mTCA-EVR and mTCA-EVM
- Five Universal I/O Modules
- Support for -DLY modules in all slots
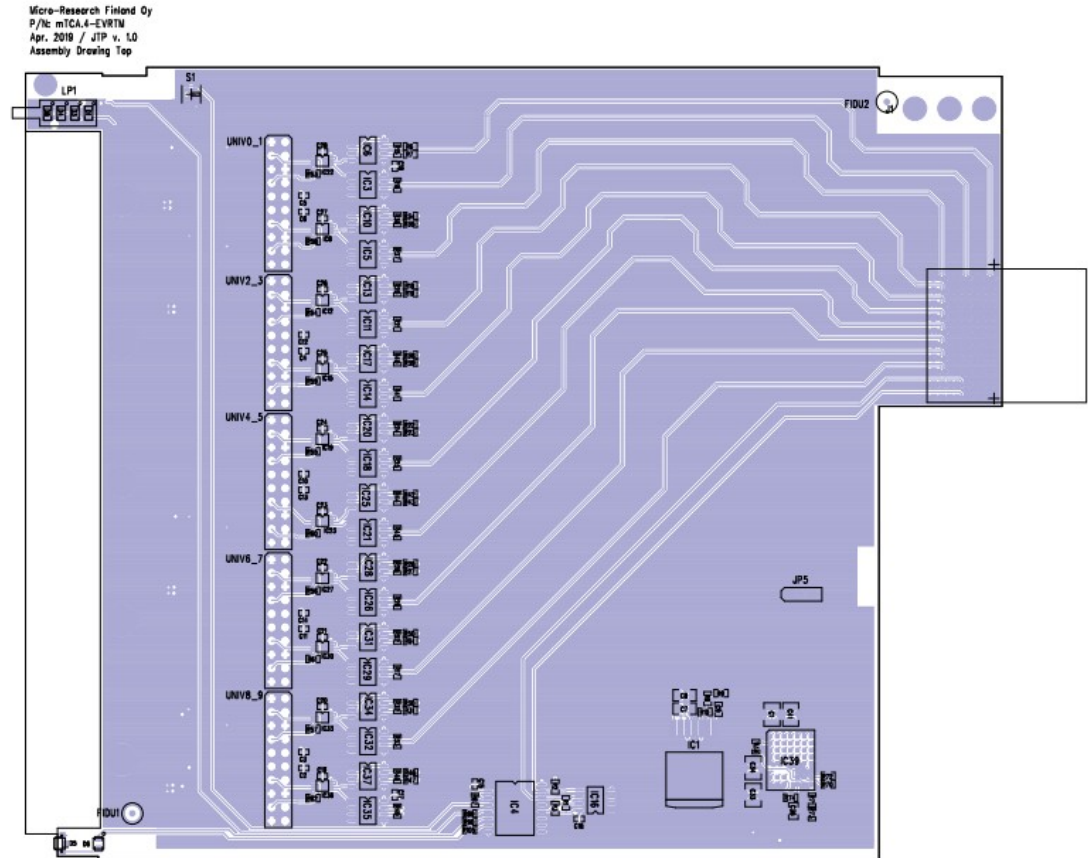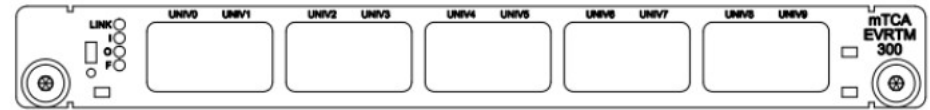
mTCA-EVR-300I, PCIe-EVR-300DC

- I/O through microSCSI connector (becoming obsolete)

I/O Breakout Issues

- IFB-300 mechanics becoming obsolete
- Molex cable discontinued
- VHDCI/microSCSI connector is fragile and you have to fix the cable mechanically, cannot rely on screwing in connector
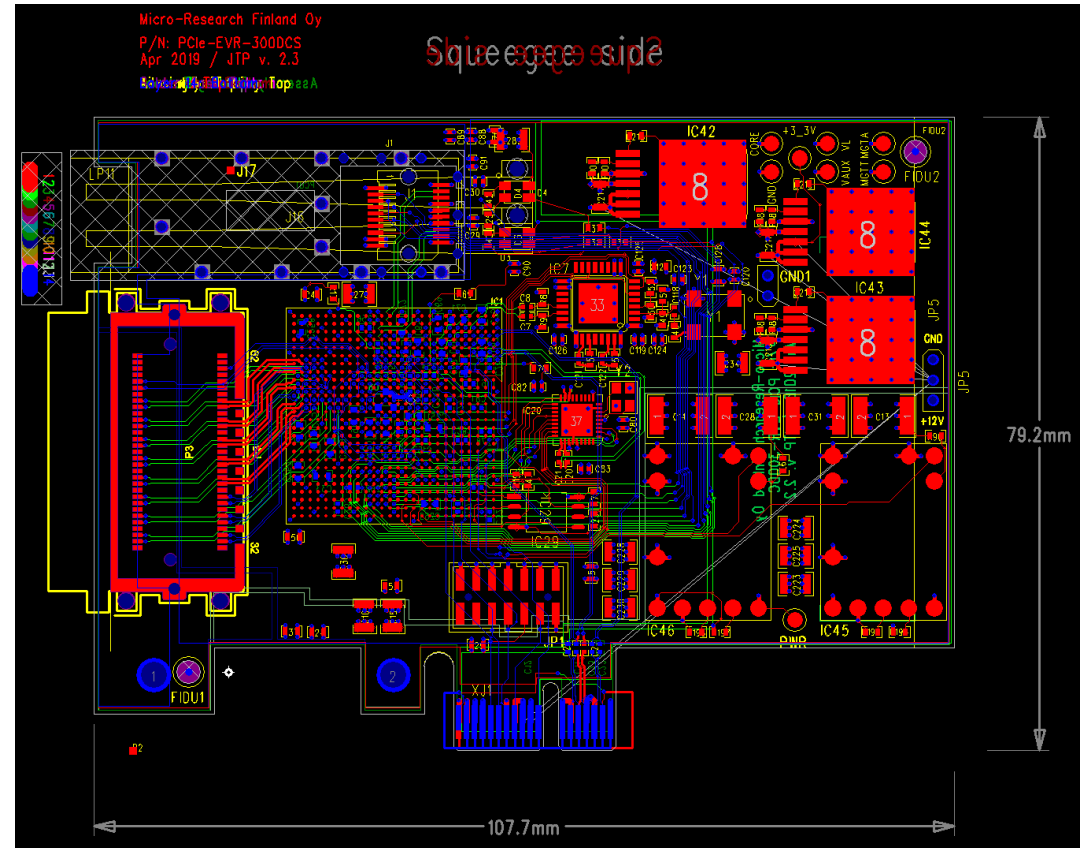- Alternative ways to get I/O from small front panels?
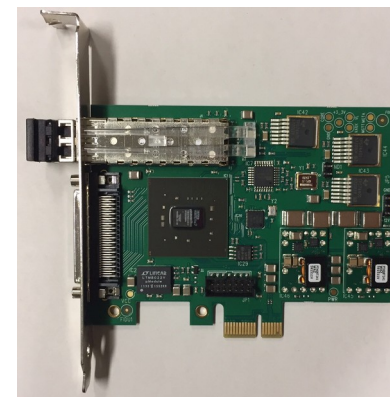
- Samtec ERC-031-01-0
- Samtec ERI8-031-S-D-RA
- Samtec EPLSP-031-1000

Solution to I/O Breakout Issue

- Redesign of PCIe-EVR-300DC → PCIe-EVR-300DCS
  - Connectors needs more space
  - 5 mm longer than PCIe-EVR-300
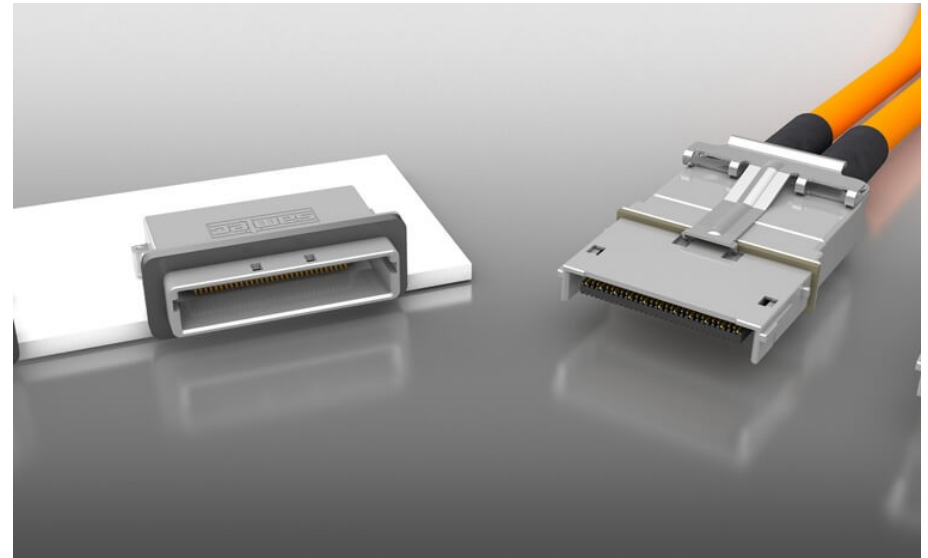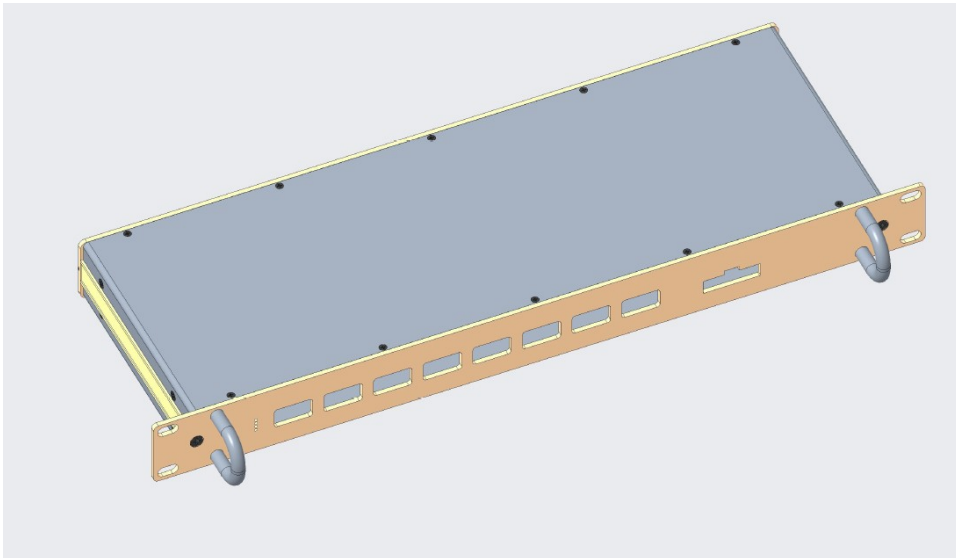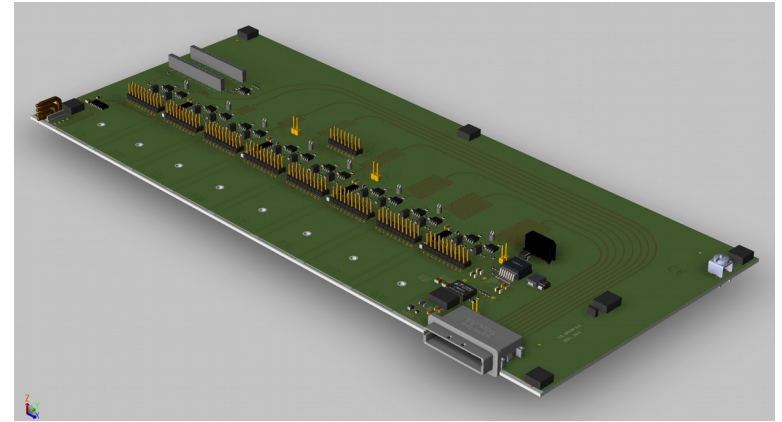
- New IFB-300SAM design

- Samtec ERC-031-01-0
- Samtec ERI8-031-S-D-RA
- Samtec EPLSP-031-1000

New IFB-300SAM breakout box

- MRF and Cosylab cooperation
- Features new Samtec ERI8 connector (more robust) and matching cable
- 1U enclosure rackmountable (19" rack closet)
- Available in Q3 2019

New PXIe-EVR-300DC

- MRF and Cosylab cooperation (co-development and co-production)
- New Kintex-7 FPGA (XC7K160T)
- EVR available also with connector to IFB-300SAM
- Available in Q3 2019
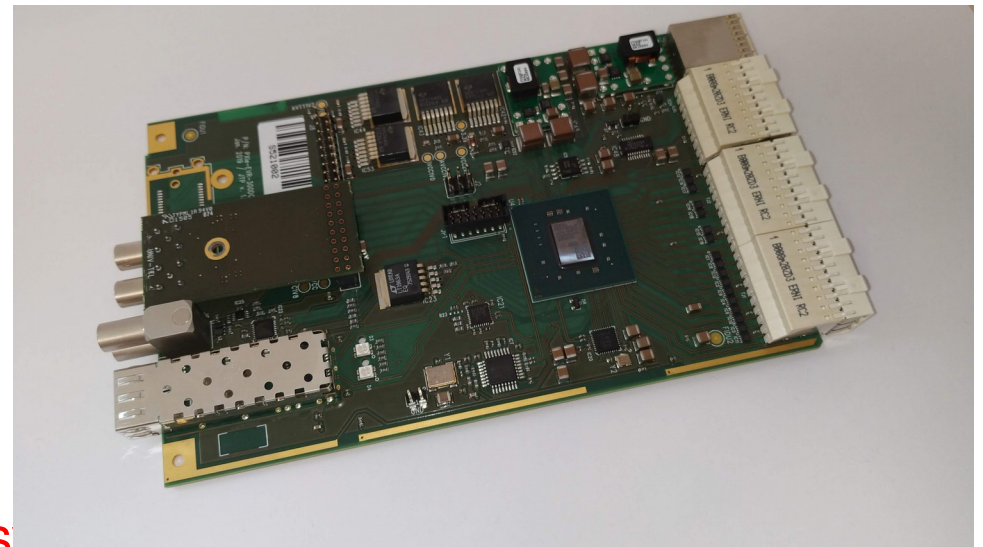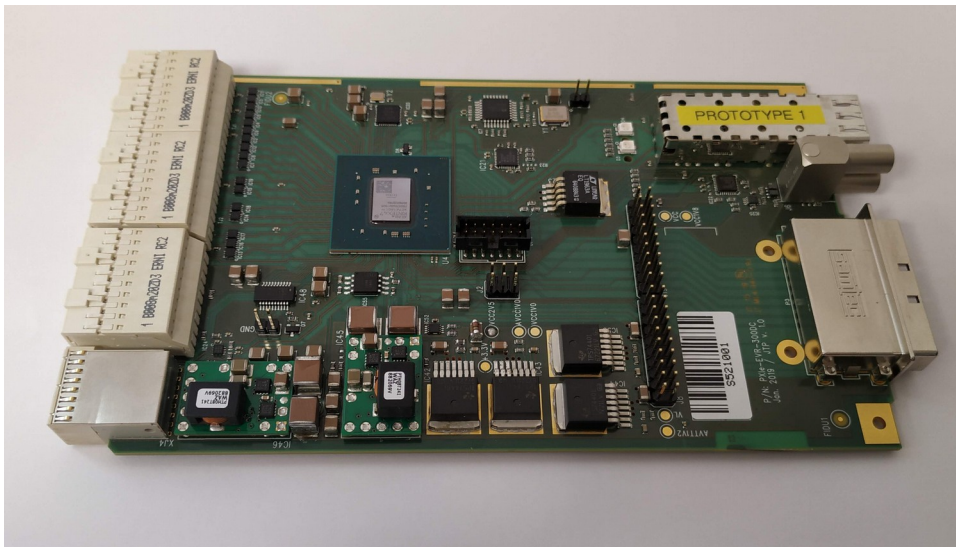
For additional information contact Cosylab (info@cosylab.com) or MRF

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
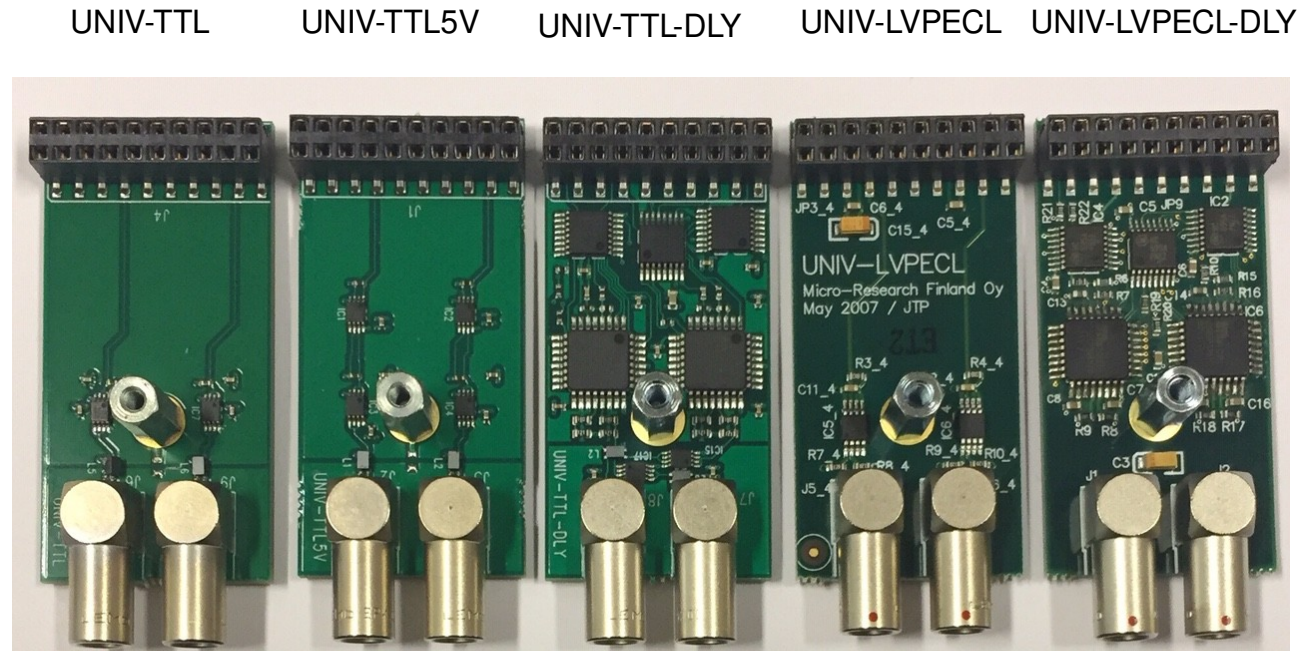
Universal I/O Modules

- **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
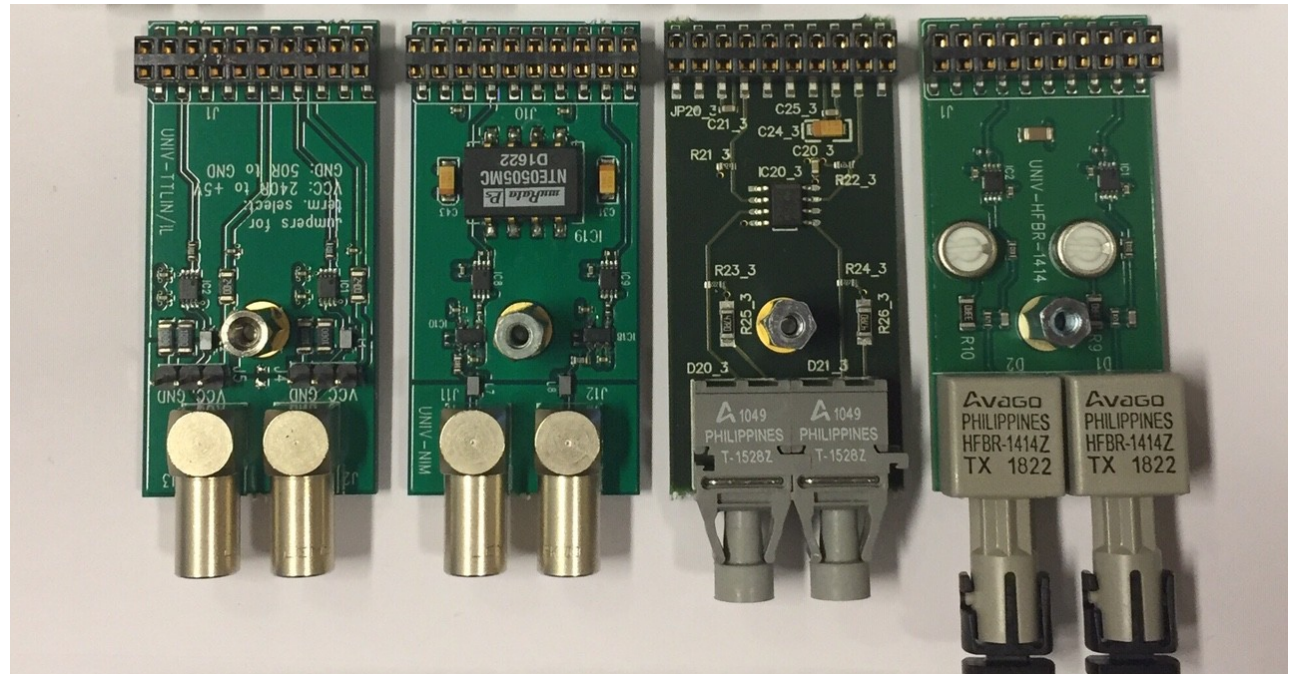
MRF Timing System Delay Compensation Details
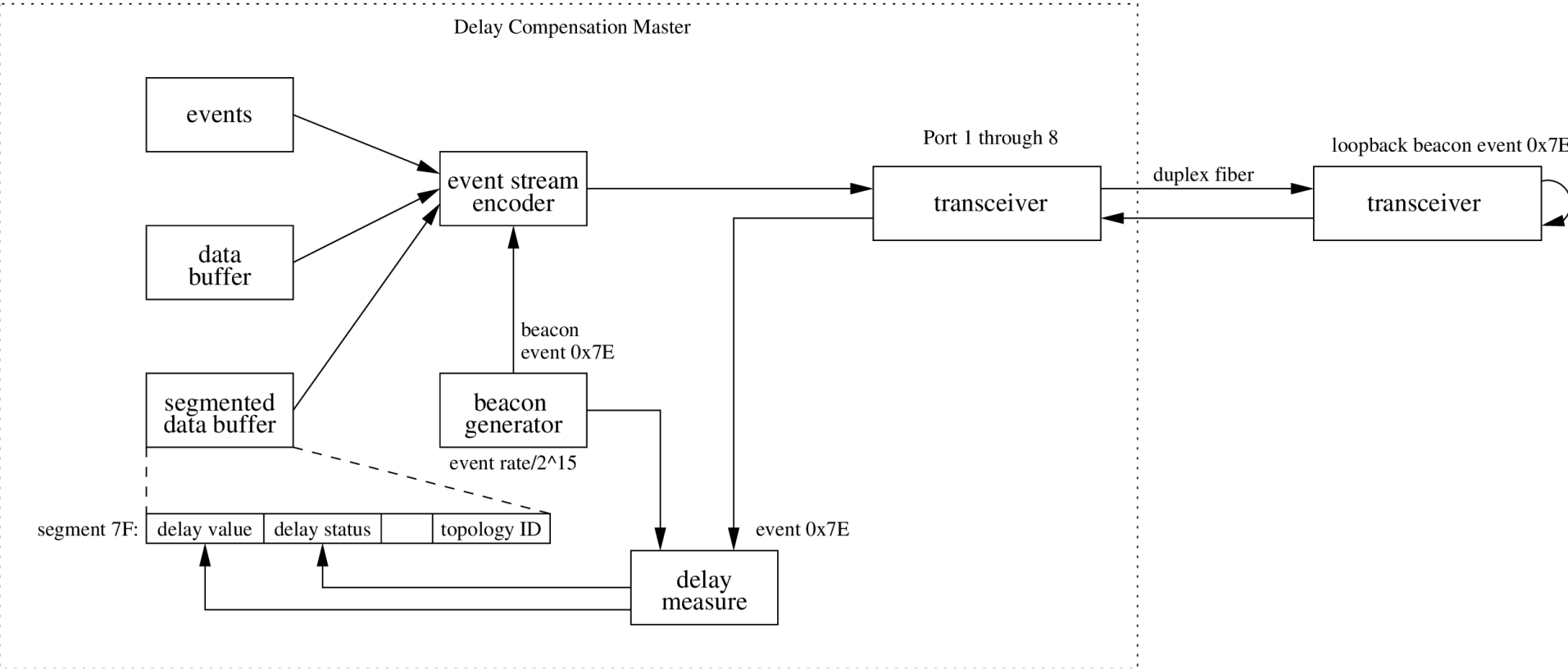
Timing Workshop EPICS Meeting Spring 2019

Jukka Pietarinen

Micro-Research Finland Oy

Delay Compensation Implementation

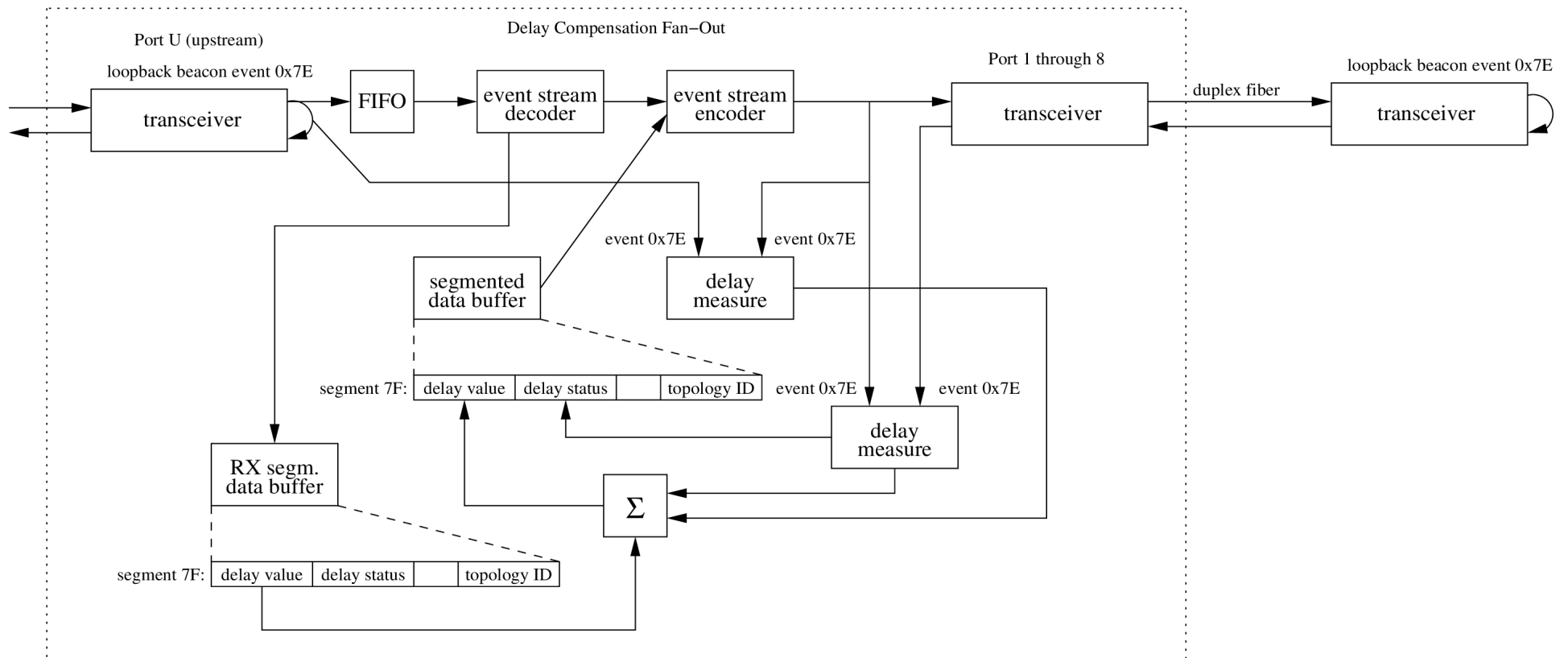- The Timing System Master sends out a periodic beacon event

- Each node immediately sends the beacon event back upon reception

- The loop delay between the sent and "reflected" beacon is measured and this delay value is added up and forwarded to the downstream node

- Compensation is done at the last stage by adjusting the depth of the delay compensation FIFO to match a programmable target delay

VME-EVM-300 Configured as Delay Compensation Master

VME-EVM-300 Configured as Delay Compensation Fan-Out

Delay Compensation Event Receiver

Delay Compensation Event Receiver

loopback beacon event 0x7E

transceiver

FIFO

event stream decoder

recovered event clock

event 0x7E

event 0x7E

clock cleaner

delay measure

RX segm. data buffer

phase control

segment 7F: | delay value | delay status | | topology ID |

Σ

−

DC target value
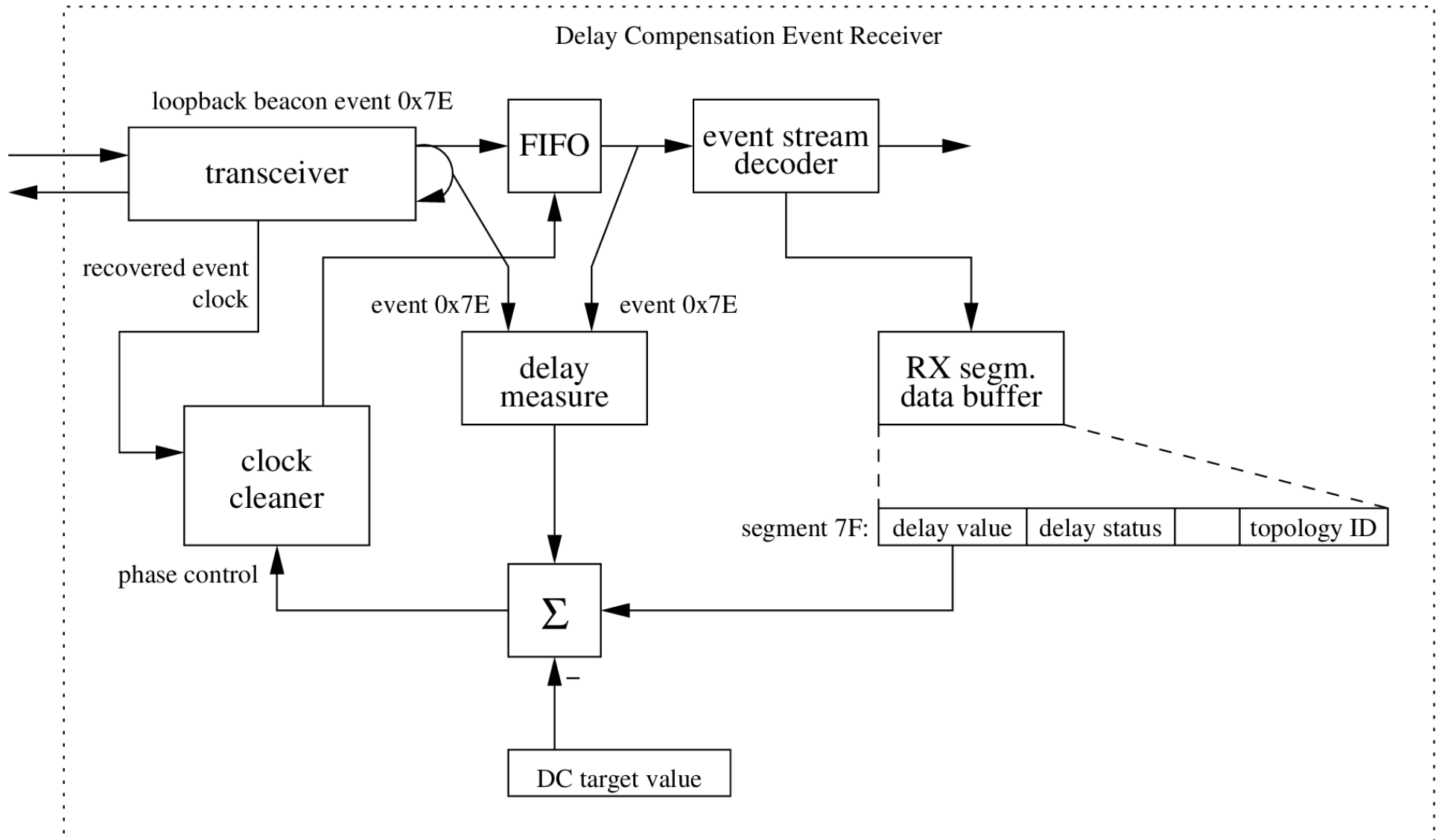
Delay Compensation Capable Event Receiver in non-DC mode
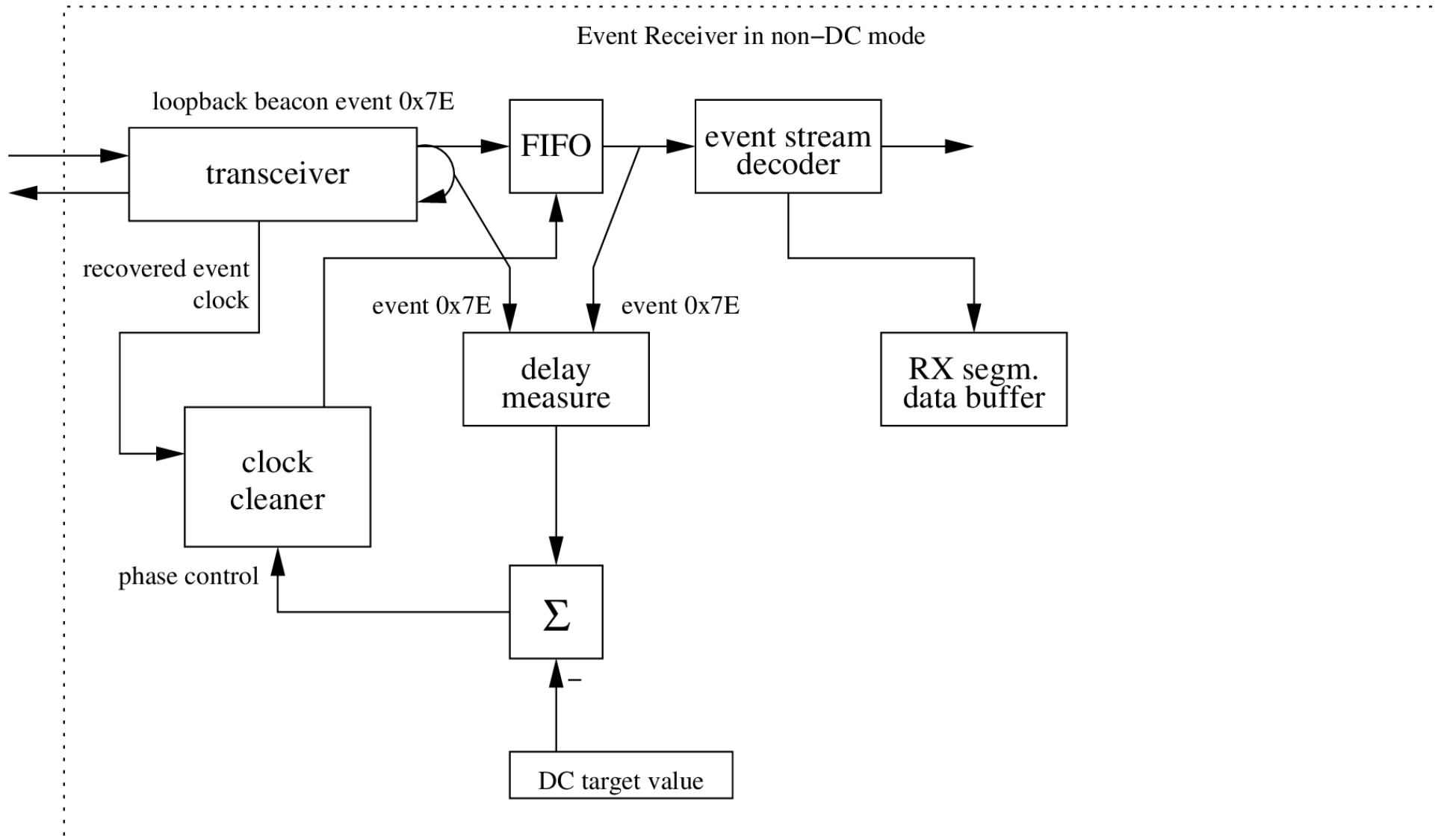
MRF Timing System Delay Compensation Protocol Details

Timing Workshop EPICS Meeting Spring 2019

Jukka Pietarinen

Micro-Research Finland Oy

Compatibility with the Earlier Protocol

- Delay Compensation extends the protocol by:

    - Using event code 0x7E as beacon event

    - Adds the segmented data buffer that uses a different transfer start "comma character" than the data buffer

- Both protocol changes are ignored by pre-DC event hardware

- Timing is similarly deterministic as with a pre-DC distribution network

- Delay Compensation EVRs can be used with the earlier protocol in non-DC mode

    - The Delay Compensation FIFO can be used to fine tune the EVR output delays as a group with very high resolution

- EVMs require the top node to be configured as the Delay Compensation Master

- EVMs can be used in Delay Compensation Mode only

MRF Event System Protocol

- Based on 8B10B encoding using data characters D00.0 through D31.7 and K-characters K28.0, K28.1, K28.2 and K28.5
- Each event clock cycle two characters are transmitted
  - One "event code" character
  - One "data" character
- Event code slot
  - D00.0 – null event
  - D01.0 through D31.7 – event codes 0x01 through 0xFF
  - K28.5 comma character for synchronization, both 8B10B encoding and event protocol, sent out every 4th event slot instead of character D00.0
- Data slot
  - Multiplexed between 8 bit distributed data bus and data transfers
  - Distributed data bus transfers parallel to K28.5 event slots

MRF Event System Protocol

- Data transfers
  - Configurable size data buffer
    - K28.0 start of data packet
    - Dxx.x, Dxx.x, Dxx.x, … packet data
    - K28.1 end of data packet
    - Dxx.x, Dxx.x checksum 0xFFFF – sum of data bytes
  - Segmented data buffer (16 byte segments)
    - K28.2 start of segmented data packet
    - Dxx.x start segment number 0x00 to 0x7F (last segment reserved for delay compensation)
    - Dxx.x, Dxx.x, Dxx.x, … segment data
    - K28.1 end of transfer
    - Dxx.x, Dxx.x checksum 0xFFFF – segment address – sum of data bytes