# Software-assisted Event Builder

Dima Levit

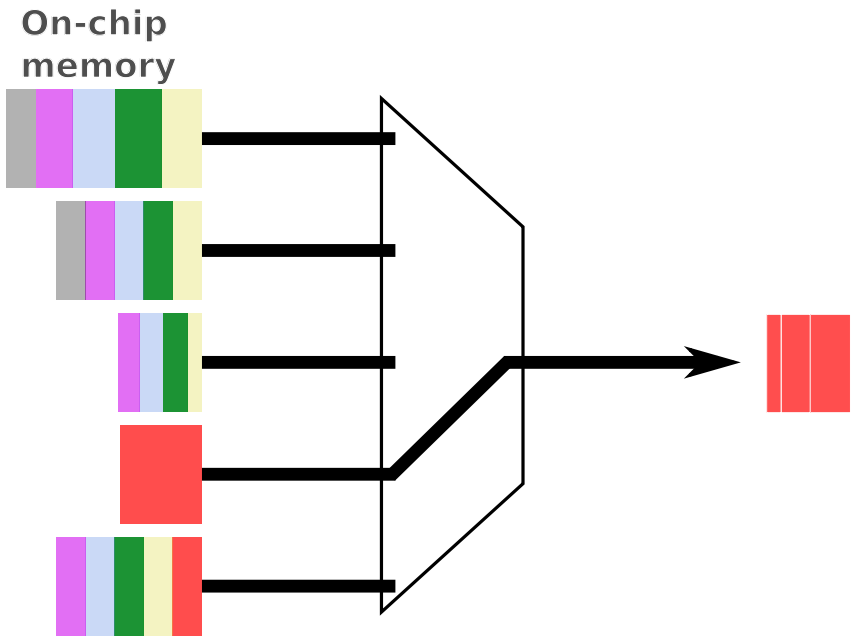
Institute of Particle and Nuclear Studies

November 30, 2022

# Motivation

# Shortcomings of the Current Event Builder

- Buffer data in FPGA's memory before event building
  - no external memory
  - small internal memory
- Events read sequentially
- Full event must be in memory before event builder can start data processing
- Sources of deadtime
  - large events
  - delayed events



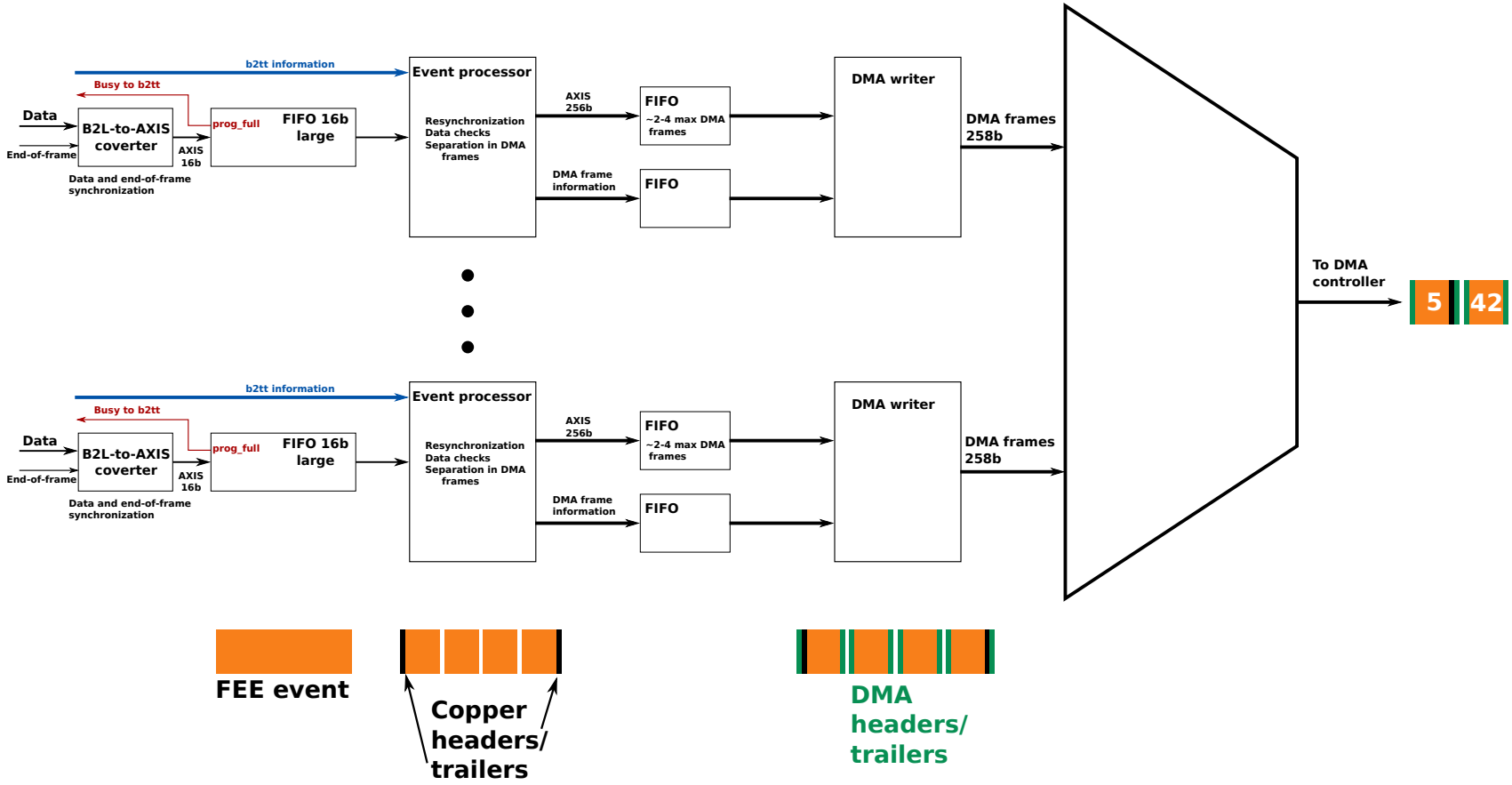
# New Event Building Scheme

- Main idea:
  1. write data directly to ROPC
  2. build events in software
- Non-sequential data transmission in firmware

$\Rightarrow$ **Software-assisted event builder**

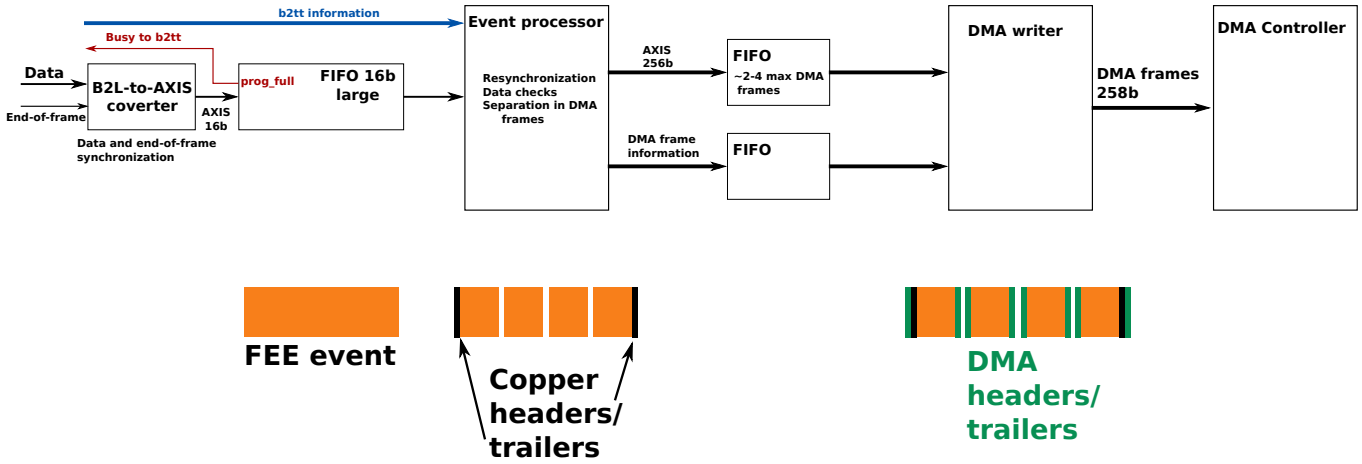
# PCIe40 Firmware

# User Logic Firmware Layout




- Channel-based data processing chain
- Multiplexer with interface to the DMA core

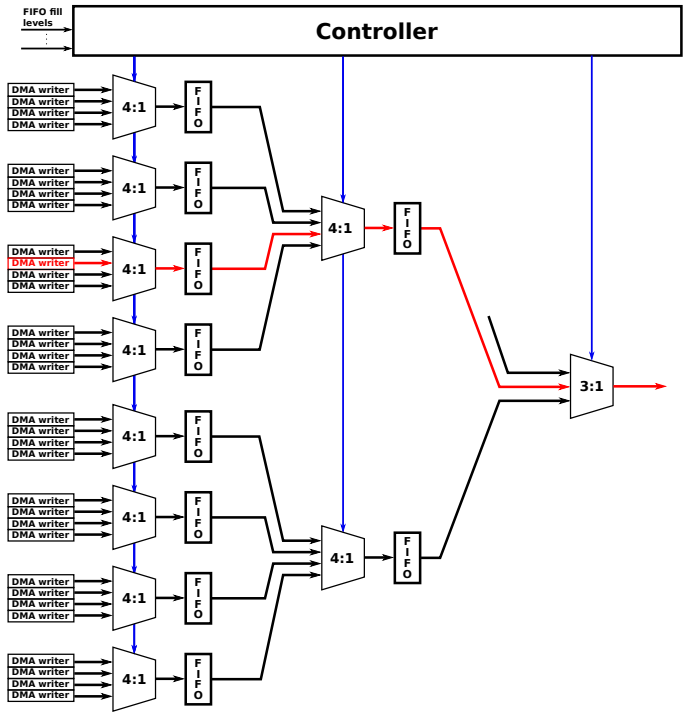
# Channel-based Data Processing Chain




- Interface conversion from B2L to AXI4Stream
- Event processor
  - data headers analysis
  - trigger number matching with **b2tt information**
  - synchronization recovery
  - separation of long frames into sub-frames
- DMA frames as units of data transmission
  - **no need to wait for end of event before starting data transmission**
- DMA writer
  - assign headers and trailers to the DMA frames

# Multiplexer

- Controller
  - selects channel with highest FIFO occupancy
  - **AND** DMA frames ready for transmission
  - changes channel once DMA frame transmission finishes
- Multiplexer structure
  - connects one DMA writer to the DMA controller
- 7 clock cycles needed to change channel

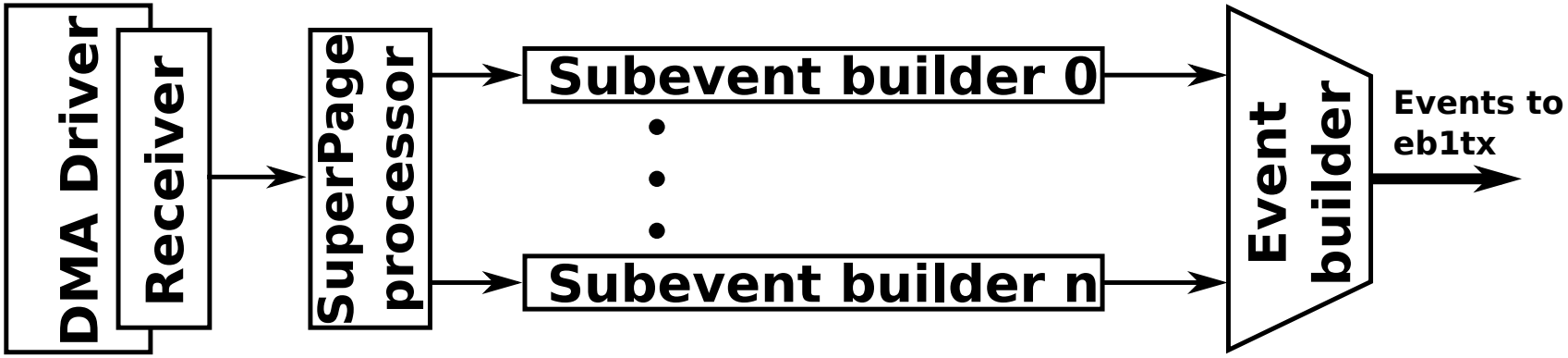


# Ressource Utilization

| Resource | Usage | % |
|---|---|---|
| Logic utilization (ALMs needed / total ALMs on device) | 202,339 / 427,200 | 47 % |
| ALMs needed [=A-B+C] | 202,339 | |
| [A] ALMs used in final placement [=a+b+c+d] | 264,769 / 427,200 | 62 % |
| [B] Estimate of ALMs recoverable by dense packing | 64,119 / 427,200 | 15 % |
| [C] Estimate of ALMs unavailable [=a+b+c+d] | 1,689 / 427,200 | < 1 % |
| | | |
| Difficulty packing design | Low | |
| | | |
| Total LABs: partially or completely used | 33,809 / 42,720 | 79 % |
| | | |
| Combinational ALUT usage for logic | 278,597 | |
| Combinational ALUT usage for route-throughs | 80,930 | |
| Memory ALUT usage | 8,280 | |
| | | |
| | | |
| Dedicated logic registers | 333,627 | |
| | | |
| ALMs adjustment for power estimation | 32,752 | |
| | | |
| Virtual pins | 0 | |
| I/O pins | 419 / 960 | 44 % |
| I/O registers | 2 | |
| | | |
| M20K blocks | 2,277 / 2,713 | 84 % |
| Total MLAB memory bits | 83,776 | |
| Total block memory bits | 34,923,372 / 55,562,240 | 63 % |
| Total block memory implementation bits | 46,632,960 / 55,562,240 | 84 % |

# Software

# Software Layout




- Driver
  - copy data from kernel to user space (**data copy!**)
- Superpage processor
  - divide superpage into DMA frames
- Subevent builder
  - combine DMA frames from the same channel to an event
  - calculate and compare CRC (**data access!**)
  - check data consistency
- Event builder
  - combine subevents to an event (**data copy!**)
  - send event to **eb1tx** over a **ZMQ socket**

# Performance and Limitations

# Effect of Core Frequency

- 35 channels with 5 kB/event running at 20 kHz (not fully optimized)
- 127 MHz

```
Clock FTSW
Face plate clock : 127216234 Hz
Run number : 224    Trigger tag : 10869716    DMA deadtime : 0.125452
Clock Up : OK  TTD Up : OK  Trigger type :  7 Trigger rate : 15100
Trigger counter : 3029
```

# Effect of Core Frequency

- 35 channels with 5 kB/event running at 20 kHz (not fully optimized)
- 127 MHz

```
Clock FTSW
Face plate clock : 127216234 Hz
Run number : 224    Trigger tag : 10869716    DMA deadtime : 0.125452
Clock Up : OK  TTD Up : OK  Trigger type :  7 Trigger rate : 15100
Trigger counter : 3029
```

- 150 MHz

```
Clock FTSW
Face plate clock : 127216238 Hz
Run number : 236    Trigger tag : 2112495     DMA deadtime : 0.146884
Clock Up : OK  TTD Up : OK  Trigger type :  7 Trigger rate : 17319
Trigger counter : 1020
```

  - further increase fails due to timing violations within FIFOs

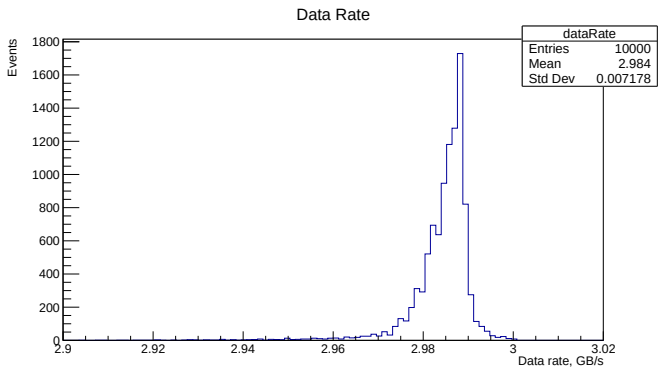
$$\frac{150}{127} \approx \frac{17319}{15100}$$

$\Rightarrow$ PCIe data transmission scales linearly with clock frequency

- Expected throughput at 150 MHz: 4.8 GB/s

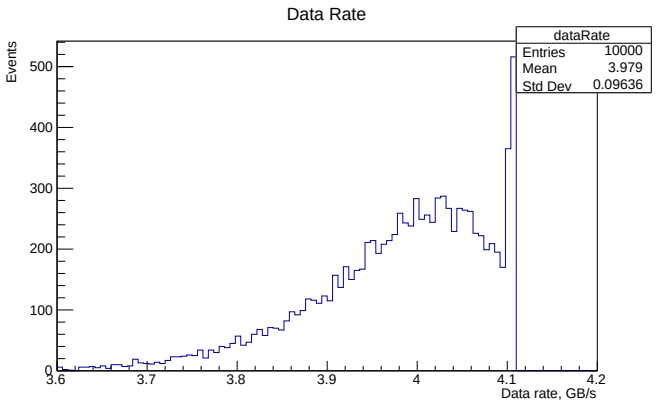
# Further Optimizations

- Full event builder software, no CRC calculation
- Throughput: 2.98 GB/s
  - far away from theoretical limit



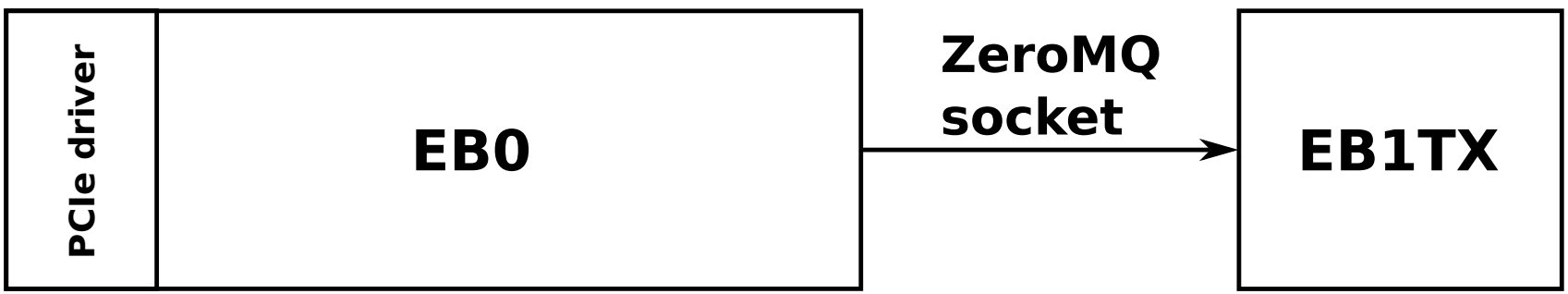
# Further Optimizations

- Full event builder software, no CRC calculation
- Throughput: 2.98 GB/s
  - far away from theoretical limit
- Too coarse sleep function in PCIe driver
  - **schedule()**: granularity 1 ms
  - replaced by **udelay(1)**: granularity 1 us, busy wait
- Throughput: 4 GB/s
  - 12 % inefficiency due to MUX switching



# Test with eb1tx

- Implemented event headers
- Data transmission over a single ZMQ push-pull socket
  - one event – one message



# Test with eb1tx

- Implemented event headers
- Data transmission over a single ZMQ push-pull socket
  - one event – one message
- Performance sufficient for 25 GB/s Ethernet

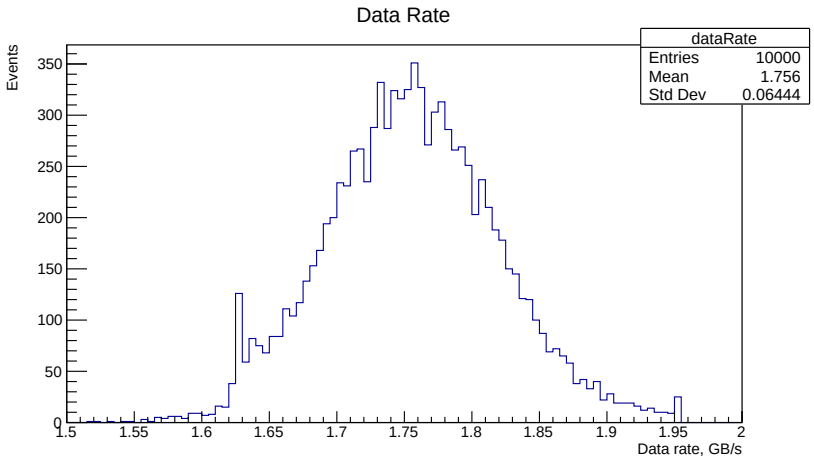



Figure: Data rate with eb1tx (with CRC checks)

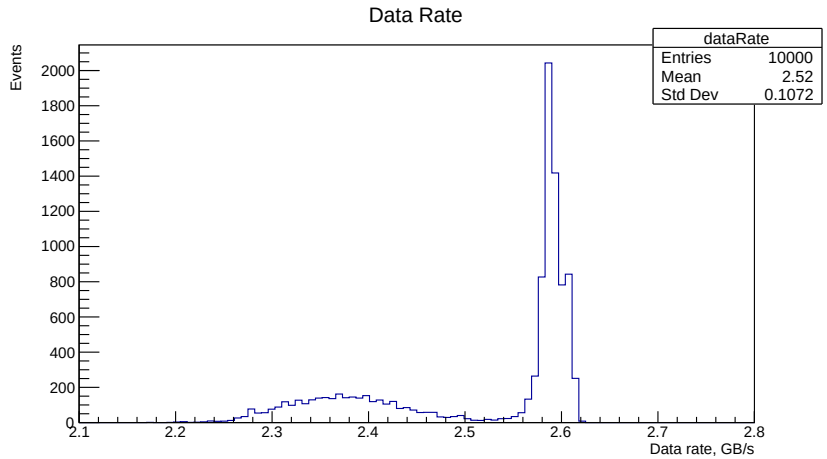
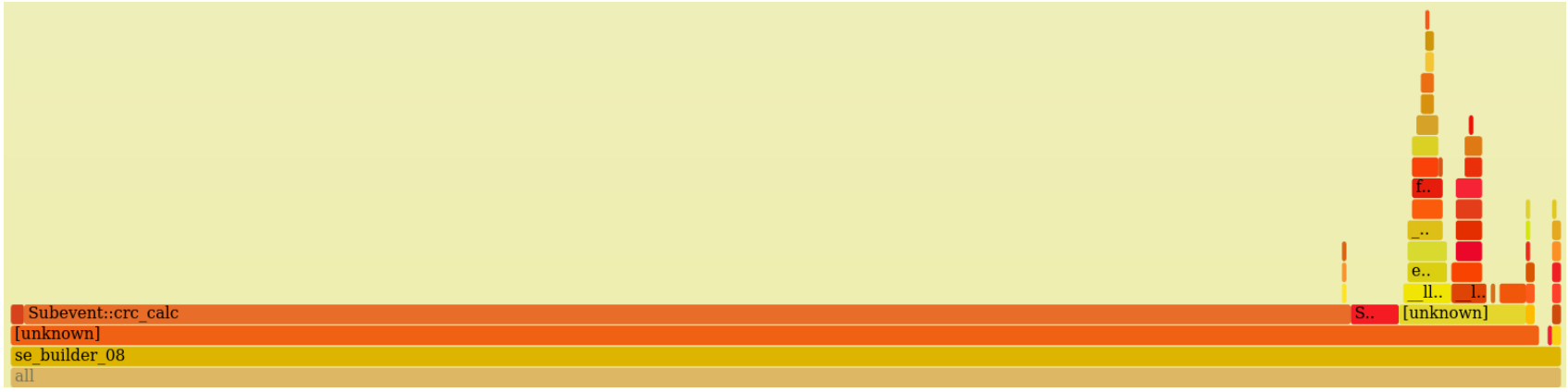



Figure: Data rate with eb1tx using 4 ZMQ contexts for internal data transmission (with CRC checks)

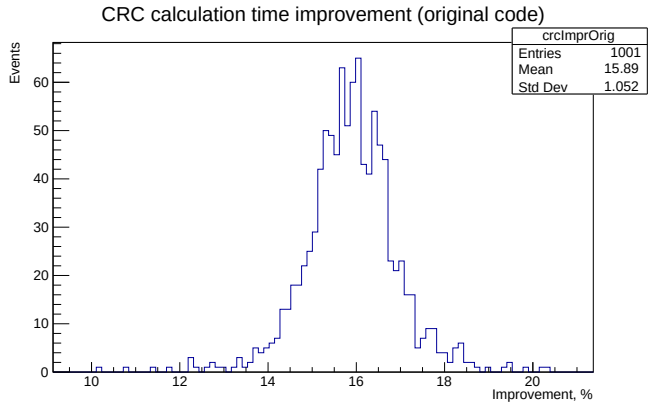
# CRC Optimization




- 85 % of subevent builder time taken by CRC calculation
  - bottleneck of the performance

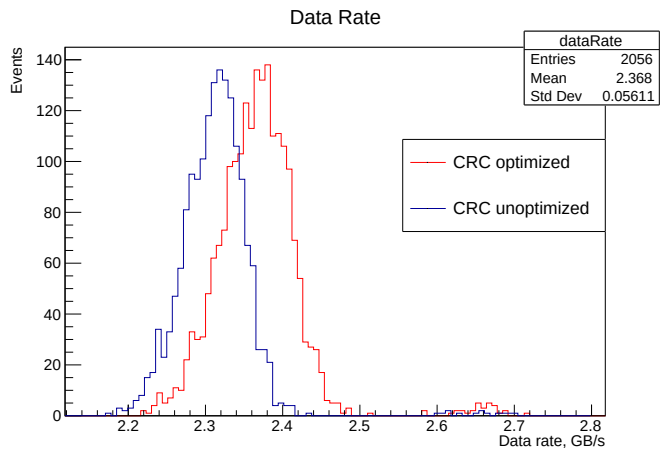
# CRC Optimization

- Reduce number of computations
  - 16 % improvement on standalone process



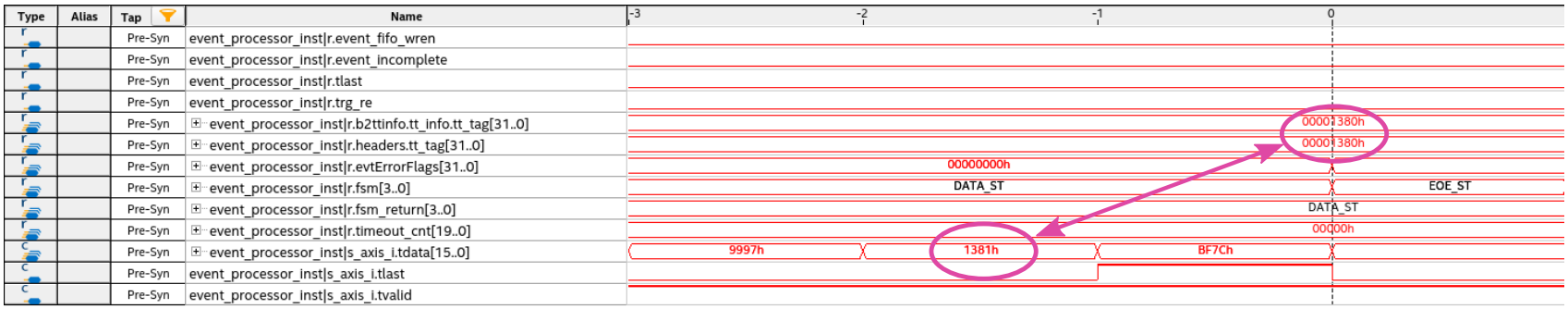
# CRC Optimization

- Reduce number of computations
  - 16 % improvement on standalone process
- 2 % improvement in event builder
  - performance limited not by CPU power
  - possible limitation by memory bandwidth



# PCIe40 Handling of Delayed Data

- In general, **no problems** with data delay up to 6 ms on one channel even at high trigger rates ( 22 kHz)
  - only if **far from throughput limit**
- **Data mismatch** if operated at throughput limit
- Event mixup in the channel with delayed data



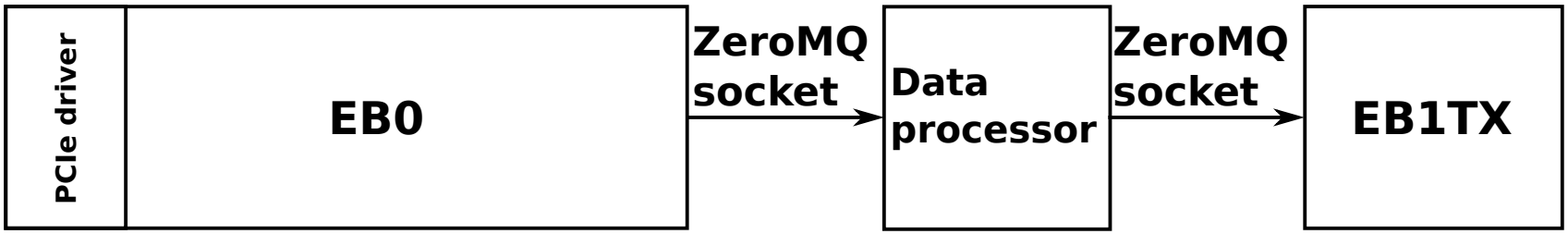

- Two events merged to one
  - lost last data word of the first event
- System stability depends on the error rate
  - can recover if slightly above throughput limit
  - no recovery if far above limit

# Remaining Problems

1. Delayed data sources may cause buffer overflow in PCIe40
   - if operated at throughput limit
   - details in my next talk
2. Handling of the end-of-run situation
   - DMA controller buffers frames to fill full superpage
   - data not transmitted if run stops in the middle of superpage
   - flush data out: use dummy channel ID(63) and discard data in software

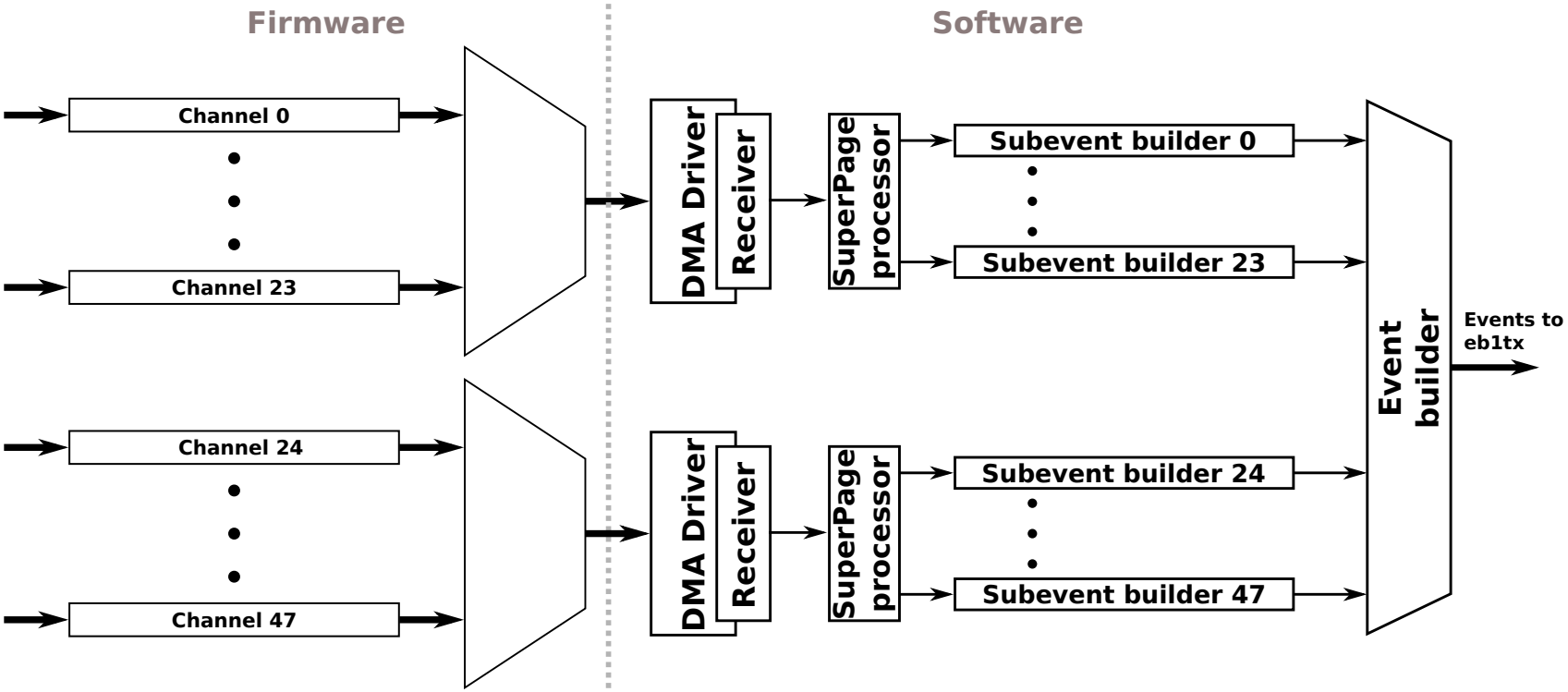
# System Extension

# Possible Extensions: Data Processing




- Optional data processing module as independent process
  - for example, TOP feature extraction
- ZeroMQ socket as interface to EB0 and EB1TX
  - stable interface
  - no modification to EB0 required
  - can be scaled to another PC if performance of ROPC not sufficient

# Possible Extensions: Second PCIe Interface

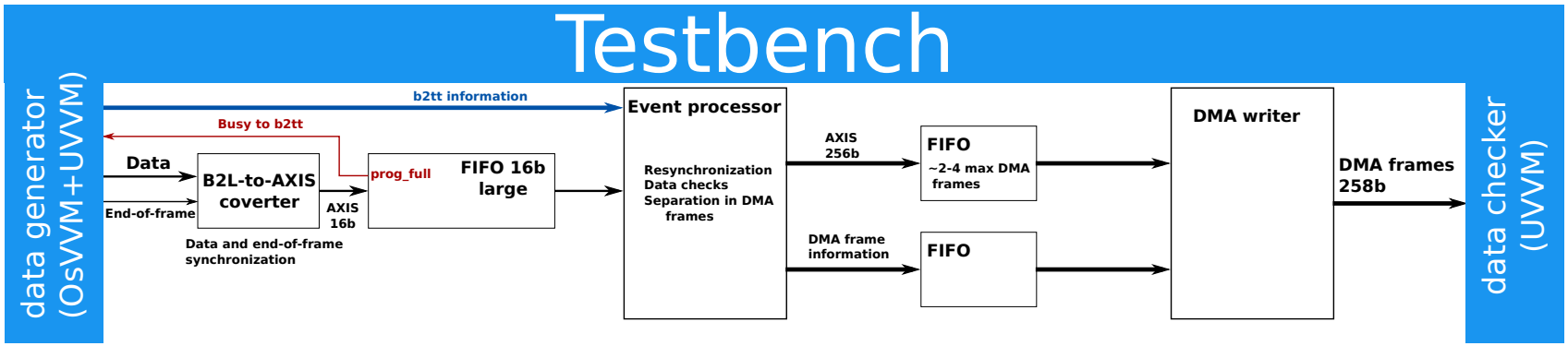



- Increase throughput with second PCIe interface
- Firmware and software easy to adapt

# Lessons Learned

# Lessons Learned

1. Use standard interfaces
   - AXI4Stream, ZMQ
   - for example, to connect single channel to the DMA controller
2. Use formal code verification



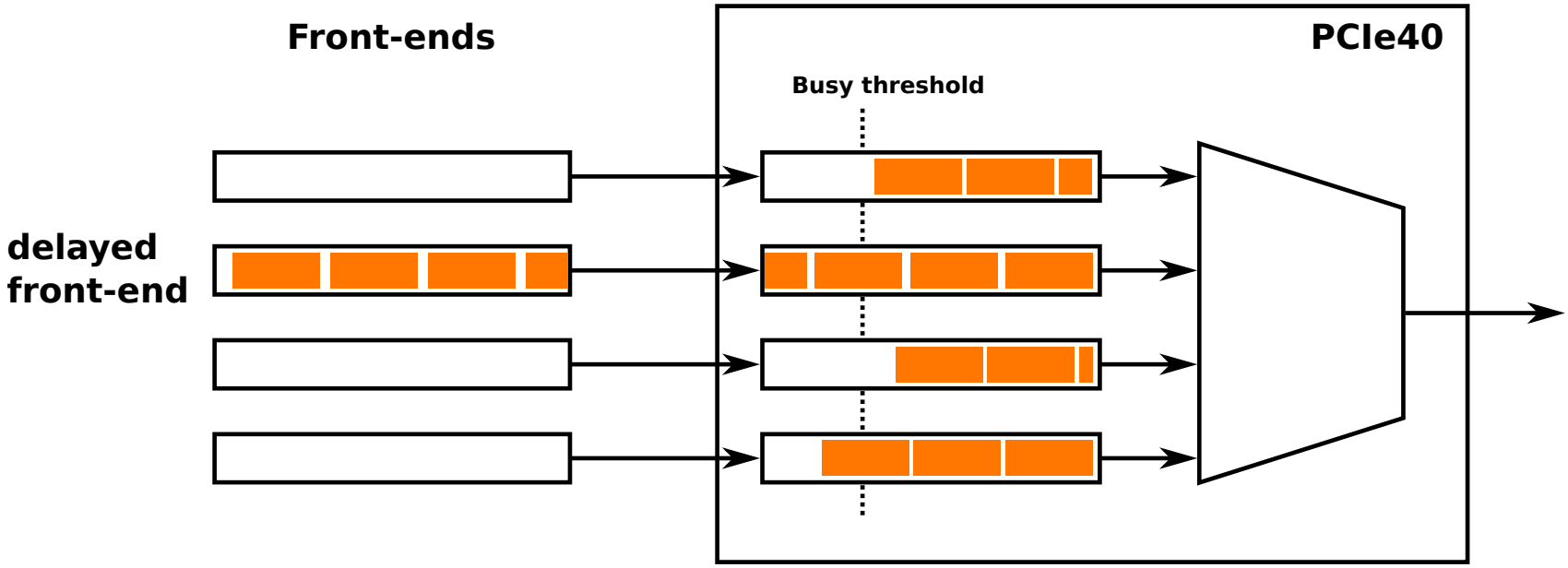

   - OsVVM for randomization and scoreboards
   - UVVM for AXI4Stream BFMs
   - after passing verification, code "just works"

# Summary

- Software-assisted event builder designed and tested at B4 testbench
  - solves the problem with large events
  - no principal problem with delayed data
  - performance bottlenecks understood
- Performance of the system measured and bottlenecks identified
  - **2.4 GB/s** sustained data rate with CRC calculation
  - **4 GB/s** sustained data rate without CRC calculation
- Proposed **modular system extension**

# Backup Slides

# Backpressure Problem




- Data in front-ends after busy issued
- Starts at constant delay if event size is constant
  - time needed to transmit event

# Backup Slides: Memory bandwidth

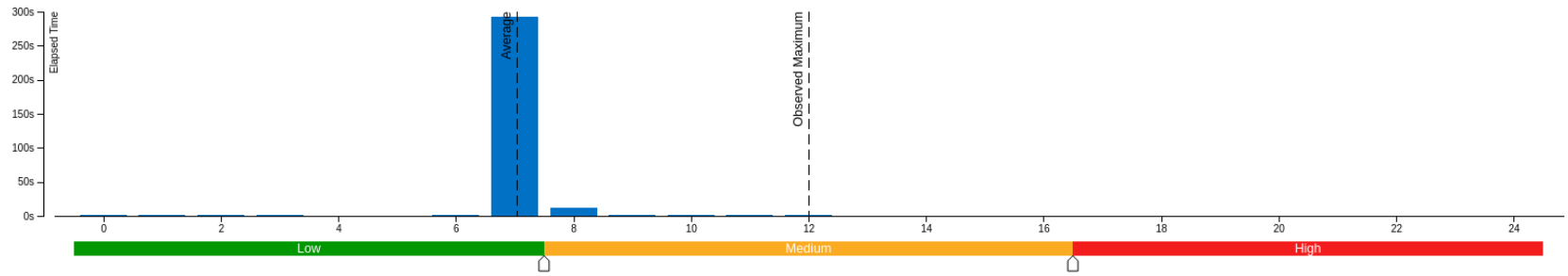



Figure: Without CRC calculation

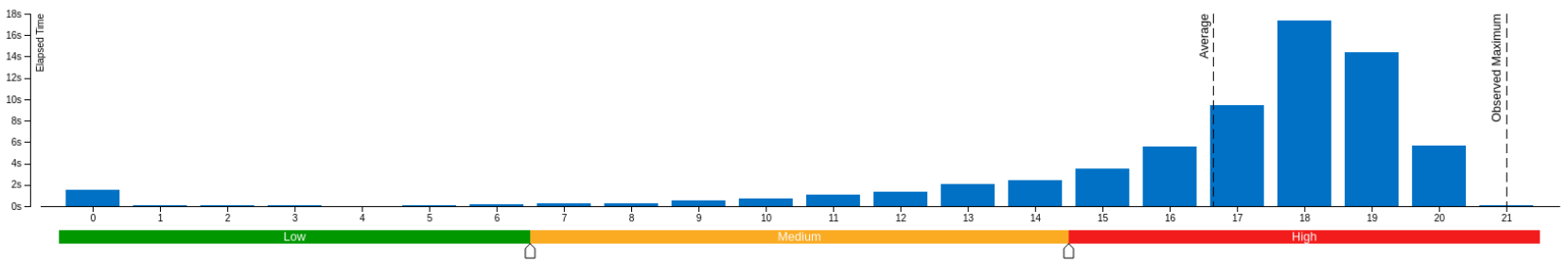



Figure: With CRC calculation

# Profiling of the EB0

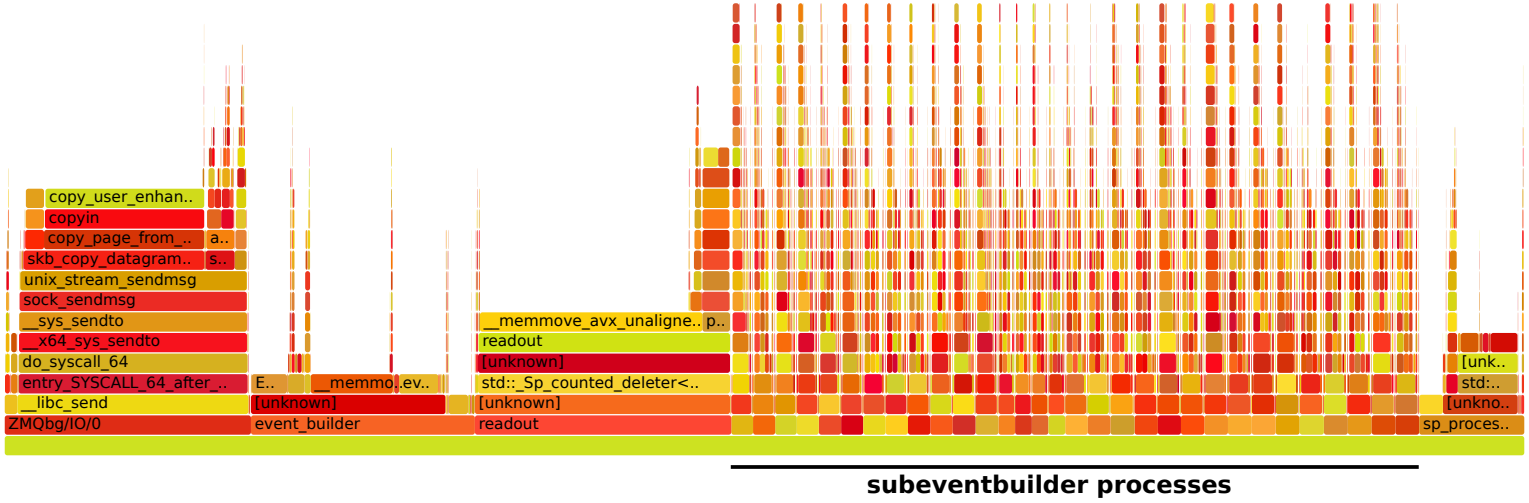



Figure: Without CRC calculation

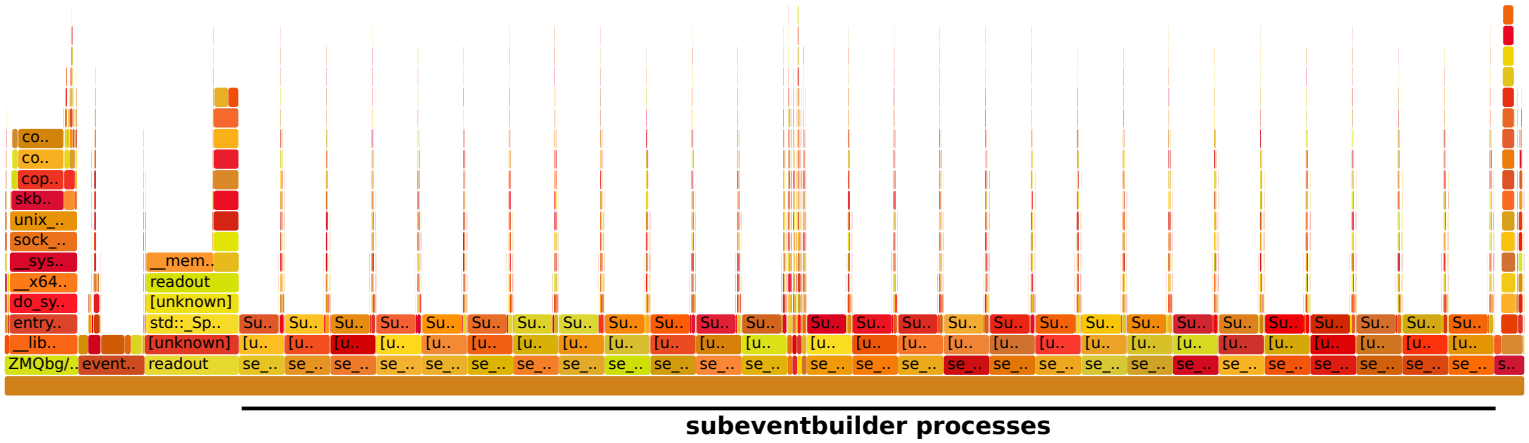



Figure: With CRC calculation