

SpaceWire Electronic Ground Support Equipment (EGSE)

Stephen Mudie, Paul E McKechnie,
Steve Parkes, Martin Dunstan



- What is the SpaceWire EGSE?
- How does it work?
 - Hardware
 - Software
- Scripting Language
- Capabilities/Benefits
- Examples



- What it is: SpaceWire test and development unit developed by STAR-Dundee
- Purpose: Simulate instruments or other SpaceWire equipment in real time during testing and integration
- Generates user defined packets in pre-defined sequences at specific times and data rates
 - i.e. packet 1 followed by packet 2 10ms later at 100Mbps

STAR-Dundee SpaceWire EGSE Hardware



- 2 SpaceWire ports
- 4 External Triggers (3 IN, 1 OUT)
- Indicator LEDs
- 128MB Memory
- USB connection to host PC



- **Compiler**
 - Compiles scripts into EGSE configuration files
- **Configurator**
 - Loads EGSE configuration files onto hardware
- **Software API**
 - Can be used to interact with EGSE whilst in operation

- Packet Definition
- Variables
 - Used to define packets with dynamic data
- Packet Generation Schedules
- State Machines
 - Control current packet generation schedule
- Events
 - State machines reacts to events



- Format
 - Header, body, footer

```
packet myPkt
    hex(0A 0B 0C 0D)
    eop
end packet
```



- **Format**
 - Header, body, footer
- **Data defined in hex or decimal**

```
packet myPkt
    hex(0A 0B 0C 0D)
    dec(1 2 3 4)
    eep
end packet
```




- **Format**
 - Header, body, footer
- **Data defined in hex or decimal**
- **EOP and EEP control characters**

```
packet myPkt
    hex(0A 0B 0C 0D)
    dec(1 2 3 4)
    eop
end packet
```



- **Format**
 - Header, body, footer
- **Data defined in hex or decimal**
- **EOP and EEP control characters**
- **CRC and checksum calculations**

```
packet myPkt
    start(crc8)
    hex(0A 0B 0C 0D)
    dec(1 2 3 4)
    stop(crc8)
    crc8
    eep
end packet
```



- Used to define packets with dynamic data
- Referenced in packet definition
- Types:
 - Increment

```
variables
    transactionID inc8 = 0
end variables

packet myPkt
    hex(0A 0B 0C 0D)
    transactionID
    eop
end packet
```



- Used to define packets with dynamic data
- Referenced in packet definition
- Types:
 - Increment
 - Decrement

```
variables
    myDecVar dec8 = 10
end variables

packet myPkt
    hex(0A 0B 0C 0D)
    myDecVar
    eop
end packet
```



- Used to define packets with dynamic data
- Referenced in packet definition
- Types:
 - Increment
 - Decrement
 - Rotate Right

```
variables
    myRRVar ror8 = 1
end variables

packet myPkt
    hex(0A 0B 0C 0D)
    myRRVar
    eop
end packet
```



- Used to define packets with dynamic data
- Referenced in packet definition
- Types:
 - Increment
 - Decrement
 - Rotate Right
 - Rotate Left

```
variables
    myRLVar rol8 = 1
end variables

packet myPkt
    hex(0A 0B 0C 0D)
    myRLVar
    eop
end packet
```



- Used to define packets with dynamic data
- Referenced in packet definition
- Types:
 - Increment
 - Decrement
 - Rotate Right
 - Rotate Left
 - Random

```
variables
    myRandVar rnd = 0
end variables

packet myPkt
    hex(0A 0B 0C 0D)
    myRandVar
    eop
end packet
```



- Send pre-defined packets at specific times

```
packet myPkt1
    hex(0A 0B 0C 0D)
    eop
end packet
...

schedule mySchedule
    send myPkt1
    send myPkt2
end schedule
```




- Send pre-defined packets at specific times
- Specify the number of times to send packets

```
packet myPkt1
    hex(0A 0B 0C 0D)
    eop
end packet
...

schedule mySchedule
    send myPkt1 * 2
    send myPkt2
end schedule
```



- Send pre-defined packets at specific times
- Specify the number of times to send packets
- Timing
 - Relative to schedule start

```
packet myPkt1
    hex(0A 0B 0C 0D)
    eop
end packet
...

schedule mySchedule
    5ms send myPkt1
    send myPkt2
end schedule
```



- Send pre-defined packets at specific times
- Specify the number of times to send packets
- Timing
 - Relative to schedule start
 - Relative to previous packet

```
packet myPkt1
    hex(0A 0B 0C 0D)
    eop
end packet
...

schedule mySchedule
    5ms send myPkt1
    +5ms send myPkt2
end schedule
```

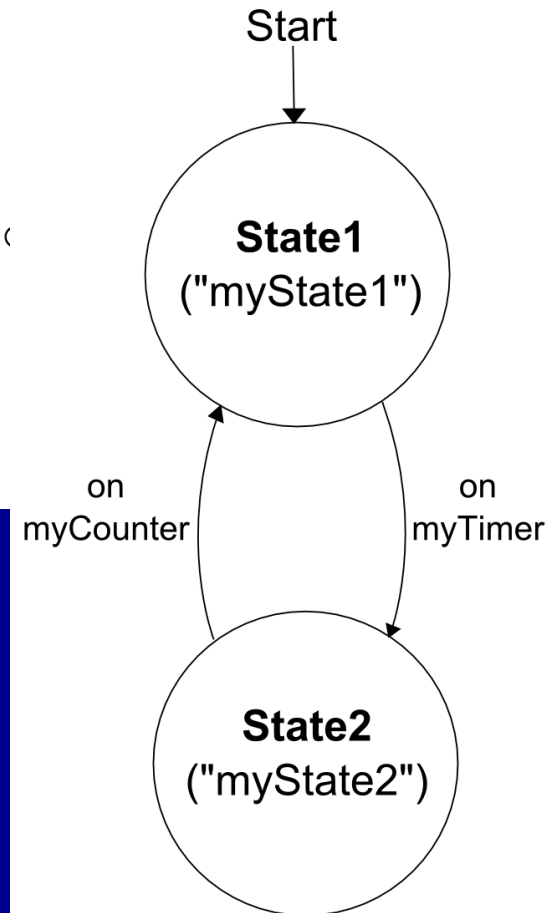


- Controls the EGSE state
- One state machine per SpaceWire interface
- Consists of state definitions



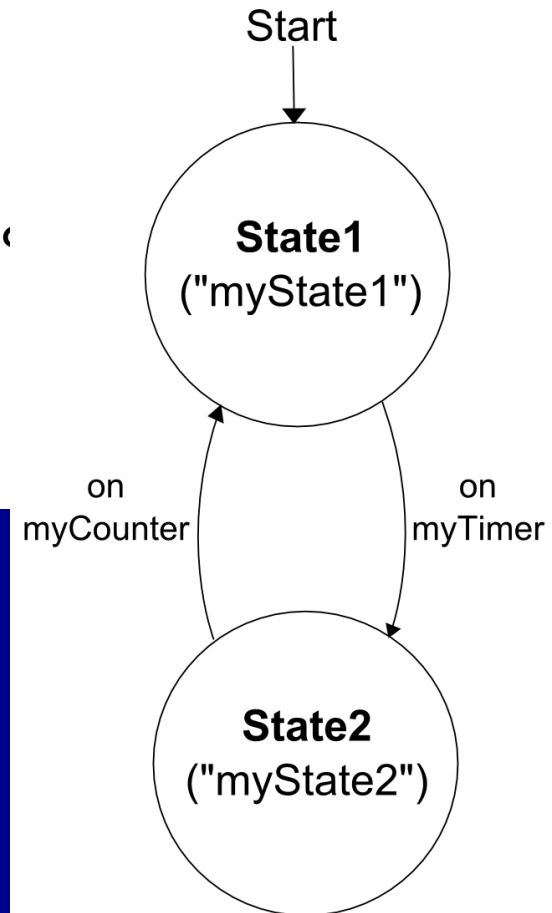
```
statemachine 1
  initial state myState1
    do mySchedule1 @ 20Mbps
    transition at end of schedule
    on myTimer goto myState2
  end state
  state myState2
    do mySchedule2 @ 50Mbps repeatedly
    transition at end of packet
    on myCounter goto myState1
  end state
end statemachine
```

```
statemachine 1
  initial state myState1
    do mySchedule1 @ 20Mbps
    transition at end of schedule
    on myTimer goto myState2
  end state
  state myState2
    do mySchedule2 @ 50Mbps repeated
    transition at end of packet
    on myCounter goto myState1
  end state
end statemachine
```



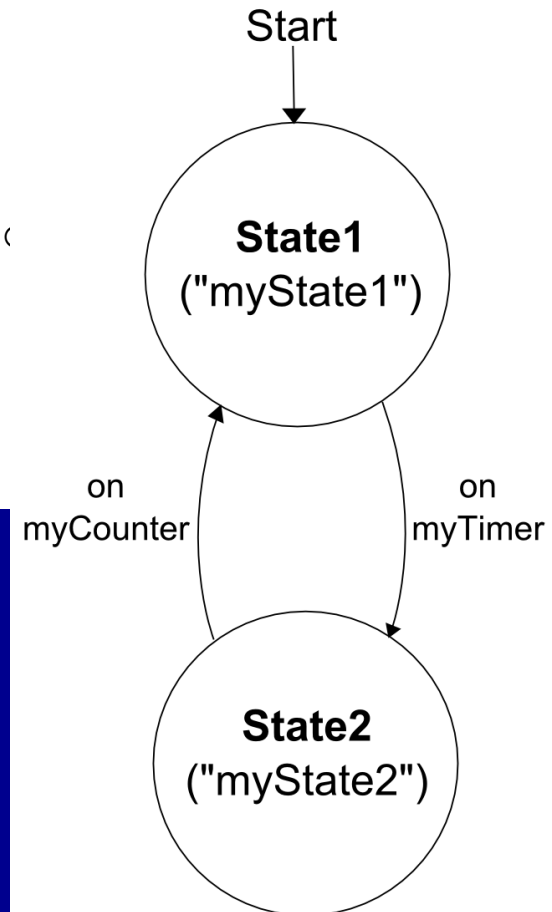
```
statemachine 1
  initial state myState1
    do mySchedule1 @ 20Mbps
    transition at end of schedule
    on myTimer goto myState2
  end state
  state myState2
    do mySchedule2 @ 50Mbps repeated
    transition at end of packet
    on myCounter goto myState1
  end state
end statemachine
```

- Each state has
 - a schedule



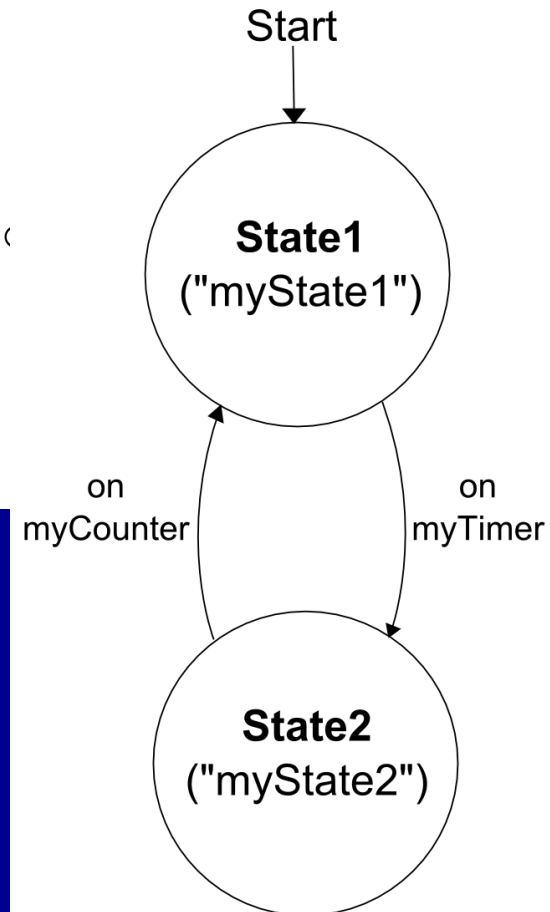
```
statemachine 1
  initial state myState1
    do mySchedule1 @ 20Mbps
    transition at end of schedule
    on myTimer goto myState2
  end state
  state myState2
    do mySchedule2 @ 50Mbps repeated
    transition at end of packet
    on myCounter goto myState1
  end state
end statemachine
```

- Each state has
 - a schedule
 - conditions under which to change state




```
statemachine 1
  initial state myState1
    do mySchedule1 @ 20Mbps
    transition at end of schedule
    on myTimer goto myState2
  end state
  state myState2
    do mySchedule2 @ 50Mbps repeated
    transition at end of packet
    on myCounter goto myState1
  end state
end statemachine
```

- Each state has
 - a schedule
 - conditions under which to change state
 - when to change state



- State machines react to events, changing state and therefore schedule
- Pre-defined events
 - Include: Link started, link errors, time-code received and packet generation events
- User defined events

- State machine reacts to events, changing state and therefore schedule
- Pre-defined events
 - Include: Link started, link errors, time-code received and packet generation events
- User defined events
 - Timers

```
timers
    myTimer 10ms start on mySWEvent
end timers
```

- State machine reacts to events, changing state and therefore schedule
- Pre-defined events
 - Include: Link started, link errors, time-code received and packet generation events
- User defined events
 - Timers
 - Counters

```
counters
    myCounter 10 on myTrigIn1
end counters
```

- State machine reacts to events, changing state and therefore schedule
- Pre-defined events
 - Include: Link started, link errors, time-code received and packet generation events
- User defined events
 - Timers
 - Counters
 - Software

```
software
    mySWEvent1 1
end software
```

- State machine reacts to events, changing state and therefore schedule
- Pre-defined events
 - Include: Link started, link errors, time-code received and packet generation events
- User defined events
 - Timers
 - Counters
 - Software
 - External triggers

```
triggers
    myTrigIn1 input 1 rising
    output high on myTimer
end software
```



- Requirements
 - Simulate an instrument sending house keeping information, consisting of four pre-defined packets, every two seconds at 200Mbps.

SpaceWire EGSE Example 1

SpaceWire Link Analyser MK2 (Serial Number: 35100030) - [~ ~ Not Saved ~ ~]

File View Find Settings Trigger Help

| Time From Trigger | Time Delta | End A | End A Delta | End B | End B Delta |
|-------------------|--------------|----------------------|--------------|-------|-------------|
| 0 ns | | Header: 01 | | | |
| 50 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 520 ns | 470 ns | EOP | 470 ns | | |
| 570 ns | 50 ns | Header: 02 | 50 ns | | |
| 620 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.090 µs | 470 ns | EOP | 470 ns | | |
| 1.140 µs | 50 ns | Header: 03 | 50 ns | | |
| 1.190 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.660 µs | 470 ns | EOP | 470 ns | | |
| 1.710 µs | 50 ns | Header: 04 | 50 ns | | |
| 1.760 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 2.230 µs | 470 ns | EOP | 470 ns | | |
| 1.99999683 s | 1.9999946 s | Header: 01 | 1.9999946 s | | |
| 1.99999688 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999735 s | 470 ns | EOP | 470 ns | | |
| 1.9999974 s | 50 ns | Header: 02 | 50 ns | | |
| 1.99999745 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999792 s | 470 ns | EOP | 470 ns | | |
| 1.99999797 s | 50 ns | Header: 03 | 50 ns | | |
| 1.99999802 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999849 s | 470 ns | EOP | 470 ns | | |
| 1.99999854 s | 50 ns | Header: 04 | 50 ns | | |
| 1.99999859 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999906 s | 470 ns | EOP | 470 ns | | |
| 3.99999367 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |
| 3.99999372 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999419 s | 470 ns | EOP | 470 ns | | |
| 3.99999424 s | 50 ns | Header: 02 | 50 ns | | |
| 3.99999429 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999476 s | 470 ns | EOP | 470 ns | | |
| 3.99999481 s | 50 ns | Header: 03 | 50 ns | | |
| 3.99999486 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999533 s | 470 ns | EOP | 470 ns | | |
| 3.99999538 s | 50 ns | Header: 04 | 50 ns | | |
| 3.99999543 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.9999959 s | 470 ns | EOP | 470 ns | | |
| 5.99999042 s | 1.99999452 s | Header: 01 | 1.99999452 s | | |
| 5.99999047 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999094 s | 470 ns | EOP | 470 ns | | |
| 5.99999099 s | 50 ns | Header: 02 | 50 ns | | |
| 5.99999104 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999151 s | 470 ns | EOP | 470 ns | | |
| 5.99999156 s | 50 ns | Header: 03 | 50 ns | | |
| 5.99999161 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999208 s | 470 ns | EOP | 470 ns | | |
| 5.99999213 s | 50 ns | Header: 04 | 50 ns | | |
| 5.99999218 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999265 s | 470 ns | EOP | 470 ns | | |
| 7.99998726 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |

Expand All Collapse All

Format: Data Protocol ASCII Integer Hex

Data View: ASCII Integer Hex

Bit Width: 8 Bit (Byte) 32 Bit (DWord) LSB First 16 Bit (Word) 64 Bit (QWord)

Byte Settings: Bytes Per Row 8

Character Display Packet Display Bit-Stream Display

SpaceWire EGSE Example 1

SpaceWire Link Analyser MK2 (Serial Number: 35100030) - [~ ~ Not Saved ~ ~]

File View Find Settings Trigger Help

| Time From Trigger | Time Delta | End A | End A Delta | End B | End B Delta |
|-------------------|--------------|----------------------|--------------|-------|-------------|
| 0 ns | | Header: 01 | | | |
| 50 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 520 ns | 470 ns | EOP | 470 ns | | |
| 570 ns | 50 ns | Header: 02 | 50 ns | | |
| 620 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.090 µs | 470 ns | EOP | 470 ns | | |
| 1.140 µs | 50 ns | Header: 03 | 50 ns | | |
| 1.190 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.660 µs | 470 ns | EOP | 470 ns | | |
| 1.710 µs | 50 ns | Header: 04 | 50 ns | | |
| 1.760 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 2.230 µs | 470 ns | EOP | 470 ns | | |
| 1.99999683 s | 1.9999946 s | Header: 01 | 1.9999946 s | | |
| 1.99999688 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999735 s | 470 ns | EOP | 470 ns | | |
| 1.9999974 s | 50 ns | Header: 02 | 50 ns | | |
| 1.99999745 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999792 s | 470 ns | EOP | 470 ns | | |
| 1.99999797 s | 50 ns | Header: 03 | 50 ns | | |
| 1.99999802 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999849 s | 470 ns | EOP | 470 ns | | |
| 1.99999854 s | 50 ns | Header: 04 | 50 ns | | |
| 1.99999859 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999906 s | 470 ns | EOP | 470 ns | | |
| 3.99999367 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |
| 3.99999372 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999419 s | 470 ns | EOP | 470 ns | | |
| 3.99999424 s | 50 ns | Header: 02 | 50 ns | | |
| 3.99999429 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999476 s | 470 ns | EOP | 470 ns | | |
| 3.99999481 s | 50 ns | Header: 03 | 50 ns | | |
| 3.99999486 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999533 s | 470 ns | EOP | 470 ns | | |
| 3.99999538 s | 50 ns | Header: 04 | 50 ns | | |
| 3.99999543 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.9999959 s | 470 ns | EOP | 470 ns | | |
| 5.99999042 s | 1.99999452 s | Header: 01 | 1.99999452 s | | |
| 5.99999047 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999094 s | 470 ns | EOP | 470 ns | | |
| 5.99999099 s | 50 ns | Header: 02 | 50 ns | | |
| 5.99999104 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999151 s | 470 ns | EOP | 470 ns | | |
| 5.99999156 s | 50 ns | Header: 03 | 50 ns | | |
| 5.99999161 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999208 s | 470 ns | EOP | 470 ns | | |
| 5.99999213 s | 50 ns | Header: 04 | 50 ns | | |
| 5.99999218 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999265 s | 470 ns | EOP | 470 ns | | |
| 7.99998726 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |

200Mhz

Expand All Collapse All

Format: Data Protocol ASCII Integer Hex

Data View: ASCII Integer Hex

Bit Width: 8 Bit (Byte) 32 Bit (DWord) LSB First 16 Bit (Word) 64 Bit (QWord)

Byte Settings: Bytes Per Row 8

Character Display Packet Display Bit-Stream Display

Complete

End A: 200.002 MHz End B: 200.002 MHz



SpaceWire EGSE Example 1

SpaceWire Link Analyser MK2 (Serial Number: 35100030) - [~ ~ Not Saved ~ ~]

File View Find Settings Trigger Help

| Time From Trigger | Time Delta | End A | End A Delta | End B | End B Delta |
|-------------------|--------------|----------------------|--------------|-------|-------------|
| 0 ns | | Header: 01 | | | |
| 50 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 520 ns | 470 ns | EOP | 470 ns | | |
| 570 ns | 50 ns | Header: 02 | 50 ns | | |
| 620 ns | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.090 µs | 470 ns | EOP | 470 ns | | |
| 1.140 µs | 50 ns | Header: 03 | 50 ns | | |
| 1.190 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.660 µs | 470 ns | EOP | 470 ns | | |
| 1.710 µs | 50 ns | Header: 04 | 50 ns | | |
| 1.760 µs | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 2.230 µs | 470 ns | EOP | 470 ns | | |
| 1.99999683 s | 1.9999946 s | Header: 01 | 1.9999946 s | | |
| 1.99999688 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999735 s | 470 ns | EOP | 470 ns | | |
| 1.9999974 s | 50 ns | Header: 02 | 50 ns | | |
| 1.99999745 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999792 s | 470 ns | EOP | 470 ns | | |
| 1.99999797 s | 50 ns | Header: 03 | 50 ns | | |
| 1.99999802 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999849 s | 470 ns | EOP | 470 ns | | |
| 1.99999854 s | 50 ns | Header: 04 | 50 ns | | |
| 1.99999859 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 1.99999906 s | 470 ns | EOP | 470 ns | | |
| 3.99999367 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |
| 3.99999372 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999419 s | 470 ns | EOP | 470 ns | | |
| 3.99999424 s | 50 ns | Header: 02 | 50 ns | | |
| 3.99999429 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999476 s | 470 ns | EOP | 470 ns | | |
| 3.99999481 s | 50 ns | Header: 03 | 50 ns | | |
| 3.99999486 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.99999533 s | 470 ns | EOP | 470 ns | | |
| 3.99999538 s | 50 ns | Header: 04 | 50 ns | | |
| 3.99999543 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 3.9999959 s | 470 ns | EOP | 470 ns | | |
| 5.99999042 s | 1.99999452 s | Header: 01 | 1.99999452 s | | |
| 5.99999047 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999094 s | 470 ns | EOP | 470 ns | | |
| 5.99999099 s | 50 ns | Header: 02 | 50 ns | | |
| 5.99999104 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999151 s | 470 ns | EOP | 470 ns | | |
| 5.99999156 s | 50 ns | Header: 03 | 50 ns | | |
| 5.99999161 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999208 s | 470 ns | EOP | 470 ns | | |
| 5.99999213 s | 50 ns | Header: 04 | 50 ns | | |
| 5.99999218 s | 50 ns | Cargo Size: 10 bytes | 50 ns | | |
| 5.99999265 s | 470 ns | EOP | 470 ns | | |
| 7.99998726 s | 1.99999461 s | Header: 01 | 1.99999461 s | | |

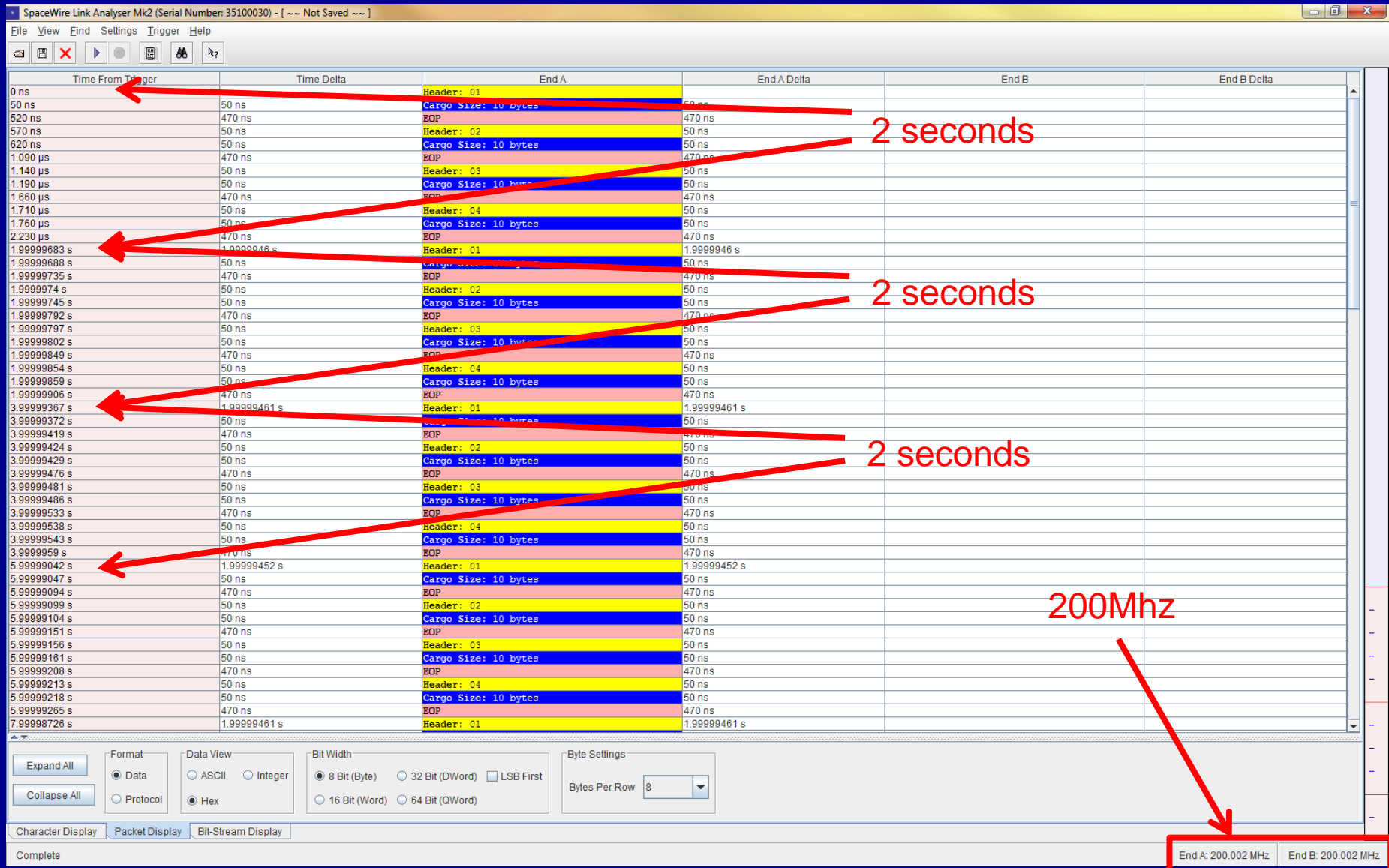
2 seconds

200Mhz

End A: 200.002 MHz End B: 200.002 MHz



SpaceWire EGSE Example 1





- Requirements
 - Simulate an instrument continuously sending packets in both directions over a SpaceWire link at 350Mbps.



STAR-Dundee

SpaceWire EGSE Example 2

SpaceWire Link Analyser Mk2 (Serial Number: 35100030) - [~ ~ Not Saved ~ ~]

File View Find Settings Trigger Help

| Time From Trigger | Time Delta | End A Event | End A Error | End A Delta | End B Event | End B Error | End B Delta |
|-------------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1.530 µs | 20 ns | | | | NCHAR [00] | | 20 ns |
| 1.540 µs | 10 ns | NCHAR [00] | | 30 ns | | | |
| 1.560 µs | 20 ns | | | | NCHAR [02] | | 30 ns |
| 1.570 µs | 10 ns | NCHAR [00] | | 30 ns | | | |
| 1.590 µs | 20 ns | | | | NCHAR [00] | | 30 ns |
| 1.600 µs | 10 ns | NCHAR [04] | | 30 ns | | | |
| 1.620 µs | 20 ns | | | | NCHAR [00] | | 30 ns |
| 1.630 µs | 10 ns | NCHAR [FF] | | 30 ns | | | |
| 1.640 µs | 10 ns | EOP | | 10 ns | | | |
| 1.650 µs | 10 ns | | | | NCHAR [00] | | 30 ns |
| 1.670 µs | 20 ns | NCHAR [FE] | | 30 ns | | | |
| 1.680 µs | 10 ns | | | | NCHAR [00] | | 30 ns |
| 1.690 µs | 10 ns | NCHAR [01] | | 20 ns | | | |
| 1.710 µs | 20 ns | FCT | | 20 ns | NCHAR [00] | | 30 ns |
| 1.730 µs | 20 ns | NCHAR [48] | | 20 ns | NCHAR [00] | | 20 ns |
| 1.750 µs | 20 ns | | | | FCT | | 20 ns |
| 1.760 µs | 10 ns | NCHAR [20] | | 30 ns | | | |
| 1.770 µs | 10 ns | | | | NCHAR [00] | | 20 ns |
| 1.790 µs | 20 ns | NCHAR [FE] | | 30 ns | | | |
| 1.800 µs | 10 ns | | | | NCHAR [04] | | 30 ns |
| 1.820 µs | 20 ns | NCHAR [00] | | 30 ns | | | |
| 1.830 µs | 10 ns | | | | NCHAR [FF] | | 30 ns |
| 1.840 µs | 10 ns | | | | EOP | | 10 ns |
| 1.850 µs | 10 ns | NCHAR [03] | | 30 ns | | | |
| 1.870 µs | 20 ns | | | | NCHAR [FE] | | 30 ns |
| 1.880 µs | 10 ns | NCHAR [00] | | 30 ns | | | |
| 1.900 µs | 20 ns | | | | NCHAR [01] | | 30 ns |
| 1.910 µs | 10 ns | NCHAR [00] | | 30 ns | | | |
| 1.930 µs | 20 ns | NCHAR [00] | | 20 ns | NCHAR [48] | | 30 ns |
| 1.950 µs | 20 ns | FCT | | 20 ns | | | |
| 1.960 µs | 10 ns | | | | NCHAR [20] | | 30 ns |
| 1.970 µs | 10 ns | NCHAR [00] | | 20 ns | FCT | | 10 ns |
| 2.000 µs | 30 ns | NCHAR [00] | | 30 ns | NCHAR [FE] | | 30 ns |
| 2.030 µs | 30 ns | NCHAR [00] | | 30 ns | NCHAR [00] | | 30 ns |
| 2.050 µs | 20 ns | | | | NCHAR [03] | | 20 ns |
| 2.060 µs | 10 ns | NCHAR [00] | | 30 ns | | | |
| 2.080 µs | 20 ns | | | | NCHAR [00] | | 30 ns |
| 2.090 µs | 10 ns | NCHAR [04] | | 30 ns | | | |
| 2.110 µs | 20 ns | | | | NCHAR [00] | | 30 ns |
| 2.120 µs | 10 ns | NCHAR [D3] | | 30 ns | | | |
| 2.130 µs | 10 ns | EOP | | 10 ns | | | |
| 2.140 µs | 10 ns | | | | NCHAR [00] | | 30 ns |
| 2.160 µs | 20 ns | NCHAR [FE] | | 30 ns | | | |
| 2.170 µs | 10 ns | FCT | | 10 ns | NCHAR [00] | | 30 ns |
| 2.200 µs | 30 ns | NCHAR [01] | | 30 ns | NCHAR [00] | | 30 ns |
| 2.210 µs | 10 ns | | | | FCT | | 10 ns |
| 2.230 µs | 20 ns | NCHAR [48] | | 30 ns | | | |
| 2.240 µs | 10 ns | | | | NCHAR [00] | | 30 ns |
| 2.250 µs | 10 ns | NCHAR [20] | | 20 ns | | | |
| 2.270 µs | 20 ns | | | | NCHAR [00] | | 30 ns |
| 2.280 µs | 10 ns | NCHAR [FE] | | 30 ns | | | |
| 2.290 µs | 10 ns | | | | NCHAR [04] | | 20 ns |
| 2.310 µs | 20 ns | NCHAR [00] | | 30 ns | | | |
| 2.320 µs | 10 ns | | | | NCHAR [D3] | | 30 ns |
| 2.330 µs | 10 ns | | | | EOP | | 10 ns |
| 2.340 µs | 10 ns | NCHAR [04] | | 30 ns | | | |

Character Display Packet Display Bit-Stream Display

Complete End A: 350.003 MHz End B: 350.003 MHz

STAR-Dundee SpaceWire EGSE Example 2

SpaceWire Link Analyser Mk2 (Serial Number: 35100030) - [~ ~ Not Saved ~ ~]

File View Find Settings Trigger Help

| Time From Trigger | Time Delta | End A Event | End A Error | End A Delta | End B Event | End B Error | End B Delta |
|-------------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1.530 µs | 20 ns | | | | | | |
| 1.540 µs | 10 ns | NCHAR [00] | | 30 ns | NCHAR [00] | | 20 ns |
| 1.560 µs | 20 ns | | | | | | 30 ns |
| 1.570 µs | 10 ns | NCHAR [00] | | 20 ns | NCHAR [02] | | 30 ns |
| 1.590 µs | 20 ns | | | | | | 30 ns |
| 1.600 µs | 10 ns | | | | | | 30 ns |
| 1.620 µs | 20 ns | | | | | | 30 ns |
| 1.630 µs | 10 ns | | | | | | 30 ns |
| 1.640 µs | 10 ns | | | | | | 30 ns |
| 1.650 µs | 10 ns | | | | | | 30 ns |
| 1.670 µs | 20 ns | | | | | | 20 ns |
| 1.680 µs | 10 ns | | | | | | 30 ns |
| 1.690 µs | 10 ns | | | | | | 30 ns |
| 1.710 µs | 20 ns | | | | | | 30 ns |
| 1.730 µs | 20 ns | | | | | | 20 ns |
| 1.750 µs | 20 ns | | | | | | 20 ns |
| 1.760 µs | 10 ns | | | | | | 30 ns |
| 1.770 µs | 10 ns | | | | | | 30 ns |
| 1.790 µs | 20 ns | | | | | | 30 ns |
| 1.800 µs | 10 ns | | | | | | 30 ns |
| 1.820 µs | 20 ns | | | | | | 30 ns |
| 1.830 µs | 10 ns | | | | | | 10 ns |
| 1.840 µs | 10 ns | | | | | | 30 ns |
| 1.850 µs | 10 ns | | | | | | 30 ns |
| 1.870 µs | 20 ns | | | | | | 30 ns |
| 1.880 µs | 10 ns | | | | | | 30 ns |
| 1.900 µs | 20 ns | | | | | | 30 ns |
| 1.910 µs | 10 ns | | | | | | 30 ns |
| 1.930 µs | 20 ns | | | | | | 30 ns |
| 1.950 µs | 20 ns | | | | | | 30 ns |
| 1.960 µs | 10 ns | | | | | | 30 ns |
| 1.970 µs | 10 ns | | | | | | 30 ns |
| 2.000 µs | 30 ns | | | | | | 30 ns |
| 2.030 µs | 30 ns | | | | | | 30 ns |
| 2.050 µs | 20 ns | | | | | | 20 ns |
| 2.060 µs | 10 ns | | | | | | 30 ns |
| 2.080 µs | 20 ns | | | | | | 30 ns |
| 2.090 µs | 10 ns | | | | | | 30 ns |
| 2.110 µs | 20 ns | | | | | | 30 ns |
| 2.120 µs | 10 ns | | | | | | 30 ns |
| 2.130 µs | 10 ns | | | | | | 30 ns |
| 2.140 µs | 10 ns | | | | | | 30 ns |
| 2.160 µs | 20 ns | | | | | | 30 ns |
| 2.170 µs | 10 ns | | | | | | 30 ns |
| 2.200 µs | 30 ns | | | | | | 30 ns |
| 2.210 µs | 10 ns | | | | | | 30 ns |
| 2.230 µs | 20 ns | | | | | | 30 ns |
| 2.240 µs | 10 ns | | | | | | 30 ns |
| 2.250 µs | 10 ns | | | | | | 30 ns |
| 2.270 µs | 20 ns | | | | | | 30 ns |
| 2.280 µs | 10 ns | NCHAR [FF] | | 30 ns | | | 20 ns |
| 2.290 µs | 10 ns | | | | NCHAR [04] | | 20 ns |
| 2.310 µs | 20 ns | NCHAR [00] | | 30 ns | | | 30 ns |
| 2.320 µs | 10 ns | | | | NCHAR [D3] | | 30 ns |
| 2.330 µs | 10 ns | | | | EOP | | 10 ns |
| 2.340 µs | 10 ns | NCHAR [04] | | 30 ns | | | |

Status Counters

Characters/Events Per Second

| | End A | End B |
|---------------------|-------------|-------------|
| Signaling Rate | 350.003 MHz | 350.003 MHz |
| Disconnect Error | 0 | 0 |
| Parity Error | 0 | 0 |
| Credit Error | 0 | 0 |
| Escape Error | 0 | 0 |
| Char Sequence Error | 0 | 0 |
| Data Character | 32,464,091 | 32,464,092 |
| EOP Character | 2,029,006 | 2,029,005 |
| EEP Character | 0 | 0 |
| FCT Character | 4,311,637 | 4,311,637 |
| NULL Character | 0 | 0 |
| Time-code Character | 0 | 0 |

End A

100 %

0 %

End B

100 %

0 %

Draw Errors On Graph

Clear Pause

Character Display Packet Display Bit-Stream Display

Complete End A: 350.003 MHz End B: 350.003 MHz



- Detailed packet definitions
 - Via raw data, variables, automatic CRC and checksum calculation
- Precise packet generation scheduling at specific data rates.
- Packet generation control
 - Via state machines and events



SpaceWire EGSE Key Benefits

- Mimic real-time behaviour of SpaceWire units
- Integrate with equipment via external triggers
- Minimal development time



- Hardware
- Software
- Scripting Language
- Capabilities and Benefits
- Release date: Q1 2012