

### SpaceWire EGSE

The SpaceWire Electronic Ground Support Equipment (EGSE) is a test and development unit that simulates instruments or other SpaceWire equipment in real-time. The EGSE is configured using a simple yet powerful scripting language designed specifically for SpaceWire applications. Once configured the EGSE operates independent of software resulting in real-time performance. This can be used to rapidly mimic the behaviour of SpaceWire equipment, vastly reducing traditional development time, risk and cost associated with writing equivalent software in a real-time operating system.

The SpaceWire EGSE can generate detailed packets in pre-defined sequences at specific times and data rates, controlled by state machines and events. It is an extremely useful tool during equipment development and validation, and spacecraft assembly, integration and test. Using the SpaceWire EGSE and its associated scripting language it is possible to produce a complete SpaceWire instrument simulation in little more than a day.

#### Key Features

- Emulates SpaceWire traffic from almost any instrument, or other flight equipment in real-time.
- Generates sophisticated sequences of packets with varying payload content, timing and data-rates.
- Stores pre-defined packet cargoes in memory to emulate large payloads from high data-rate instruments.
- Monitors received packet streams, reacting to pre-programmed sequences.
- Interacts with external hardware through external triggers, which can be used to trigger different packet sequences.
- Interacts with host software via software event and the provided API to trigger packet sequences for real-time transmission.
- Notifies host software of state changes and events.
- Link data-signalling rates can be changed on the fly, with high resolution, enabling close simulation of an instrument or other SpaceWire unit. Supports time-code generation and can act as a time-code master.
- Highly configurable using the powerful and intuitive scripting language.



SpaceWire EGSE Front View

#### Features

**Two SpaceWire interfaces:** Each fully compatible with the SpaceWire standard and able to operate at up to 200 Mbits/s. Tri-colour LEDs for each link indicate status and activity.

**External triggers:** To integrate with external equipment the EGSE has three external input triggers and one external output trigger. The external input triggers can be used to change the behaviour of the EGSE in response to an input signal. The external output trigger can generate a signal in response to an event, such as the receipt of a specific packet. This can be used to trigger oscilloscopes, logic analysers, or other test equipment to make measurements concurrent with sequences generated by the EGSE.

**Large memory:** 128MBytes for storing pre-defined packet.

**USB 2.0 interface:** Provides a high speed connection between the host PC or laptop to the EGSE.

**19" rack mountable:** Readily mounted in a 1U high, 19" rack alongside other STAR-Dundee products using STAR-Dundee's rack mounting kit. Up to four EGSEs can be mounted on a 1U shelf.

#### SpaceWire EGSE Operation

The SpaceWire EGSE comprises a SpaceWire interface unit and software running on a host PC. The interface unit connects to the host PC using a USB 2.0 cable and is powered via a 5V power brick.

The interface unit is configured using a user authored script that is compiled and loaded onto the SpaceWire EGSE unit. Once configured, it can generate complex SpaceWire packet sequences without further interaction from host PC software.

```
# Set the line rates to 200Mbits/s
config
    spw_tx_rate(1, 200Mbps)
    spw_tx_rate(2, 200Mbps)
end config

# Packet defined with 8 hex
# bytes followed by EOP
packet pkt1
    hex(0A FF 34 C8 11 4D 54 AB)
    eop
end packet

# Send "pkt1" 0.5s after schedule
# starts at 100Mbps
schedule schedule1 @ 100Mbps
    500ms send pkt1
end schedule

# SpW link 1 state machine
statemachine 1
    # State in which "schedule1"
    # is executed repeatedly
    state state1
        do schedule1 repeatedly
    end state
end statemachine
```

Simple SpaceWire EGSE Script

The SpaceWire EGSE's ability to generate packets independent of host software means it can operate in real-time, closely mimicking the packet generation behaviour of an instrument or other SpaceWire equipment.

Time From Trigger	Time Delta	End A Event	End A Error	End A Delta	End B Event	End B Error	End B Delta
2,030 µs	50 ns	WCSAA (E1F)		50 ns	WCSAA (151)		50 ns
2,080 µs	50 ns	WCSAA (1E1)		50 ns	WCSAA (161)		50 ns
2,100 µs	20 ns	EOP		20 ns			50 ns
2,130 µs	30 ns				WCSAA (171)		50 ns
2,150 µs	20 ns	WCSAA (181)		50 ns			50 ns
2,170 µs	20 ns	EOP		20 ns			50 ns
2,180 µs	10 ns				WCSAA (1E1)		50 ns
2,200 µs	20 ns				EOP		20 ns
2,220 µs	20 ns	WCSAA (191)		50 ns			50 ns
2,250 µs	30 ns				WCSAA (1D1)		50 ns
2,270 µs	20 ns	WCSAA (1F2)		50 ns			50 ns
2,300 µs	30 ns				WCSAA (18A)		50 ns
2,320 µs	20 ns	WCSAA (E5)		50 ns			50 ns
2,350 µs	30 ns				WCSAA (17E)		50 ns
2,370 µs	20 ns	WCSAA (1C1)		50 ns			50 ns
2,400 µs	30 ns				WCSAA (E0)		50 ns
2,420 µs	20 ns	WCSAA (1A4)		50 ns			50 ns
2,450 µs	30 ns				WCSAA (1A)		50 ns
2,470 µs	20 ns	WCSAA (151)		50 ns			50 ns
2,500 µs	30 ns				WCSAA (161)		50 ns
2,520 µs	20 ns	WCSAA (131)		50 ns			50 ns
2,550 µs	30 ns				WCSAA (163)		50 ns
2,570 µs	20 ns	WCSAA (1A7)		50 ns			50 ns
2,590 µs	20 ns				WCSAA (20)		50 ns
2,600 µs	10 ns	EOP		20 ns			50 ns
2,620 µs	20 ns				EOP		20 ns
2,640 µs	20 ns	WCSAA (1E)		50 ns			50 ns
2,670 µs	30 ns				WCSAA (11)		50 ns
2,690 µs	20 ns	WCSAA (10)		50 ns	EOP		20 ns
2,740 µs	50 ns	WCSAA (19)		50 ns	WCSAA (112)		50 ns
2,760 µs	50 ns	WCSAA (17)		50 ns			50 ns
2,780 µs	50 ns	WCSAA (17)		50 ns	WCSAA (1A)		50 ns
2,840 µs	50 ns	WCSAA (EC)		50 ns	WCSAA (9C)		50 ns
2,890 µs	50 ns	WCSAA (E)		50 ns	WCSAA (13)		50 ns

Trace of Generated Packet Sequence

The screenshot above is taken from a SpaceWire Link Analyser Mk2 and demonstrates the high data rates that can be achieved. It shows the EGSE generating a sequence of small packets on both SpaceWire links, running at 200MHz with no NULL characters in the trace.

### SpaceWire EGSE Scripting

The SpaceWire EGSE is configured using a simple yet powerful scripting language that was designed specifically for SpaceWire applications:

**Packet definitions:** Can consist of data defined in hexadecimal or decimal bytes, data imported from file, variable references, CRC and checksum calculations, EEP and EOP markers and time-code manager instructions.

**Variables:** Used to define packets with dynamic data. Declared variables can be referenced in packet definitions. The value produced by a variable reference is dependent on its type: constant, random, increment, decrement, rotate right and rotate left. CRC and checksum variables are used to perform CRC and checksum calculations.

**Schedules:** Define the sequence, timing and data rate at which pre-defined packets are transmitted.

**State machines:** Responsible for control of the EGSE state. Each state in a state machine is associated with a schedule which is executed when that state is entered. State transition statements specify the event(s) on which to transition from one state to another.

**Events:** Used to control the current state of the SpaceWire EGSE state machines and therefore the current packet generation schedule. The different event types are:

- Software: Transition from one state to another in response to events generated from host software.
- State machine: Raise an event when a state of interest is entered.
- External trigger in: React to an external input trigger signal received from other equipment.
- External trigger out: Generate an external trigger output signal in response to an event of interest.
- Time-code received: React to the receipt of a time-code on a SpaceWire interface.

- Time-code transmitted: React to the transmission of a time-code from a SpaceWire interface.
- Received pattern matched: Transition from one state to another when the SpaceWire traffic received on an interface matches a specified pattern.

### Software

The EGSE is supplied with software comprising:

**Compiler and loader:** Compile scripts and configure the SpaceWire EGSE using these command line applications.

**C API:** Write custom software to interact with the EGSE.

**GUI:** Combines a text editor, for creating and modifying EGSE scripts, with much of the functionality provided by the compiler, loader and C API including: script compilation, EGSE configuration, software event generation, state and event notification monitoring and periodic time-code generation.

### Upgrades and Support

**Field upgradable:** Upgrades and requested customisations can be downloaded from the STAR-Dundee website and installed quickly and efficiently.

**First class support:** As with all of STAR-Dundee's products, a year's support and maintenance is included with the SpaceWire EGSE. Support is provided directly from the team that developed the product allowing us to respond quickly with answers to customer questions, give assistance with application development, and resolve any problems quickly.

### Specifications

Size: 110 x 30 x 112 mm (excluding feet).

Power: +5V DC, power brick supplied.

SpaceWire Ports:

- 2 SpaceWire ports, micro-D connectors located on the front panel.
- ECSS-E50-12A and ECSS-E-ST-50-12C compliant.
- Maximum Speed: 200 Mbits/s each port.

USB 2.0:

- High Speed 480 Mbits/s.

Trigger input and output:

- 3 dedicated trigger inputs; 1 on front panel, 2 on the rear panel
- 1 dedicated trigger output on front panel
- SMB connectors with +3.3V signal level and 5V tolerances.

Mictor logic analyser connectors:

- 2 on the rear panel show device state

API: C language.

Supporting software included:

- Operates under Windows (Windows 10, 8, 7, Vista, XP) and Linux (2.6 and 3 kernels).

### Application Notes

Application notes are available that describe how the SpaceWire EGSE can be used to emulate SpaceWire devices and integrate with spacecraft EGSE equipment; see [www.star-dundee.com](http://www.star-dundee.com).