

# **DYNAMIC ENGINEERING**

150 DuBois St. Suite C Santa Cruz CA 95060

831-457-8891 Fax 831-457-4793

<http://www.dyneng.com>

[sales@dyneng.com](mailto:sales@dyneng.com)

Est. 1988

## **Software User's Guide (VxWorks-6.9)**

# **SpaceWire**

Four-Channel SpaceWire Interface



## SpaceWire

Dynamic Engineering  
150 DuBois St Suite C  
Santa Cruz, CA 95060  
831-457-8891

©2019-2020 by Dynamic Engineering.  
Other trademarks and registered trademarks are owned by their  
respective manufactures.  
Revised 02/19/2020

This document contains information of proprietary interest to Dynamic Engineering. It has been supplied in confidence and the recipient, by accepting this material, agrees that the subject matter will not be copied or reproduced, in whole or in part, nor its contents revealed in any manner or to any person except to meet the purpose for which it was delivered.

Dynamic Engineering has made every effort to ensure that this manual is accurate and complete. Still, the company reserves the right to make improvements or changes in the product described in this document at any time and without notice. Furthermore, Dynamic Engineering assumes no liability arising out of the application or use of the device described herein.

The electronic equipment described herein generates, uses, and can radiate radio frequency energy. Operation of this equipment in a residential area is likely to cause radio interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Dynamic Engineering's products are not authorized for use as critical components in life support devices or systems without the express written approval of the president of Dynamic Engineering.

Connection of incompatible hardware is likely to cause serious damage.



---

---

# Table of Contents

---

---

<b>PRODUCT DESCRIPTION</b>	<b>5</b>
Software Description	5
<b>INSTALLATION</b>	<b>6</b>
<b>VXBUS DRIVER METHODS</b>	<b>7</b>
<b>SAMPLE APPLICATION</b>	<b>7</b>
Invocation parameters	7
Warranty and Repair	8
Service Policy	8
Out of Warranty Repairs	8
For Service Contact:	8

## Product Description

The SpaceWire I/O card is available in multiple formats. All variants support four channels or ports of the SpaceWire protocol. Various configurations are available including addition of external FIFOs to support further burst capability.

This driver supports SpaceWire Xilinx Design Rev 10 or greater and all BK revisions.

For a detailed description of the hardware including register definitions, see HW User Manual, SpaceWire.

## Software Description

The SpaceWire driver supports simultaneous operation of all ports independently. The driver auto-detects the presence of external FIFOs on a per port basis and controls the HW accordingly.

Some BSPs map host memory to a non-zero base for PCI externally visible memory. If applicable, define `DE_EXT_PCI_MEM_BASE` found in `de_SpwrDrv.c` to the appropriate base address for your BSP. A symptom of this issue is that DMA transfers do not complete.

The driver has been validated on VxWorks revision 6.9 on a P2020 platform (multi-core PPC) which is big endian.

## Application Programming model

The following is the applicable section from the SpaceWire specification ECSS-E-ST-50-12C:

### 4.8 Application programming interface

The application programming interface (API) is not defined in this Standard. However, a typical application interface comprises the following services:

- **Open link:** Starts a link interface and attempts to establish a connection with the link interface at the other end of the link.
- **Close link:** Stops a link and breaks the connection.
- **Write packet:** Sends a packet out of the link interface.
- **Read packet:** Reads a packet from the link interface.
- **Status and configuration:** Reads the current status of the link interface and sets the link configuration.

The Dynamic Engineering driver implements this functionality as follows:



- 1) During system initialization, a default configuration is applied. The port is configured in internal loopback, thus external link access is disabled. Link will remain disconnected.
- 2) Port must be configured via DE\_CONFIG\_PT ioctl to enable link and read/write access to remote end of the link. If the link connection is not established in 4 seconds, the ioctl will return failure status code.
- 3) Link status may be interrogated via DE\_GET\_STATS ioctl. Current link state as well as accumulated error counts and I/O byte counts are returned.
- 4) Read/write implemented via deReadPacket/deSendPacket APIs.
- 5) Links are closed upon system shutdown.

## Configuration Considerations

Two types of link connection/establishment are supported, manual and auto-start. The value of this parameter is determined by the requirements/implementation of the external device. Nominally, this parameter should be set to 0 (manual start). If problems establishing the link are encountered, set this parameter to 1 (auto-start). This allows the other end of the link to initiate the link start.

Besides link connection parameters, the following are set via the config ioctl: Time code generation, DMA priority, packet mode (enable/disable), and link speed in MHz.

## Installation

Copy the tar ball containing this driver to the `$(WIND_BASE)/target/3rdparty/dyneng` directory of your project. If a dyneng directory does not exist, create one. After extraction you should find the following files in the dyneng tree:

```

Makefile README
de_SpaceWire
  40de_Spwr.cdf deSendPacket.c deSendPacket.mk deloctl.c
  deloctl.mk deRcvPacket.c deRcvPacket.mk de_SpwrDrv.c,
  de_SpwrDrv.dc de_SpwrDrv.dr de_SpwrDrv.h de_Common.h,
  dePIIDefs.h Makefile README release_notes.txt
de_SpaceWire/apps
  de_loApp.c, de_loctlApp.c, Makefile

```

The README file contained in de\_SpaceWire directory contains the configuration and build steps required to include this driver in your VxWorks



image. This information is specifically not included in this document to avoid conflicts that may occur due to updates in the source tree.

## VxBus driver methods

The SpaceWire driver supports custom ioctl, and port methods (deIoctl, deSendPacket deRcvPacket) implemented via VxBus.

The application/middleware must configure a port prior to invoking the port methods. This is the only ioctl required operationally. The remaining ioctls are either informational such as fetching I/O card specific parameters, obtaining current interrupt count, or provided to enable debug or user specific customizations. Please see `dyneng/de_SpaceWire/de_SpwrDrv.h` for a detailed description of the supported driver methods.

## Sample application

Two sample applications (`de_ioApp.c`, `de_ioctlApp.c`) are provided to demonstrate configuration, ioctl invocation, and I/O in the supported modes.

## Invocation parameters

`de_io` (VXB\_DEVICE\_ID pDev, lpbck (0=ext,1=wrapback), port(0-3), frame\_len(bytes), reader(0=writer,1=reader), iter)

**pDev** can be obtained by issuing `vxBusShow`, find the entry corresponding to the device to be utilized by this application, `pDev = 0x0034c308` in the following example:

```
PCI_BUS @ 0x003467C8 with bridge @ 0x0034c308
```

```
Device Instances:
```

```
de_SpwrDrv unit 0 on PCI_BUS @ 0x0034cb08 with busInfo 0x00000000
```

Second parameter I/O mode (external or internal loopback) followed by port to be exercised, frame/packet length, reader/writer, and number of iterations to execute.

The application can be invoked from the serial console, or a telnet shell. For internal loopback the application will transmit and receive on the same port. Thus, only one instance of the app is required to exercise both Tx and Rx.

In external I/O mode 2 instances of the app must be invoked to exercise both the read and write ports. It is assumed the two ports are connected directly or via a switch.

## Warranty and Repair

Please refer to the warranty page on our website for the current warranty offered and options.

<http://www.dyneng.com/warranty.html>

## Service Policy

Before returning a product for repair, verify as well as possible that the suspected unit is at fault. Then call the Customer Service Department for a RETURN MATERIAL AUTHORIZATION (RMA) number. Carefully package the unit, in the original shipping carton if this is available, and ship prepaid and insured with the RMA number clearly written on the outside of the package. Include a return address and the telephone number of a technical contact. For out-of-warranty repairs, a purchase order for repair charges must accompany the return. Dynamic Engineering will not be responsible for damages due to improper packaging of returned items. For service on Dynamic Engineering Products not purchased directly from Dynamic Engineering contact your reseller. Products returned to Dynamic Engineering for repair by other than the original customer will be treated as out-of-warranty.

### Out of Warranty Repairs

Out of warranty repairs will be billed on a material and labor basis. The Customer approval will be obtained before repairing any item if the repair charges will exceed one half of the quantity one list price for that unit. Return transportation and insurance will be billed as part of the repair and is in addition to the minimum charge.

### For Service Contact:

Customer Service Department  
Dynamic Engineering  
150 DuBois St. Suite C  
Santa Cruz, CA 95060  
831-457-8891  
831-457-4793 fax  
InterNet Address [support@dyneng.com](mailto:support@dyneng.com)

