

# **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-7)**

# **SpaceWire**

Four-Channel SpaceWire Interface

K & BK models

## 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 manufacturers.  
Revised 02/26/20

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>4</b>
Software Description	4
<b>INSTALLATION</b>	<b>5</b>
<b>APPLICATION INVOCATION</b>	<b>6</b>
Warranty and Repair	7
Service Policy	7
Out of Warranty Repairs	7
For Service Contact:	7

## 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.

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.

The driver has been validated on VxWorks-7 SR0540 SMP on the 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) Upon first open, 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.



- 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 standard read/write APIs.
- 5) Upon last close of a port, the port is reset and link is disconnected.

Besides blocking read timeouts and access modes to the FIFOs, the following parameters are set via the config ioctl: time code generation, DMA priority, read and write buffer lengths, packet mode (enable/disable), auto start enable. and link speed in MHz.

## 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.

Please see de\_SpwrDrv.h for details of the parameters for this ioctl.

## Installation

Create a directory in your WindRiver directory, WindRiver/vxworks-7/pkg/os/drv/vxbus-x.x.x.x/drv-x.x.x.x/src/dyneng. Extract the contents of the tar-ball to this directory.

### Driver configuration and build

- 1) Copy 40deSpwr.cdf to WindRiver/vxworks-7/pkg/os/drv/vxbus-x.x.x.x/drv-x.x.x.x/cdf directory.
- 2) Edit makefile in WindRiver/vxworks-7/pkg/os/drv/vxbus-x.x.x.x/drv-x.x.x.x/src. Add dyneng to the SUBDIRS line, it should look something like this:  
SUBDIRS = intCtrl i2c spi gpio timer dma busCtrl sio resource sensor dyneng
- 3) Edit your VIP kernel configuration, include deSpwrDrv driver for the SpaceWire IO card (DRV\_DE\_SPWR)
- 4) Build your VSB, then your VIP.
- 5) Download image to target.

### Application builds



Two applications are included, an I/O application demonstrating/testing proper I/O operation, and an ioctl application which can invoke PLL programming, and direct register interface.

### **spwrApp (I/O application)**

- 1) Create a DKM (Downloadable Kernel Module) project named spwrApp based upon your VSB.
- 2) Add include path which points to driver source directory
- 3) Copy dkm.c to your spwrApp directory in your workspace from spwrApp directory which can be found in driver directory apps/spwrApp
- 4) Build project
- 5) Download spwrApp.out to your target via ld command  
ld < /your\_path/spwrApp.out

### **spwrioctl (ioctl application)**

- 1) Create a DKM (Downloadable Kernel Module) project named spwrioctl based upon your VSB.
- 2) Add include path which points to driver source directory
- 3) Copy dkm.c to your spwrioctl directory in your workspace from spwrioctl directory which can be found in driver directory apps/spwrioctl
- 4) Build project
- 5) Download spwrioctl.out to your target via ld command  
ld < /your\_path/spwrioctl.out

## **Application invocation**

### **spwrApp:**

de\_io (lpbck (0=ext,1=wrapback),port(0-3),frame\_len(bytes)  
,reader(0=writer,1=reader), iter)

If executing app with lpbck=1, only 1 instance needs to be invoked and frame\_len must be < 4k. reader is don't care.

When testing with external cabling connecting 2 ports, 2 instances must be invoked from separate shells (telnet). 1 instance is specified as the reader, the other as a writer. Frame\_len must be same for both instances as the as the number of iterations to execute as follows:

```
de_io (0, 0, 8192, 1, 100000) // Reader, 8K packet, Reader should be
                               invoked first
de_io (0, 1, 8192, 0, 100000) // Writer, must be invoked within a few seconds
                               after reader
```



This assumes port 0 is cabled to port 1. Packet size and data integrity validated by the reader.

### **spwrloctl:**

```
de_ioctl // No parameters
```

A menu will be displayed enabling user to invoke PLL programming or direct register access demonstrations

## **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**

Software support contracts are available to update add features, change for different versions of OS etc. Please contact Dynamic Engineering for these options.

### **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)

