

# **DYNAMIC ENGINEERING**

150 DuBois St. Suite B/C Santa Cruz CA 95060

831-457-8891

<https://www.dyneng.com>

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

Est. 1988

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

# **ccPMC-BiSerial-VI-ORN1 SDLC & NRZL**

**2 Ports SDLC**

**2 Ports NRZL**

## SpaceWire

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

©2022 by Dynamic Engineering.  
Other trademarks and registered trademarks are owned by their  
respective manufactures.  
Revised 08/18/2022

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>4</b>
<b>VXBUS DRIVER METHODS</b>	<b>5</b>
<b>SAMPLE APPLICATIONS</b>	<b>5</b>
Invocation parameters	5
Warranty and Repair	7
Service Policy	7
For Service Contact:	7

## Product Description

The ccPMC-BiSerial-VI-ORN1 implementation supports 2 ports of SCLC function and 2 ports of NRZL. The card also supports parallel port operation as well as (LM75) temperature sensor.

For a detailed description of the hardware including register definitions, see HW User Manual, ccPMC-BiSerial-VI-ORN1\_HwMan.

## Software Description

The Orn1 driver supports simultaneous full-duplex operation of the four protocol ports (SDLC & NRZL). The frame length for SDLC read/write functions is specified in bytes and must be a multiple of 16 bits. The packet length for NRZL can be specified in bytes or bits. This is configurable via the configuration IOCTL. All ports must be configured prior to issuing any reads or writes.

Although HW supports configuration of all data I/O lines as either register or protocol controlled, software reserves the first 18 bits for SDLC & NRZL. I/O bits 31-18 may be configured for parallel port operation

An IOCTL is provided to read the onboard LM75 temp sensor. The value returned is of type float, units are Celsius. Other IOCTLs are provided to return board parameters, configure ports, and return port statistics. Port transmit/receive counts are always in units of bytes. A PLL ioctl is provided to allow programming or reading current PLL settings. By default, the PLL is programmed upon initial port configuration

The driver has been validated on VxWorks revision 6.9 on a NXP P2020 RDB evaluation board.

## 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/de\_BiSerOrn1 tree:

```
Makefile, README, release_notes.txt
40de_Orn1.cdf de_BirSerOrn1.dc, de_BiSerOrn1.dr, de_BiSerOrn1.c
de_BiSerOrn1.h, de_Common.h, dePllDefs.h, deWrite.c, deWrite.mk,
deRead.c, deWrite.mk, deloctl.c, deloctl.mk
de_BiSerOrn1/apps
    Makefile, de_loApp.c, de_loctlApp.c
```



The README file contained in de\_BiSerOrn1 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, deWrite, deRead) implemented via VxBus.

The application/middleware must configure a port prior to invoking the port methods. This is the only IOCTL required operationally for the protocol ports. The remaining IOCTLs are either informational such as fetching I/O card specific parameters, obtaining current port statistics, or provided to enable debug or user specific customizations. Please see dyneng/de\_BiSerOrn1/de\_BiSerOrn1.h for a detailed description of the supported driver methods.

## Sample applications

The sample application, de\_IoApp.c (dyn\_io) is provided to demonstrate configuration, and frame/packet transfer operations as well as validating proper I/O operation. de\_IoctlApp.c (dyn\_ioctl) demonstrates the usage of other IOCTLs.

## Invocation parameters

dyn\_io (VXB\_DEVICE\_ID pDev, port(0-3), reader(0=writer,1=reader, frame\_len (bytes/bits), iterations)

**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 @ 0x0034aa24 with bridge @ 0x0034aa94
```

Device Instances:

```
de_BiSerOrn1 unit 0 on PCI_BUS @ 0x0039b018 with busInfo 0x0000000
```

Ports 0 and 1 are SDLC ports, ports 2 and 3 are NRZL. To test a port, a Dynamic Engineering Loopback fixture must be attached. Ports are externally looped back to themselves. Two instances of the application must be invoked to test a port, a reader and a writer. All invocation parameters must be identical



except for the reader parameter. The reader must be invoked first, then the writer in a separate terminal window:

Example testing port 0, frame length = 2048, iterations = 10000

1<sup>st</sup> Terminal (reader):

```
dyn_io 0x0039b018, 0,1,2048,10000
```

2<sup>nd</sup> Terminal (writer):

```
dyn_io 0x0039b018,0,0,2048,10000
```

Application executes until iteration count is reached, terminated by hitting any key (kbhit), or an error is encountered during I/O execution.

The reader validates data received from the writer.

Other IOCTLs maybe invoked/demonstrated by executing de\_ioctlApp.c.

```
dyn_ioctl 0x0039b018
```

Upon invocation, a menu will be displayed of options that can be invoked.

## Warranty and Repair

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

<https://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.

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 B/C

Santa Cruz, CA 95060

831-457-8891

InterNet Address [support@dyneng.com](mailto:support@dyneng.com)

