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
Est. 1988

Software User's Guide
(Linux)

PMC-BiSerial-III SDLC
8-Channel SDLC Interface

PMC-BiSerial-III SDLC

Dynamic Engineering
150 DuBois St Suite C
Santa Cruz, CA 95060
831-457-8891
831-457-4793 FAX

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.

Product Description

The PMC-BiSerial-III-SDLC board is an eight channel full-duplex SDLC interface card. Further any unused I/O bits can be utilized for general purpose parallel port I/O. 32 bits of I/O are consumed if all 8 SDLC are opened (4 per port).

For a detailed description of the hardware including register definitions, see HW User Manual, PMC-BiSerrial-III-SDLC.

Software Description

The driver supports full-duplex operation of the SDLC ports. Both blocking and non-blocking reads are supported. Further, a timeout may be specified for blocking reads.

A default configuration is applied when a SDLC port is opened. Currently, the default setting is as follows: Idles disabled, repeated flags disabled, external Tx clock selected. The default configuration may be changed by altering the value of default_sdlc_config which is declared in the driver header file, de_BiSerSdlc.h. Further, the blocking read timeout is disabled.

New configuration parameters may be applied after a port is opened via the DE_CONFIG_PT ioctl. Please see the driver header file, de_BiSerSdlc.h for other defined ioctls which includes inline documentation..

The version of this driver is v1.0.2. The driver has been validated on an i7 Ubuntu server running 3.8.0-44 kernel (64 bit) SMP (little Endian platform) as well as a P2020 (PPC) running 3.0.48-r5 kernel (32 bit) SMP (big Endian platform).

Installation

1)   Copy de_BiSerSdlc.c and de_BiSerSdlc.h to your module build directory. Invoke the system %make.+ A makefile for this module has been included in the release tar-ball.
2)   Copy the resulting de_BiSerSdlc.ko module to the target platform/directory.
3)   Copy the startup script bnm to the target.
4)   Invoke the script (./bnm), it will create the devices required by the driver and performs an insmod of the module. You may invoke this script from the systems rc.local file as well

Application Programming model

After a port is opened, it may be configured for the desired mode of operation via the DE_CONFIG_PT ioctl. Both blocking and non-blocking modes of operation are supported. This behavior is set via the standard file flags upon open.

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.

1)   Compile the sample application for your platform, the output executable for these examples are dyn_io and dyn_ioctl.
a.   Nominal compilation gcc
        gcc . Wall . o dyn_io de_IoApp.c
        gcc . Wall . o dyn_ioctl de_IoctlApp.c
        The apps should compile without warnings, it is assumed de_PcieBiSerSdlc.h is resident in the same directory as the applications for these examples.

Invocation parameters

I/O application invocation is as follows:

./dyn_io port(0-8) mode(r(eader)|w(riter), frame_len(32 bit words, optional), iterations(optional)

The first parameter specifies the port. The second is mode, either reader or writer. The third optional parameter is frame length is specified in 32 bit words. The final optional parameter is number of iterations to execute.

Port 8 is the parallel port, only port number is required when invoking this test.

The app assumes a Dynamic Engineering loop-back fixture is attached in order to execute successfully. Rx data is compared to Tx data upon read completion for each iteration. Two instances of the app must be invoked for each SDLC port under test, parallel port requires only 1. For SDLC, the reader must be invoked first, followed by the writer within 5 seconds as follows:

./dyn_io 0 r 64 100
./dyn_io 0 w 64 100

Ioctl application invocation is as follows:
./dyn_ioctl

A menu will be displayed:
Enter p(ll program)||r(eg ops)|e(xit)

The Ioctl application demonstrates pll programming, register R/W/RMW operations.

Support Contract

Dynamic Drivers are provided AS-IS and sometimes our clients need a little help. Please refer to the support contract page on our website for options about getting help with your driver use and SW development.

http://www.dyneng.com/TechnicalSupportFromDE.pdf

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 revisions 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