Release Note

pSOS+™ Driver Installation for NetBlaster PCI and CompactPCI® Fast Ethernet Adapters

Installing pSOS+ 2.2.6 (Intel Platforms) Driver

These instructions are relevant to the pSOS+ driver (version 1.41) for ZNYX NetBlaster PCI and CompactPCI adapters, models ZX345, ZX345Q, ZX346, ZX346Q, ZX348, ZX348Q, ZX412 and ZX414. The pSOS+ driver for the NetBlaster adapters can be obtained from either the ZNYX web site at www.znyx.com. The distribution is a self-extracting archive containing all the files needed to complete installation.

RETRIEVING DRIVERS

1. Visit the Driver Download area of the ZNYX web site to obtain the appropriate pSOS+ driver for your NetBlaster adapter: http://www.znyx.com
2. The driver is a self-extracting archive. Save the file and make note of its location.
3. Execute the file to extract the driver library and configuration files. The README.TXT file contains the instructions presented below.

INSTALLING DRIVERS

1. Copy the driver library file to a suitable location within the Prism+ pSOS build environment.

   Modify the "Makefile" in your application directory to include the driver library file. For example, if you put the file in /znyx:

   Change:

   ```
   DRV_LIB1=
   ```

   to:

   ```
   DRV_LIB1=/znyx/znyx.lib
   ```

2. Modify the file $PSS_BSP/bsp.h to add the following definitions. Add an entry for ZNYX to the list of network interfaces:

   ```
   #define  ZNYX          7             /* include ZNYX support */
   ```

   Set BSP_LAN1 to YES to include network interface support, and define ZNYX as BSP_LAN1_MODEL. All ports take the settings defined for LAN1.

   ```
   #define BSP_LAN1        YES     /* include network interface */
   #define BSP_LAN1_MODEL  ZNYX    /* network interface is ZNYX */
   ```

   Add the following lines for the ZNYX network interface:

```
#if BSP_LAN1_MODEL == ZNYX
#define BSP_LAN1_ENTRY  NiZNYX         /* entry point for driver */
#define BSP_LAN1_MTU           1500    /* MTU size for all ports */
#define BSP_LAN1_HWALEN 6              /* length of hardware address */
#define BSP_LAN1_FLAGS  0x8000+0x2000+0x1+0x800      /* reserved */
#define BSP_LAN1_NMCAST 0              /* no multicast support */
#define BSP_LAN1_PKB           100     /* reserved */
#define BSP_LAN1_BUFFS_MIN     10      /* reserved */
#define BSP_LAN1_BUFFS_MAX     100     /* reserved */
#endif
```

3. Modify your application to include support for the ZNYX network interface. Edit the drv_conf.c file associated with your application, and add the lines necessary to call InstallNi once for each instance of the driver. Set the number of ports to be activated, and whether to use RARP, the media speed, IP address and netmas for each port. The default value is to not use RARP and to autonegotiate media speed. IP addresses and netmasks should be indicated as hexadecimal values. Make the following changes to drv_conf.c:

   After "#if SC_PNA", but before the *init* function, add:

```
#if BSP_LAN1_MODEL==ZNYX
/*
 * This structure cannot be changed because znyx.lib
 * uses this structure to obtain the media type on
 * a per-port basis.
 *
 * Do not attempt to activate more ports than are actually
 * installed in the target machine.
 */

#define ZNYX_MAX_PORTS  16     /* Matches znyx.lib's precompiled value */

/*
 * Options for media speed :
 *
 * Auto Negotiation:           0
 * 10BaseT - Half Duplex:      1
 * 10BaseT - Full Duplex:      4
 * 100BaseTX - Half Duplex:    5
 * 100BaseTX - Full Duplex:    6
 *
 * Other values are reserved.
 */
int i;
struct zxe_ports {
 int use_rarp;  /* 1 if yes, 0 if no */
 int media_speed;      /* media speed, options above */
 int ip_addr;          /* IP address, in hexadecimal */
 int subnet_addr;      /* Subnet address, in hexadecimal */
};

struct zxe_config {
```

```
 int num_ports;        /* Max is ZNYX_MAX_PORTS */
 struct zxe_ports zxe_portdefs[ZNYX_MAX_PORTS];
};

struct zxe_config zxe_conf =
 {
 4,             /* Number of ports to be activated. */
 { {0, 0, 0x0a000001, 0xff000000},    /* first port  */
 {0, 0, 0x0b000001, 0xff000000},      /* second port */
 {0, 0, 0x0c000001, 0xff000000},      /* ... */
 {0, 0, 0x0d000001, 0xff000000},
 {0, 0, 0x0e000001, 0xff000000},
 {0, 0, 0x0f000001, 0xff000000},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0},
 {0, 0, 0, 0} }
};
#endif
```

Alter the original BSP_LAN1 routine with the ZNYX specific routine to bring up multiple ports:

```
#if BSP_LAN1
#if BSP_LAN1_MODEL != ZNYX     /* Configuration for non-ZNYX ports */
    if (SysVars.Lan1)
        {
        /*-------------------------------------------------------------*/
        /* If RARP is specified for the IP address, use it now to      */
        /* obtain the IP address for this interface.                   */
        /*-------------------------------------------------------------*/
        if (SysVars.Lan1IP == USE_RARP)
            {
            #if (SC_DEV_DLPI && SE_SHARE_NI)
                IPaddr = RarpEth((long (*)())DlpiEnet);
            #else
                IPaddr = RarpEth((long (*)())BSP_LAN1_ENTRY);
            #endif
            if (IPaddr == 0)
                SysInitFail(RarpErrTimeout);
            if (IPaddr == -1)
               SysInitFail(RarpErrNet);
            }
        else
            IPaddr = SysVars.Lan1IP;

        #if (SC_DEV_DLPI && SE_SHARE_NI)
```

```
            InstallNi((int (*)())DlpiEnet, IPaddr,
        #else
            InstallNi((int (*)())BSP_LAN1_ENTRY, IPaddr,
        #endif
                    BSP_LAN1_MTU, BSP_LAN1_HWALEN, BSP_LAN1_FLAGS,
                    SysVars.Lan1SubnetMask, 0);
        }
#else
  if (zxe_conf.num_ports > ZNYX_MAX_PORTS)
         zxe_conf.num_ports = ZNYX_MAX_PORTS;

  for (i = 0; i < zxe_conf.num_ports; i++) {
         if (zxe_conf.zxe_portdefs[i].use_rarp) {
                IPaddr = RarpEth((long (*)())NiZNYX);
                if (IPaddr == 0)
                         SysInitFail(RarpErrTimeout);
                if (IPaddr == -1)
                        SysInitFail(RarpErrNet);
         } else {
                IPaddr = zxe_conf.zxe_portdefs[i].ip_addr;
         }

         InstallNi((int (*)())NiZNYX, IPaddr,
                        BSP_LAN1_MTU, BSP_LAN1_HWALEN, BSP_LAN1_FLAGS,
                        zxe_conf.zxe_portdefs[i].subnet_addr, 0);

  }
#endif   /* BSP_LAN1_MODEL==ZNYX */
#endif  /* BSP_LAN1 */
```

4. Make changes necessary to sys_conf.h for your application. In particular, make sure you allocate at least 1024 message blocks per port, and at least 256 0 byte buffers per port and at least 256 2K byte buffers.

5. You are now ready to build your application with the network interfaces included. See the pSOS manuals for more information on how to build applications and kernels.

- There is no multicast support in this release.
- There are no interfaces statistics available in this release

ZNYX Corporation
48501 Warm Springs Blvd., Suite 107
Fremont, CA 94539 USA
(510) 249-0800
Fax (510) 656-2460
www.znyx.com

Document # DC0102-01