Interface Masters
TECHNOLOGIES
Innovative Network Solutions

Niagara 22xxed

PCI-E Dual Port
Software Guide

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF Tables

1. Objective / Overview

1.1 Overview

This document is to serve as a guide and assist integration of the Interface Masters drivers for the Niagara 22xxed.

1.2 Required files

a) Intel igb driver, can be obtain from Intel for free, or given by Interface Masters
b) Interface Masters driver "niagara22xxed_r0.1.tar.gz"

1.3 Installation

Please perform following steps to install Intel and Interface Masters drivers.

1.3.1 Intel Drivers

Due to the fact that Intel's igb drivers are not maintained by Interface Masters, their installation instructions may change without warning. Please follow their instructions provided by their documentation.

1.3.2 Interface Masters Drivers

As of writing this document, the driver version is r0.1.

```
tar xvzf niagara22xxed_r0.1.tar.gz
cd ./niagara22xxed_r0.1
make clean
make     (see note below)
./n22xxed_load
```

Note: To rebuild for Kernel 2.4, comment out line 85 "#define K26 1" in n22xxed.c, then do the following command:

```
make –f Makefile.24
```

2. Theory of operation from software point of View

2.1 Background

Niagara 22xxed is a PCI Express, Dual Gigabit NIC card with several programmable modes of operations.

The mode of operation is selected by Default Mode Registers & Current Mode Registers.
The Default mode registers are non-volatile and therefore holds their values while powered down.
The Current Mode Registers are volatile and loose their content when the power is turned off.

Upon power up the pair of ports is programmed based on their respective Default Mode Register.
In order to alter the operation of the port pair one can select the desired mode by setting the Current Mode Registers with the appropriate value or by changing the settings of the Default Mode Register.

2.1.1 Bypass Mode

In this mode of operation no data traffic will be received by the Niagara 22xxed.
All the data traffic from port A is passed to port B and vise versa.

2.1.2 Active Mode

In this mode of operation, every port can have the data traffic be sent and/or received by the Niagara 22xxed's Ethernet controllers. Each of the ports is connected directly to their respective Ethernet controller.

2.1.3 No Link Mode

In this mode of operation, the link is broken between each port of the port pair and every port's link with their respective Ethernet controller is broken.
This mode is generally selected while the appliance is powering up and the operating system is not filly operational yet.
In this mode the network that is connected to port A is disconnected from the network that is connected to port B, and also both ports are disconnected with their respective Ethernet controller.

2.2 Local CPU

The local CPU is an 8-bit RISC processor with following integrated peripherals.

a) 8k FLASH
b) 512 byte EEPROM
c) General purpose registers
d) Two timers

2.2.1 CPU Tasks

The CPU performs following tasks

a) Control all relays
b) Control Active LED
c) Count down the heartbeat timer
d) Interface with the Intel Dual Gigabit MAC via the Altera CPLD.
e) Configure PCI Express bridge

2.2.2 Local CPU Memory MAP

The EEPROM is organized as 512 by 8 bits. The EEPROM is divided into two sections. The 1st 64 bytes of the EEPROM are accessible by the OEM customers, but the upper 64 bytes are reserved for use by Interface Masters only.

2.2.2.1 EEPROM Memory MAP

Following table list the content of EEPROM.
The user can read or write to the 1st 64 locations. Interface Masters provides software utilities to enable OEM customers to access these locations.

LOCATION (DECIMAL)	NAME	READ / WRITE ACCESS	HOLDS DATA WHILE POWER IS OFF
0	Timeout Register for lower two-port	OEM Customers	Yes
1	Reserved	OEM Customers	Yes
2	Reserved	OEM Customers	Yes
3	Reserved	OEM Customers	Yes
4	Start Up Wait Register	OEM Customers	Yes
5	Start Up Wait Override Register	OEM Customers	Yes
6 - 29	General purpose spaces	OEM Customers	Yes
30 – 31	Test mode registers	OEM Customers	Yes
32 – 34	Reserved	OEM Customers	Yes
35 - 36	Reserved	OEM Customers	Yes
37	Reserved	OEM Customers	No
38 - 42	Reserved	OEM Customers	Yes
43	Reserved	OEM Customers	No
44	Reserved	OEM Customers	Yes
45 - 46	Test mode registers	OEM Customers	Yes
47	Relay Status Register	OEM Customers	Yes
48	Reserved	OEM Customers	Yes
49	Current Mode Register	OEM Customers	No
50	Copper Default Mode Register	OEM Customers	Yes
51	Power Off Mode Register	OEM Customers	Yes
52	Reserved	OEM Customers	Yes
53	Reserved	OEM Customers	Yes
54	Reserved	OEM Customers	Yes
55	Reserved	OEM Customers	No
56	Reserved	OEM Customers	Yes
57 - 64	Reserved	OEM Customers	-
65	OEM ID	Interface Masters	Yes
66	Product ID	Interface Masters	Yes
67	Product Revision Number	Interface Masters	Yes
68	Local CPU Firmware Revision Number	Interface Masters	Yes
69 - 512	Reserved	Interface Masters	-

Table 1, EEPROM Memory MAP

Following paragraphs describes the content of the EEPROM.

2.2.2.1.1 Timeout Register (Reg 0)

These Registers hold the timeout value for the bypass circuits. The timeout period unit is in 100ms. A timeout value of 0x01 has a timeout period of 100ms, while a timeout value of 0xFF has a timeout period of 25.6s.
The factory default value is 0x0F.

2.2.2.1.2 General purpose spaces

These 31 locations are reserved for OEM customers. These spaces could be used to hold security signature or board identification number.

2.2.2.1.3 Test Mode Register

It is set to a value greater than 0 when one of the test mode is entered.
It must be set to 0 for normal operation.

2.2.2.1.4 Relay Status Registers (Reg 47)

These read only registers return the state of the relays.
Values are: 0x01 => bypass; 0x02 => active; 0x03 => no link

2.2.2.1.5 Copper Default Mode Registers (Reg 50)

Control the switching of the relays when copper medium is selected.
The table below shows the various options.

Reg Value	Relay Status	
	Heart Beat Present	**No Heart Beat**
0	Active	Bypass
1	Bypass	Active
2	Active	Active
3	Active	No Link
4	Bypass	Bypass
5	No Link	No Link

Table 2, Copper Default Mode Register

2.2.2.1.6 Current Mode Registers (Reg 49)

Control the current operational mode. Upon power up, the operational mode is selected by the Copper or Fiber Default Mode registers.
The Current Mode registers are used to select a different mode that will not be remembered when the system is powered down.
The table below shows the various options.

Selected Medium	Bit 7	Bits 6:0	Operation
Copper	0 (must be 0)	0x70	Mode selected by the Copper Default Mode Registers
		0 to 5	Same modes as described in the copper default mode registers
		other	Invalid

Table 3, Current Mode Register

2.2.2.1.7 Start Up Wait Register (Reg 4)

This register holds the value of the start up wait in 2 second units. Upon start up, the relays will be in either Bypass Mode or No Link Mode, depending on how the Power Off Mode Register (Reg. 51) is set. Once the wait timer expires, then the behavior of the Niagara 22xxed relays will be governed by the Default Mode Register and Current Mode Register. The start up wait time can be from 0 sec to 510 sec (8 min and 30 sec).

2.2.2.1.8 Start Up Wait Override Register (Reg 5)

When this register is set to 1, the start up wait will terminate after the Niagara 22xxed card receives two heartbeats, even before the wait timer expires. When this register is set to 0, the start up wait cannot terminate until the wait timer expires.

2.2.2.1.9 Power Off Mode Register (Reg 51)

When this register is set to 1, the Niagara 22xxed will set the relays to Bypass Mode upon power down. When this register is set to 0, the Niagara 22xxed will set the relays to No Link Mode upon power down.

2.2.2.1.10 OEM ID (Reg 65)

This location is programmed at the factory with the OEM customer identification number. Interface Masters assigns ID numbers to OEM customers per their request.
The default value for this register is 0x26.

2.2.2.1.11 Product ID (Reg 66)

This location holds product identification number for the Niagara 22xxed cards.

2.2.2.1.12 Product Revision Number (Reg 67)

This location holds the hardware revision number for the Niagara 22xxed cards

2.2.2.1.13 Local CPU Firmware Revision Number (Reg 68)

This location holds the software revision number for local CPU for the Niagara 22xxed cards.

2.3 CPLD

2.3.1 Overview

The Niagara 22xxed boards use an Altera MAX II CPLD to allow communication between the Host and AVR CPU, and to control the basic functionality of the board.
Some of the features controlled by the CPLD are the Bypass, Reset Sequencing and AVR firmware downloading.

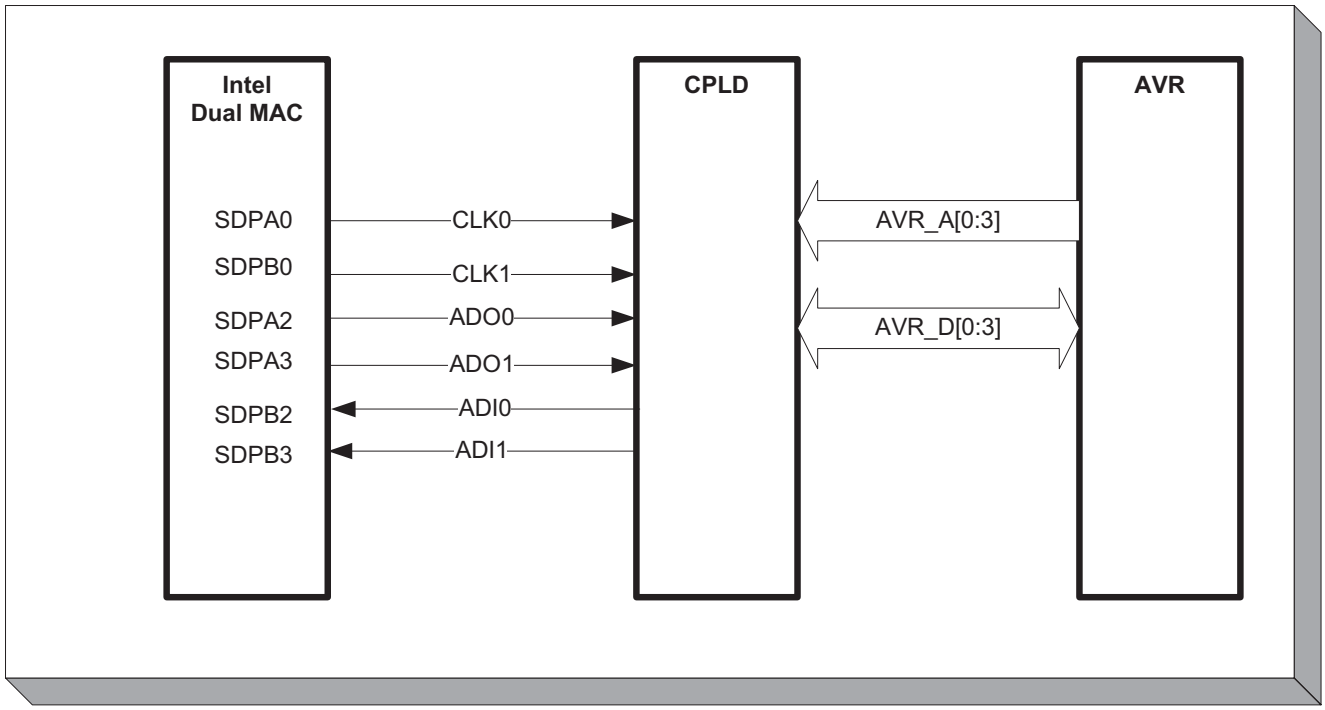


Figure 1, CPLD/AVR/MAC Interconnect

2.3.2 Internal Registers

The address map of the CPLD internal registers is shown in the table below.
The registers are accessed on a two-bit basis from the Host (MAC), and on a nibble basis from the AVR CPU. To access a byte the host must perform four access cycles while the AVR must perform two cycles.
Two additional signals from the host and one signal from the AVR are also used to select the group of bits to be accessed. The register address map is shown in the table below.

Register Name	Address Offset
AVR_ADDR	0
AVR_DAT	1
HOST_DAT	2
CTL_0	3
CTL_1	4
LED_CTL	5
MAC_CTL	6
CSR	7
SPI_CTL	8
ID_R	9
Reserved	10 (0x0A)
Reserved	11 (0x0B)
Reserved	12 (0x0C)
Reserved	13 - 15

Table 4, CPLD Registers Address Map

2.3.3 AVR_ADDR (0x00)

AVR Address register, written by the host
8-bit register used to pass an address byte from the Host to the AVR CPU.

2.3.4 AVR_DAT (0x01)

AVR Data Register, written by the host
8-bit register used to pass a data byte from the Host to the AVR CPU.

2.3.5 HOST_DAT (0x02)

Host Data Register, written by the AVR
8-bit register used to pass a data byte from the AVR CPU to the Host.

2.3.6 CTL_0 – Control Register 0 (0x03)

Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
RSVD	RSVD	RSVD	RSVD	RST_RLYA_1	SET_RLYA_1	RST_RLYA_0	SET_RLYA_0

Bits 7, 6, 5, 4 - Reserved

Bits 3, 1 RST_RLYx_y – Reset Relay, written by the AVR
These bits control the bypass relays.
A relay is Reset when the corresponding bit is pulsed high for a minimum of 3 milli_sec.

Bits 2, 0 SET_RLYx_y – Set Relay, written by the AVR
These bits control the bypass relays.
A relay is Set when the corresponding bit is pulsed high for a minimum of 3 milli_sec.

2.3.7 CTL_1 – Control Register 1 (0x04)

Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD

Bits 0 - 7 - Reserved

2.3.8 LED_CTL – LED Control Register (0x05)

Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	A_LED0

Bits 7:1 - Reserved

Bit 0 A_LEDn – Copper Activity LED Control
When this bit is set to 1, the corresponding copper status indication LED is turned ON.

2.3.9 MAC_CTL – MAC Control (0x06)

Upon reset, bits 2 and 3 are set to 1.

7	6	5	4	3	2	1	0
MAC_RST	res	SMB_SL[1]	SMB_SL[0]	SDA_O	SCL_O	SDA_I	SCL_I

bit 7 MAC_RST – MAC Reset, test only
When set to 1, the RST_MACx outputs are activated.
This bit is used during power up by the AVR CPU to reset the MACs.
Upon reset, this bit is set to 0.

bit 6 reserved

bits 5:4 SMB-SL[1:0] – SMB Bus Select
These bits are used to select one of four SMB buses.
When an SMB bus is not selected, the clock and data lines are both released.
Therefore, the software must always set SDA_O and SCL_O to 1 before changing the value of any of these bits.

bit 3 SDA_O – SDA Output.
Setting this bit to 0 forces the SDA signal low.
Setting it to 1 allows the SDA signal to be pulled high by the external pull-up resistor.
Upon reset, this bit is set to 1.

bit 2 SCL_O – SCL Output
Setting this bit to 0 forces the SCL signal low.
Setting it to 1 allows the SCL signal to be pulled high by the external pull-up resistor.
Upon reset, this bit is set to 1.

bit 1 SDA_I – SDA Input, read-only.
Returns the level of the SDA signal.

bit 0 SCL_I – SCL Input, read-only
Returns the level of the SCL signal.

2.3.10 CSR – Control and Status Register (0x07)

Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
WDI_D	WDI_C	WDI_B	WDI_A	OP_REQ	DIR	res	TEST

bit 7 WDI_D – Heart Beat Indication D, set by the host, cleared by the AVR
Set to 1 when the host writes a 1 into this bit position.
Cleared when the AVR writes a 1 into this bit position.

bit 6 WDI_C – Heart Beat Indication C, set by the host, cleared by the AVR
Set to 1 when the host writes a 1 into this bit position.
Cleared when the AVR writes a 1 into this bit position.

bit 5 WDI_B – Heart Beat Indication B, set by the host, cleared by the AVR
Set to 1 when the host writes a 1 into this bit position.
Cleared when the AVR writes a 1 into this bit position.

bit 4 WDI_A – Heart Beat Indication A, set by the host, cleared by the AVR
Set to 1 when the host writes a 1 into this bit position.
Cleared when the AVR writes a 1 into this bit position.

bit 3 OP_REQ – AVR Operation Request, set by the Host, cleared by the AVR
Set by the host by writing a 1 into this bit position to indicate to the AVR that a command is ready to be processed.
Cleared by the AVR by writing a 1 into this bit position to indicate that the operation has been completed.
Writing a 0 into this bit position has no effect.

bit 2 DIR – Data Direction, written by the Host
If set to 1 indicates that the data byte in AVR_DAT is being transferred from the Host to the AVR.
If set to 0 indicates that the Host is requesting a data byte from the AVR.
DIR can be changed only when a 1 is written into the OP_REQ bit position.

bit 1 reserved

bit 0 TEST – Test Bit

2.3.11 SPI_CTL – SPI Control Register, written by the host (0x08)

This register controls the AVR SPI interface.
It is written by the Host.

7	6	5	4	3	2	1	0
SPI_MISO	res	SPI_MOSI	SPI_CLK	PWR_GD	res	WB_FLG	SW_RST

bit 7 SPI_MISO – SPI Data Input, read-only
Returns the level of the SPI_MISO Signal

bit 6 reserved

bit 5 SPI_MOSI – SPI Data Output
Drives the SPI_MOSI output signal.

bit 4 SPI_CLK – SPI Clock
Drives the SPI_CLK Output signal

bit 3 PWR_GD – Power Good, read-only
Returns the status of the Power Good signal

bit 2 reserved

bit 1 WB_FLG – Warm Boot Flag
This bit is set by loading 0x55 into the AVR_DAT register (addr offset 1) and by writing a 1 into this bit position.
It is cleared upon power up and by writing a 1 into this bit position while the AVR_DAT register holds a value other than 0x55.

bit 0 SW_RST – Software Reset
When this bit is 1, the AVR is forced into RESET state.
Upon reset, this bit is set to 0.

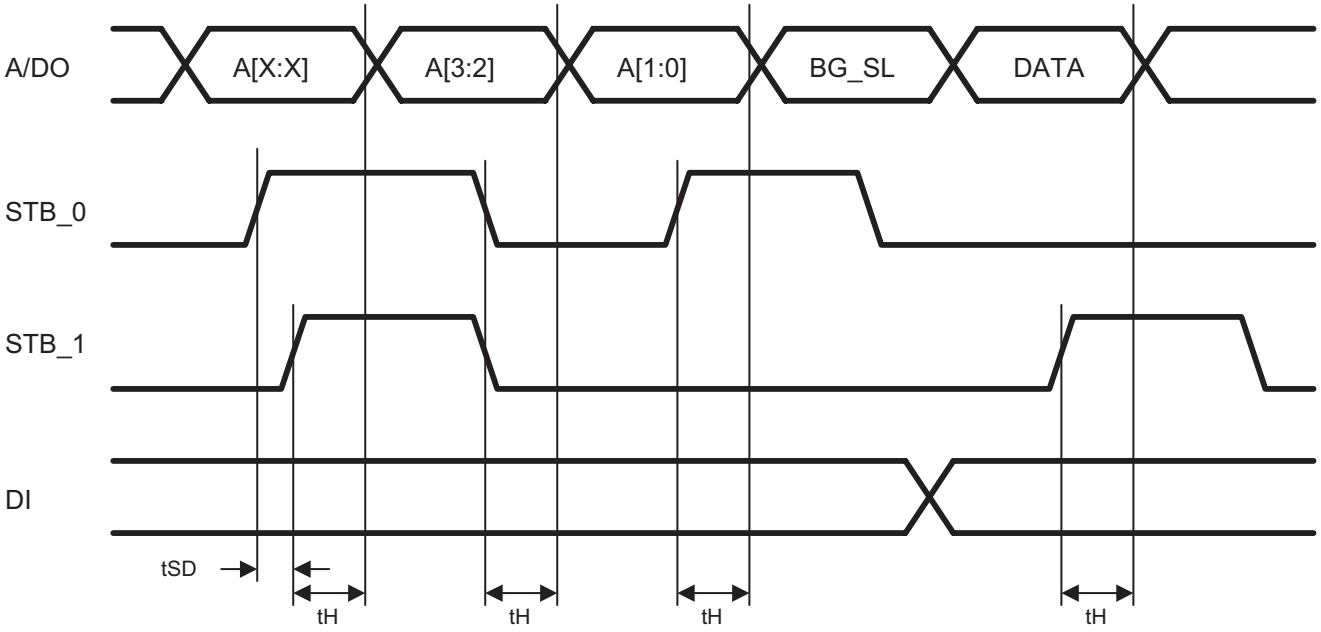
2.3.12 ID_R – ID Register, read only (0x09)

7	6-4	3-0
res	BRD_REV[2:0]	CPLD_REV[3:0]

bit 7 reserved

bits 6:4 BRD_REV[2:0] – Board Revision Level

bits 3:0 CPLD_REV[3:0] – CPLD Revision Level



tSD: min 0 ns
tH: min 120 ns

Figure 2, Host access cycles

Figure 3, AVR read cycle

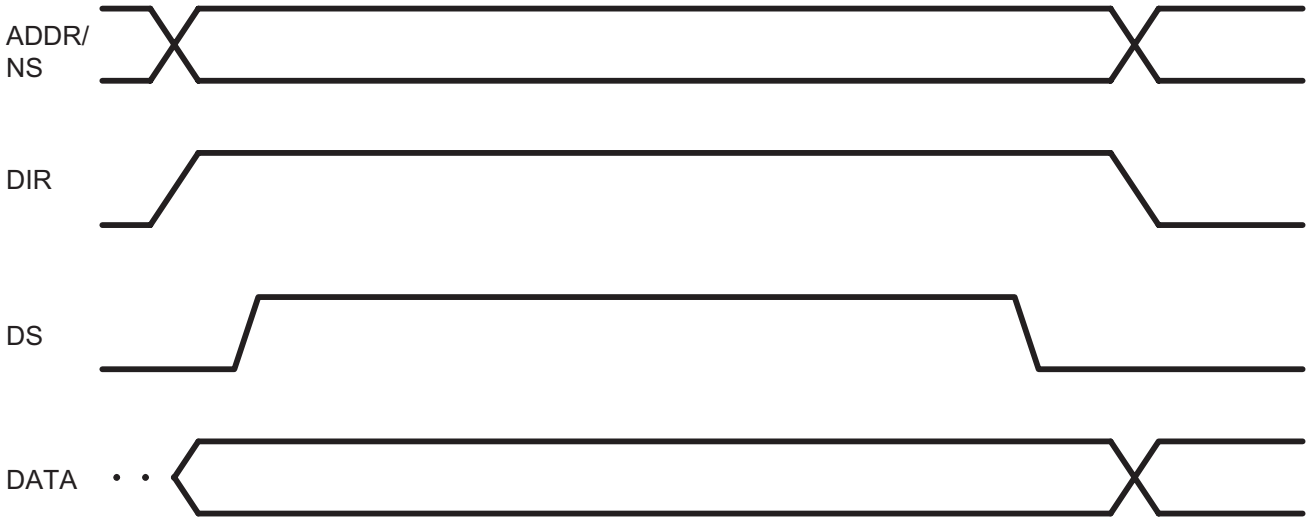


Figure 4, AVR write cycle

3. How to execute command

3.1 Command Set

We provide a command line utility program (n22xxed_util) along with our drivers. To use it, just type in "./n22xxed_util" along with any of the commands shown below.

The following table is a list of commands you can use with our command line utility.

Command	Description	Syntax
Write	Write to register or memory on the local CPU.	./n22xxed_util -w addr=[register address in decimal] -w data=[data to be written in decimal or hex]
Read	Read from register or memory on the local CPU. The contents that a read back are in hex.	./n22xxed_util -r addr=[register address in decimal]
Card select	Selects the particular card to send command to if more than 1 n22xxed card is installed.	./n22xxed_util –p [card #, starting with 0] [-command]
-a	Add a periodic heartbeat.	./n22xxed_util –a
-i	View card information. (selecting a port pair has no effect)	./n22xxed_util –i
-k	Issue single heartbeat (kick).	./n22xxed_util –k
-d	Remove periodic heartbeat.	./n22xxed_util –d
-b	Dumps content of local CPU EEPROM memory.	./n22xxed_util –b
-c	Force Active (close relay).	./n22xxed_util –c
-o	Force Bypass (open relay).	./n22xxed_util –o
-x	Force No Link.	./n22xxed_util –x
-n	Reset to normal mode.	./n22xxed_util –n
-0	Set default mode 0.	./n22xxed_util –0
-1	Set default mode 1.	./n22xxed_util –1
-2	Set default mode 2.	./n22xxed_util –2
-3	Set default mode 3.	./n22xxed_util –3
-4	Set default mode 4.	./n22xxed_util -4
-5	Set default mode 5.	./n22xxed_util -5
-L	Update local AVR firmware.	./n22xxed_util –L <binary firmware image>
-e	Dump the MAC EPROM configuration.	./n22xxed_util –e

Table 5, Command set

3.2 How to set the timeout

A timeout value of 0x1, results in 100 Msec of timeout period, while a timeout value of 0xFF results in 25.5 seconds of timeout period.

In order to set the time out, select the desired timeout value and write to timeout register.
The time out register is at location 0x0.

Here is the an example to set the timeout for 2 second:
./n22xxed_util -w addr=0 -w data=20

3.3 How to add the heartbeat

The Heartbeat is the kick that is issued by the host PC to the local CPU to keep the Niagara 22xxed in the active mode. There are two kinds of heartbeat, periodic or single.

3.3.1 Periodic Heartbeat

A periodic heartbeat could be issued by the following command
./n22xxed_util –a

3.3.2 Single Heartbeat

A single heartbeat (kick) could be issued by the following command
./n22xxed_util –k

3.4 How to remove the heartbeat

Execute the following command to delete the heartbeat
./n22xxed_util –d

3.5 How to force the bypass mode

The forced bypass command is to take the Niagara 22xxed board from active mode to bypass mode without waiting for the timeout counter to expire.
Here is the command syntax
./n22xxed_util -o

3.6 How to force active mode

Use this command to put the card to active state

./n22xxed_util -c

3.7 How to force no link mode

Use this command to put the card to active state

./n22xxed_util -x

3.8 How to resume to normal mode

Use this command to put the card to normal state

./n22xxed_util –n

3.9 How to write or read to registers

3.9.1 Write

Here is the syntax for the write register command
./n22xxed_util -w addr=[register address in decimal] -w data= [data to be written in decimal or hex]

For example in order to write to value of 0xA6(hex) to address 10(dec), which is one of the general purpose registers
./n22xxed_util -w addr=10 -w data=0xA6

3.9.2 Read

Here is the syntax for the read register command
./n22xxed_util -r addr= [register address in decimal]

For example to read OEM ID register at address 65
./n22xxed_util -r addr=65

You should read a value of 0x26

3.10 How to set default mode 0

Use this command to set default mode 0, which puts the card in bypass mode as power up, and active mode when receiving heartbeat from the host.

./n22xxed_util -0

3.11 How to set default mode 1

Use this command to set default mode 1, which puts the card in active mode as power up, and bypass mode when receiving heartbeat from the host.

./n22xxed_util -1

3.12 How to set default mode 2

Use this command to set default mode 2, which puts the card in active mode at all times, and ignores the kick, add, and delete timer commands.

./n22xxed_util –2

3.13 How to set default mode 3

Use this command to set default mode 3, which puts the card in no link mode as power up, and active mode when receiving heartbeat from the host.

./n22xxed_util –3

3.14 How to set default mode 4

Use this command to set default mode 4, which puts the card in bypass mode at all times, and ignores the kick, add, and delete timer commands.

./n22xxed_util –4

3.15 How to set default mode 5

Use this command to set default mode 5, which puts the card in no link mode at all times, and ignores the kick, add, and delete timer commands.

./n22xxed_util –5

3.16 How to update local AVR firmware

Use this command to update the local AVR firmware.
./n22xxed_util –L <firmware binary image path>

3.17 How to dump a MAC EEPROM configuration

Use this command to dump a MAC EEPROM configuration.
./n22xxed_util –e

3.18 Document History

REVISION	AUTHOR	DATE EFFECTIVE	ECO NO.	NATURE OF CHANGE
1	Kenny Wong	6/12/2009		Genesis

Table 6, Document History