SystemBase PCI(e) MultiPort Installation Manual

for Linux platform

Ver. 1.7

Installation Linux Device Driver

1. Please check whether the power is off or not in your PC.

2. Install a PCI MultiPort in the PCI slot of the PC.

3. If you have any external cable for each port, please connect the cable to the board.

4. Turn on the PC.

5. After the Linux operating system booted, log-in with administrator id as root.

6. Please copy the device driver supported from Golden Tulip to current working directory.

```
root@utu:/tmp# ls
eh_async_mpdrv.v17.sh
root@utu:/tmp#
```

7. The filename is "eh_async_mpdrv.version.sh". The current version number is v1.7.
   So you can use the device driver file, "eh_async_mpdrv.v17.sh"

8. Execute the device driver file.
   The device driver file is an executable file. You could just type the name in shell.
   After you execute the file, you can see a sub directory named "eh_async_mpdrv.v17" with installation information like a below image.
   You could see the followings; kind of multiport, type of serial interface-RS232/RS422/RS485, port name and version in the installation information

```
root@utu:/tmp# ./eh_async_mpdrv.v17.sh
Verifying archive integrity... All good.
Uncompressing Enhanced Async Multi-Port(PCI/PCIe) Linux device drvier installer.....
===============================================================
        Enhanced Async Multi-Port(PCI/PCIe) Linux Device Driver
        Version : 4.6          revision: 2012-02-03
===============================================================
1 board(s) installed
 Board No.1 : Multi-4 PCI (rev b0)
        /dev/ttyMP0 (RS232 , 16C105X)
        /dev/ttyMP1 (RS232 , 16C105X)
        /dev/ttyMP2 (RS232 , 16C105X)
        /dev/ttyMP3 (RS232 , 16C105X)
root@utu:/tmp#
```

```
root@utu:/tmp# ls
eh_async_mpdrv.v17          eh_async_mpdrv.v17.sh
root@utu:/tmp#
```

The following files exist in "eh_async_mpdrv.v17" directory.

```
root@utu:/tmp/eh_async_mpdrv.v17# ls
Install     Remove  async_multiport     ioctl       multidrop_test
root@utu:/tmp/eh_async_mpdrv.v17#
```

9. Execute the test file.

   #cd async_multiport

   #./sb_test

   If you want to know how to use the sb_test, you just type the name without any argument. And then you can see the method of the usage.

```
root@utu:/tmp/eh_async_mpdrv.v17/async_multiport# ./sb_test
Usage: ./sb_test [Port Name] [Baudrate] [TestMode]
Port Name : /dev/ttyMP0 ~ /dev/ttyMP32
Baudrate  : 9600, 19200, ...
TestMode  : 0(Loopback) 1(Send) 2(Recv)
root@utu:/tmp/eh_async_mpdrv.v17/async_multiport#
```

Usage : ./sb_test [Port Name] [Baudrate] [TestMode]

Port Name	: /dev/ttyMP0 ~ /dev/ttyMP32
Baudrate	: 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600
TestMode	: 0 (Loopback)
	1 (Send)
	2 (Receive)

Example :

$./sb_test /dev/ttyMP0 9600 0

$./sb_test/dev/ttyMP5 921600 0

$./sb_test/dev/ttyMP3 115200 1

After you connect a loopback connector to a port, you can test using loopback mode.

The test pattern is "abcdefghijklmnopqrstuvwxyz" and the program generates characters from "a" to "z" repeatedly increasing by one.

```
root@utu:/tmp/eh_async_mpdrv.v17/async_multiport# ./sb_test /dev/ttyMP0 9600 0
Loopback Test Mode   !
a
ab
abc
abcd
abcde
abcdef
abcdefg
abcdefgh
abcdefghi
abcdefghij
abcdefghijk
abcdefghijkl
abcdefghijklm
abcdefghijklmn
abcdefghijklmno
abcdefghijklmnop
abcdefghijklmnopqr
abcdefghijklmnopqrs
```

When the multiport is installed correctly, you can see the test pattern like upper picture repeatedly.

Remove Linux Device Driver

1. Move to installed directory. If you installed at "~/tmp", you move to"~/tmp/eh_async_mpdrv.v17".

2. Please execute **Remove** command.

   $./Remove

```
root@utu:/tmp/eh_async_mpdrv.v17# ls
Install     Remove  async_multiport     ioctl       multidrop_test
root@utu:/tmp/eh_async_mpdrv.v17# ./Remove
```

3. All the installed files will be removed automatically after executing Uninstall command.

```
Remove Multiports PCI/PCIe Driver..!!
remove device(/dev).....done
modify rc.local.....done
root@utu:/tmp/eh_async_mpdrv.v17#
```

Change the Linux Driver Setting Value

#cd eh_async_mpdrv.v17/ioctl

#./sb_ioctl

```
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ls
Makefile  sb_ioctl   sb_ioctl.c  sb_ioctl.o
root@utu:/tmp/eh_async_mpdrv.v17/ioctl#
```

If you want to know how to use the sb_ioctl, you just type the name without any argument. And then you can see the method of the usage.

```
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ./sb_ioctl
        Usage     : ./sb_ioctl Command [Port Name] [Parameter]
        Command : GETINFO           > Print All Port Information(DeepFifo,FCR,TTR,RTR)
                  : setdeepfifo     > Setting [Port Name] of deepfifo to [Parameter]
                  : getdeepfifo     > Printf [Port Name] of deepfifo
                  : setfcr          > Setting [Port Name] of FCR to [Parameter]
                  : getfcr          > Printf [Port Name] of FCR
                  : setttr          > Setting [Port Name] of TTR to [Parameter]
                  : getttr          > Printf [Port Name] of TTR
                  : setrtr          > Setting [Port Name] of RTR to [Parameter]
                  : getrtr          > Printf [Port Name] of RTR
        Port Name : /dev/ttyMP0 ~ /dev/ttyMP256
        Parameter : Parameter is Hexadecimal Number
root@utu:/tmp/eh_async_mpdrv.v17/ioctl#
```

First word refers to the command.

Second word refers to the port directory.

Third word refers to the setting value.

Usage : ./sb_ioctl Command [Port Name] [Parameter]

Command : GETINFO > Print All Port Information(DeepFifo,FCR,TTR,RTR)

: setdeepfifo > Setting [Port Name] of deepfifo to [Parameter]

: getdeepfifo > Printf [Port Name] of deepfifo

: setfcr > Setting [Port Name] of FCR to [Parameter]

: getfcr > Printf [Port Name] of FCR

: setttr > Setting [Port Name] of TTR to [Parameter]

: getttr > Printf [Port Name] of TTR

: setrtr > Setting [Port Name] of RTR to [Parameter]

: getrtr > Printf [Port Name] of RTR

Port Name : /dev/ttyMP0 ~ /dev/ttyMP256

Parameter : Parameter is Hexadecimal Number

Example :

$./sb_ioctl GETINFO

$./sb_ioctl setdeepfifo /dev/ttyMP0 0

$./sb_ioctl setdeepfifo /dev/ttyMP0 1

$./sb_ioctl setfcr /dev/ttyMP0 01

$./sb_ioctl setttr /dev/ttyMP0 38

$./sb_ioctl setrtr /dev/ttyMP0 10

For example, if you would like to see the setting values for every single port, type "#./sb_ioctl GETINFO" and it shows the following.

```
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ./sb_ioctl GETINFO
```

```
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ./sb_ioctl GETINFO
===DEEPFIFO INFO===
MP0=1    MP2=1    MP3=1    MP3=1


===TTRINFO===
MP0=38   MP2=38   MP3=38   MP3=38


===RTRINFO===
MP0=10   MP2=10   MP3=10   MP3=10


===FCRINFO===
MP0=1    MP2=1    MP3=1    MP3=1

root@utu:/tmp/eh_async_mpdrv.v17/ioctl#
```

For example, if you would like to deactivate '/dev/ttyMP0으| 256byte FIFO', type '#./sb_ioctl setdeepfifo /dev/ttyMP0 0'

```
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ./sb_ioctl setdeepfifo /dev/ttyMP0 0
ret = 0
root@utu:/tmp/eh_async_mpdrv.v17/ioctl# ./sb_ioctl GETINFO
===DEEPFIFO INFO===
MP0=1    MP2=1    MP3=1    MP3=1


===TTRINFO===
MP0=8    MP2=38   MP3=38   MP3=38


===DEEPFIFO INFO===
MP0=8    MP2=10   MP3=10   MP3=10


===DEEPFIFO INFO===
MP0=1    MP2=1    MP3=1    MP3=1

root@utu:/tmp/eh_async_mpdrv.v17/ioctl#
```