Freescale Semiconductor
Application Note

Document Number:AN4544
Rev. 0, 6/2012

i.MX28 Ethernet Performance on Linux

Contents

1 Introduction

The aim of this document is to show how to measure the ENET "Ethernet Controller" performance on the i.MX28 EVK/BSP platform and present the results.

The results were taken with two hardware configurations; one with the image running from NAND, and the other from SD card.

2 ENET Benchmark Performance Test Setup

The following sections describe ENET Benchmark Performance Test Setup.

2.1 Hardware Configuration

All the tests described in this document were done with the hardware described below:

- Target Processor: i.MX28
- Target Platform: i.MX28 EVK PCB Rev D
- Host Processor: Intel Pentium D 2.80GHz
- NAND: Samsung K9GAG08U0M, 2GB
- SD: SanDisk 4GB SDHC Class 2

2.2 Software Configuration

- Host Operating System: i386 Linux Ubuntu 10.04 LTS (lucid)
- Host Kernel Version: 2.6.32-33-generic (#70-Ubuntu SMP Thu Jul 7 21:09:46 UTC 2011)
- MX28EVK Linux BSP Version: L2.6.35_10.12.01
- Host Iperf version: iperf version 2.0.4 (7 Apr 2008) pthreads
- Host SCP Version: OpenSSH 1:5.3p1-3ubuntu7, OpenSSL 0.9.8k-7ubuntu8.6
- Target Kernel Version: Freescale Linux 2.6.35.3-571-gcca29a0 #8 PREEMPT Wed Jan 4 09:23:36 PST 2012
- Target IPerf Version: iperf version 1.7.0 (13 Mar 2003) pthreads
- Target SCP Version: Dropbear sshd v0.52
- Target FTP version: ftp (GNU inetutils) 1.4.2 available from LTIB

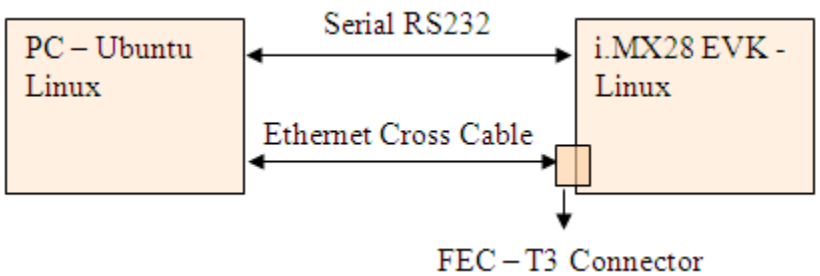

Figure 1. ENET Benchmark Topology

2.3 Chip Errata Note

Due to i.MX28 chip errata ENGR121613, "ENET: ENET big endian mode not compatible with ARM little endian," Ethernet performance is limited by the requirement that the driver perform byte swapping. When both interfaces (eth0 and eth1) are used concurrently, they must share CPU bandwidth to perform byte swapping.

2.4 Configuring the Target

Using ltib --selectype, select the min profile. Using ltib -c, select the configurations indicated below.

2.4.1 Target System Configuration

- Target System Options : Start Networking : Network Setup
  - Enable Interface 0
  - Configure static parameters
  - Alternatively, configure static parameters from target command line after boot. For example:
    - ifconfig eth0 192.168.1.2 netmask 255.255.255.0 broadcast 192.168.1.255
- Target Image Options : Root filesystem image type: ext2.gz

2.4.2 Package list

- dropbear ssh client/server

- inetutils
- iperf
- mtd-utils
- kobs-ng
- boot stream
- kernel command line (SD Card)

```
noinitrd console=ttyAM0,115200 root=/dev/mmcblk0p3 rw rootwait ip=none gpmi
```

- kernel command line (NAND)

```
noinitrd console=ttyAM0,115200 ubi.mtd=1 root=ubi0:rootfs0 rootfstype=ubifs rw ip=none gpmi
```

2.5 Target Linux Kernel Configuration

Select:

- Device Drivers
- Network Device Support
- Ethernet (10 or 100Mbit)
- FEC Ethernet Controller

2.5.1 Target File System Configuration

For SD Card, ext2 file system is used. For NAND, UBIFS is used. For Samsung K9GAG08U0M NAND, the following UBI configuration parameters are used:

- `mkfs.ubifs -x none -m 4096 -e 516096 -c 4000 -r rootfs rootfs.ubifs`
  - No compression
  - NAND block size = 4096
  - Logical Erase Block Size (LEB) = 516096
  - Max Logical Erase Blocks = 4000

2.5.2 Target user

In order to write to the target using FTP and SCP, a user should be created with the command:

- adduser <username>

2.6 Configure the Host

The Ubuntu host requires additional packages and network configuration.

2.6.1 Packages

- `sudo apt-get install iperf`
- `sudo apt-get install vsftp`
- `sudo apt-get install openssh-server`

2.6.2 Network Configuration

The direct Ethernet connection to the target must be configured. For example:

- `ifconfig eth0 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255`

3 Test Procedure

- Each test will be executed 5 times, the result will be the average.
- To make files with specific size, the following commands were used:
  - For 100MB file size:

    ```
    $ dd if=/dev/zero of=100Mfile bs=100K count=1024
    ```
  - For 10MB file size:

    ```
    $ dd if=/dev/zero of=10Mfile bs=10K count=1024
    ```
- File transfer tests will measure throughput for read and write.

3.1 Buffer Length

Performance for both TCP and UDP is measured using various buffer lengths. As can be seen in the results below, buffer length has a large impact on performance.

4 SD Image Test Descriptions and Results

The following sections describe SD image test descriptions and results.

4.1 Network Throughput (TCP and UDP [SD Image Test])

The method to measure network throughput consist of sending UDP and TCP packets from target to host using the open source iperf program.

TCP

On host, type the command: iperf -s

The table below shows results running iperf on the target with no arguments. This can be considered a typical use case.

Table 1. Network Throughput TCP - Default Arguments

Name	Target Command	Buffer length	Interval [sec]	Throughput [Mbits/sec]
Test-1.1-defaults	iperf -c $HOST_IP	8KB	10	60.1

The table below shows results running iperf on the target with various combinations of buffer length and number of bytes to transmit.

Table 2. Network Throughput TCP - Various Arguments

Name	Target Command	Buffer length	number of bytes to transmit	Throughput [Mbits/sec]
Test-1.1-a	iperf -c $HOST_IP -l 8K -n 1K	8KB	1KB	6.73
Test-1.1-b	iperf -c $HOST_IP -l 8K -n 2K	8KB	2KB	6.84
Test-1.1-c	iperf -c $HOST_IP -l 8K -n 4K	8KB	4KB	7.17
Test-1.1-d	iperf -c $HOST_IP -l 8K -n 8K	8KB	8KB	7.04
Test-1.1-e	iperf -c $HOST_IP -l 256K -n 32K	256KB	32KB	53.20
Test-1.1-f	iperf -c $HOST_IP -l 256K -n 64K	256KB	64KB	53.24
Test-1.1-g	iperf -c $HOST_IP -l 256K -n 128K	256KB	128KB	53.32
Test-1.1-h	iperf -c $HOST_IP -l 256K -n 256K	256KB	256KB	52.70
Test-1.1-i	iperf -c $HOST_IP -l 8M -n 1M	8MB	1MB	67.84
Test-1.1-j	iperf -c $HOST_IP -l 8M -n 2M	8MB	2MB	68.34
Test-1.1-k	iperf -c $HOST_IP -l 8M -n 4M	8MB	4MB	68.62
Test-1.1-l	iperf -c $HOST_IP -l 8M -n 8M	8MB	8MB	68.54
Test-1.1-m	iperf -c $HOST_IP -l 8M -n 1K	8MB	1KB	68.24
Test-1.1-n	iperf -c $HOST_IP -l 8M -n 2K	8MB	2KB	68.66
Test-1.1-o	iperf -c $HOST_IP -l 8M -n 4K	8MB	4KB	68.52
Test-1.1-p	iperf -c $HOST_IP -l 8M -n 8K	100MB	8KB	68.52
Test-1.1-q	iperf -c $HOST_IP -l 100M -n 1M	100MB	1MB	67.80
Test-1.1-r	iperf -c $HOST_IP -l 100M -n 25M	100MB	25MB	67.78
Test-1.1-s	iperf -c $HOST_IP -l 100M -n 50M	100MB	50MB	67.84
Test-1.1-t	iperf -c $HOST_IP -l 100M -n 100M	100MB	100MB	71.08

UDP

On host, type iperf -su to enable receiving of UDP datagrams.

The -b parameter (in target command) determines the maximum bandwith that iperf will transmit. In Table 2, Test-1.2-a can be considered a typical use case.

Ping Time Response

Table 3. Network Throughput UDP

Name	Target Command	Length Buffer [kB]	Throughput [Mbits/sec]
Test-1.2-a	iperf -c $HOST_IP -u -b 100M	- (default)	89.60
Test-1.2b	iperf -c $HOST_IP -u -b 100M -l 12	12	0.980
Test-1.2c	iperf -c $HOST_IP -u -b 100M -l 120	120	9.58
Test-1.2d	iperf -c $HOST_IP -u -b 100M -l 1200	1200	78.66

Ping Time Response is measured using the command: ping <host IP address> -s <packet size>

In Table 3, Test-2.1-a can be considered a typical use case.

Table 4. Ping Time Response

Name	Target Command	Packet size	Time Response [ms]
Test-2.1-a	ping -s 64 $HOST_IP	64B	0.36
Test-2.1-b	ping -s 16384 $HOST_IP	16384B	4.22
Test-2.1-c	ping -s 32768 $HOST_IP	32768B	8.17

4.2 File Transfer using SCP (SD Image Test)

File transfer using SCP is measured using the command: scp <file> <user@host:file>

Table 5. File Transfer using SCP

Name	Target Command	File size	Write Transfer Rate [Mbits/s]
Test-3.1-a	scp <file> <host>:<file>	100MB	24.80
Test-3.1-b	scp <file> <host>:<file>	10MB	25.12
	Host Command	File size	Read Transfer Rate [Mbits/s]
Test-3.1-c	scp <file> <target>:/dev/null	100MB	10.56
Test-3.1-d	scp <file> <target>:/dev/null	10MB	10.88

4.3 File Transfer using FTP (SD Image Test)

File transfer using FTP is measured using the command:

```
ftp $HOST_IP
<autentication>
send <file>
```

Table 6. File Transfer using TFTP

Name	Target Command	File size	Write Transfer Rate [Mbits/s]
Test-4.1-a	ftp $HOST_IP send <file>	100MB	54.46
Test-4.1-b	ftp $HOST_IP send <file>	10MB	54.56
	Host Command	File size	Read Transfer Rate [Mbits/s]
Test-4.1-c	ftp $TARGET_IP send <file> /dev/null	100MB	71.93
Test-4.1-d	ftp $TARGET_IP send <file> /dev/null	10MB	61.65

5 NAND Image Test Descriptions and Results

The following sections describe NAND image test descriptions and results.

5.1 Network Throughput (TCP and UDP [NAND Image Test])

The method to measure network throughput consist of sending UDP and TCP packets from target to host using the open source iperf program.

TCP

On host, type the command: iperf -s

The table below shows results running iperf on the target with no arguments. This can be considered a typical use case.

Table 7. Network Throughput TCP - Default Arguments

Name	Target Command	Buffer length	Interval [sec]	Throughput [Mbits/sec]
Test-1.1-defaults	iperf -c $HOST_IP	8KB	10	61.70

The table below shows results running iperf on the target with various combinations of buffer length and number of bytes to transmit.

Table 8. Network Throughput TCP - Various Arguments

Name	Target Command	Buffer length	number of bytes to transmit	Throughput [Mbits/sec]
Test-1.1-a	iperf -c $HOST_IP -l 8K -n 1K	8KB	1KB	7.07
Test-1.1-b	iperf -c $HOST_IP -l 8K -n 2K	8KB	2KB	7.11
Test-1.1-c	iperf -c $HOST_IP -l 8K -n 4K	8KB	4KB	7.09
Test-1.1-d	iperf -c $HOST_IP -l 8K -n 8K	8KB	8KB	7.08
Test-1.1-e	iperf -c $HOST_IP -l 256K -n 32K	256KB	32KB	53.12
Test-1.1-f	iperf -c $HOST_IP -l 256K -n 64K	256KB	64KB	53.28
Test-1.1-g	iperf -c $HOST_IP -l 256K -n 128K	256KB	128KB	53.16
Test-1.1-h	iperf -c $HOST_IP -l 256K -n 256K	256KB	256KB	53.26
Test-1.1-i	iperf -c $HOST_IP -l 8M -n 1M	8MB	1MB	70.00
Test-1.1-j	iperf -c $HOST_IP -l 8M -n 2M	8MB	2MB	70.86
Test-1.1-k	iperf -c $HOST_IP -l 8M -n 4M	8MB	4MB	71.00
Test-1.1-l	iperf -c $HOST_IP -l 8M -n 8M	8MB	8MB	70.90
Test-1.1-m	iperf -c $HOST_IP -l 8M -n 1K	8MB	1KB	70.70
Test-1.1-n	iperf -c $HOST_IP -l 8M -n 2K	8MB	2KB	70.86
Test-1.1-o	iperf -c $HOST_IP -l 8M -n 4K	8MB	4KB	71.00

Table continues on the next page...

Table 8. Network Throughput TCP - Various Arguments (continued)

Test-1.1-p	iperf -c $HOST_IP -l 8M -n 8K	100MB	8KB	70.96
Test-1.1-q	iperf -c $HOST_IP -l 100M -n 1M	100MB	1MB	70.26
Test-1.1-r	iperf -c $HOST_IP -l 100M -n 25M	100MB	25MB	71.02
Test-1.1-s	iperf -c $HOST_IP -l 100M -n 50M	100MB	50MB	70.98
Test-1.1-t	iperf -c $HOST_IP -l 100M -n 100M	100MB	100MB	71.08

UDP

On host, type iperf -su to enable receiving of UDP datagrams.

The -b parameter (in target command) determines the maximum bandwidth that iperf will transmit. In Table 7, Test-1.2-a can be considered a typical use case.

Table 9. Network Throughput UDP

Name	Target Command	Length Buffer [kB]	Throughput [Mbits/sec]
Test-1.2-a	iperf -c $HOST_IP -u -b 100M	- (default)	89.76
Test-1.2b	iperf -c $HOST_IP -u -b 100M -l 12	12	0.99
Test-1.2c	iperf -c $HOST_IP -u -b 100M -l 120	120	9.64
Test-1.2d	iperf -c $HOST_IP -u -b 100M -l 1200	1200	78.22

5.2 Ping Time Response (NAND Image Test)

Ping time response is measured using the command: ping <host IP address> -s <packet size>

In Table below, Test-2.1-a can be considered a typical use case.

Table 10. Ping Time Response

Name	Target Command	Packet size	Time Response [ms]
Test-2.1-a	ping -s 64 $HOST_IP	64B	0.38
Test-2.1-b	ping -s 16384 $HOST_IP	16384B	4.24
Test-2.1-c	ping -s 32768 $HOST_IP	32768B	8.13

5.3 File Transfer using SCP (NAND Image Test)

File transfer using SCP is measured using the command: scp <file> <user@host:file>

Table 11. File Transfer using SCP

Name	Target Command	File size	Write Transfer Rate [Mbits/s]
Test-3.1-a	scp <file> <host>:<file>	100MB	22.24

Table continues on the next page...

Table 11. File Transfer using SCP (continued)

Test-3.1-b	scp <file> <host>:<file>	10MB	23.36
	Host Command	File size	Read Transfer Rate [Mbits/s]
Test-3.1-c	scp <file> <target>:/dev/null	100MB	10.88
Test-3.1-d	scp <file> <target>:/dev/null	10MB	11.00

5.4 File Transfer using FTP (NAND Image Test)

File transfer using FTP is measured using the command :

```
ftp $HOST_IP
<autentication>
send <file>
```

Table 12. File Transfer using TFTP

Name	Target Command	File size	Write Transfer Rate [Mbits/s]
Test-4.1-a	ftp $HOST_IP send <file>	100MB	23.18
Test-4.1-b	ftp $HOST_IP send <file>	10MB	54.38
	Host Command	File size	Read Transfer Rate [Mbits/s]
Test-4.1-c	ftp $TARGET_IP send <file> /dev/null	100MB	76.01
Test-4.1-d	ftp $TARGET_IP send <file> /dev/null	10MB	71.79

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or
+1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku
Tokyo 153-0064
Japan
0120 191014 or
+81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

Document Number: AN4544
Rev. 0
06/2012