A Meta Analysis of Gigabit Ethernet over Copper Solutions for Cluster-Networking

Torsten Hoefler, Wolfgang Rehm

{htor,rehm}@informatik.tu-chemnitz.de

03th May 2004

Abstract

The IEEE Standard for Gigabit Networking was accepted in June 1998 and ratified as IEEE 802.3z. This standard uses considers an optical cable for signal transmission. One year later a new standard for Gigabit Ethernet over unshielded twisted pair of the 5th category was certified under the name 802.3ab. Nowadays, there are a couple of younger and older studies about Gigabit Ethernet technology and performance. This Meta Analysis is intended to put the main results altogether into one document suitable for a proper choice of gigabit networking equipment for cluster computers.

1 Introduction

Gigabit Ethernet is mostly derived from Fast Ethernet, and inherited most of its features. Nevertheless, several improvements have been added on the basis of the higher speed, for example to reduce the load on the sending and receiving side. All those features have to be analyzed to evaluate the suitability of Gigabit Ethernet for high performance clusters.

1.1 Features

- throughput of 1000MBit/sec which equals to 125MB/sec
- Adapters are available for all known PCI standards (32bit/33MHz, 32bit/66MHz, 64bit/66MHz, PCI-X (64bit/133MHz) ...)
- it is backward compatible to legacy standards (10/100MBit Ethernet)
- all features of 10/100MBit Ethernet are inherited like CSMA/CD, full-duplex, addressing mechanism etc.
- some new features are mostly devoted to the high speed as
  - packet bursting
  - jumbo packets

1.2   Interface Types

Several physical interface types are defined for the two standards. The original 802.3z interface type is called 1000BASE-X. It splits up into two fiber link specifications and one shielded twisted pair for very short distances

- 1000Base-SX (fiber - low cost, small distance)
- 1000Base-LX (fiber - higher cost, long haul)
- 1000Base-CX (STP[1] - up to 25m)

The newer 802.3ab standard defines the 1000BASE-T interface and uses all four pairs of a category five twisted pair cable with a maximum length of 100m. Each pair is utilized to transmit 250MBit/sec full duplex. The necessary high modulation frequency causes noticeable pickups in neighbor cables. A new scrambled noise immune transmission method to handle this problem was found in PAM-5[2] encryption.

1.3   Differences To Old Ethernet

1.3.1   Minimal Frame Size

Another result of the higher frequency was that the length of the cable would be limited to 20m to allow the CSMA/CD mechanism to work properly with the standardized minimal frame size[3] of 64 bytes[4]. So the physical minimal frame size was set to 512 bytes, which actually allowed a segment diameter up to 200m. But the logical minimal frame size was left at 64 bytes mainly to keep the compatibility to the older ethernet standards. To guarantee the physical minimal frame size a new field called carrier extension was introduced which is padded up to 512 bytes (if the packets are smaller) before sending.

Remember that the CSMA/CD mechanism is not used in full duplex mode that this issue only matters for half duplex. In full duplex mode the usual 96 byte inter frame gap is used together with a physical/logical minmal packet size of 64 bit.

1.3.2   Packet Bursting

However, this introduces a possible overhead of 700% ($512\ b - 64\ b = 448\ b\ overhead$) which usually would lead to a horrendous loss of bandwidth when small packets are used. Small packets are often used in clusters for synchronization purposes, and would cause this overhead frequently. Therefore a new technique called Packet Bursting was developed. This means that small packets can be sent after one padded without any padding (but with 96 bytes gaps between them). So it is guaranteed that cable never gets "quiet"[5] and the possible collision only appears in the first frame.

1.3.3   Maximum Frame Size - Jumbo Packets

Also the amount of transferred data gets more and more. It emerged that the standard maximum frame size of 1518 bytes [6] was not sufficient for big amounts of data because too many

---

[1]Shielded Twisted Pair

[2]this increases the amount of information per Signal to five bits (five states)

[3]for 10/100MBit Ethernet

[4]the minimal segment size must allow the bidirectional transmitter to recognize any transmitting failures (collisions) before the transmission is completed - the packet must have time to reach the other end and return

[5]the network adapters waits for this before starting a new try to send a packet

[6](18 bytes header and Logical Link Control (LLC), 1500 bytes payload (MTU)

packets need to be generated, which increases the overhead and causes a high load on both communication partners to fragment and reassemble the data into those small packets. Additionally the load is increased by many interrupts (each for one received packets) on the receiving side. Consequentially, the CPU power was the new bottleneck for network transactions.

So in 1999 a new (optional) add-on to the ethernet standard was defined by some vendors. The frame sizes can be up to 9018 bytes[7], some manufacturers even support bigger frame sizes (e.g. 16k bytes). Thus, the processor load and the interrupts per transmit are reduced by 6 (or even more for bigger frames). The impact of degrading the avarage latency time with bigger packets can be easily adjusted by the admin. Packets bigger than 1500 bytes are often called Jumbo Packets.

1.4 Potential Improving Features

1.4.1 Zero-Copy - Scatter/Gather

Zero copy means that no copy operation should be necessary to send data to another host. Without this technique two copy operations, one on the sending side from memory to network card and one on the receiving side from network card to memory are required. Also a method called one-copy exist, where only one copy operation is conducted either on the receiving host or on the sending side. But oviously, zero copy is the fastest for data transferring.

There are two main models to implement zero-copy:

1. **memory mapping** - the network card's address space (mostly some PCI range) is mapped directly to the memory of the user process - this can be very complicated because it avoids the memory protection functionality of the operating system

2. **DMA** - the data is copied by a busmaster capable network card directly into the appropriate main memory block

A new approach called "speculative defragmentation" [2] may improve the overall performance and lower the latency times by a hardware based defragmentation combined with zero-copy.

1.4.2 Alignment Issues

Network interface cards should be able to copy or map the received data not only to aligned addresses. The ethernet header itself is 14 bytes long, so if the whole packet is copied to an aligned address, the operating system must access an unaligned address to reach the beginning of the IP header. This is a very expensive operation on most architectures (e.g. x86), or is even not possible on others (e.g. RISC).

1.4.3 Interrupt Moderation

Usually every arriving packet causes an interrupt. Thus gigabit interfaces will cause many interrupts especially with low MTU values. Each interrupt needs some CPU cycles to complete the request, so the overall performance suffers heavily. The NIC could combine IRQ's in times of very high traffic on the wire.

---

[7]18 byte header and LLC, 9000 byte payload

1.4.4   Hardware Checksumming - Offloading

Every packet is provided with a checksum to detect transmission errors. The checksum algorithm is quite cheap but in times of high network load the CPU can be working with full capacity on calculating and validating checksums. The calculation and/or checking process can easily be done in hardware on the NIC. This is supported by most cards, but the driver support for linux is still rudimentary.

1.4.5   Hardware Fragmentation

As mentioned before the maximum frame size (MTU) of ethernet packets is limited for various reasons. If the operating system tries to send packets bigger than the actual MTU it has to fragment the packets into several small ones. This could also be done in hardware. That means that the MTU provided to the operating system is much bigger than the physical MTU on wire and the NIC does all necessary fragmentation itself.

2   Network Interface Cards

2.1   Introduction

Network Interface Card (NIC) Design is also defined in the IEEE 802.3 standard. Simplified said, a NIC consists of two layers which correspond with OSI layer 1 (Physical) and 2 (Data Link) - they are named PHY[8] and MAC[9] (see figure below). The NIC acts as multi-layered adapter, one endpoint is given by the physical medium (cable), whose interface is strictly defined by electrical parameters (see IEEE 802.3), the other endpoint is the application. Thus there are two interfaces to be designed:

1. MAC↔Application: transmitting and receiving frames, status information
2. MAC↔PHY: framing, contention resolution (CSMA/CD), timing and handing over bit streams (transmit/receive)

These two functionalities are sometimes delivered in different chips, sometimes combined in a one chip solution. Another interface, called MII[10] or for Gigabit Ethernet GMII[11] exists to deliver status information about link speed and negotiation results to the application level - this is not relevant for this study.

---
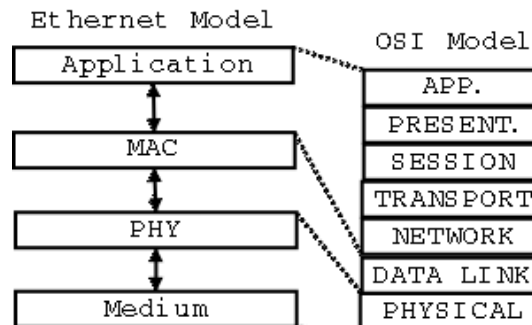
[8]Physical Layer Device
[9]Medium Access Control
[10]Medium Independent Interface
[11]Gigabit MII

2.1.1 PCI Connection

There are several possibilities to implement the connection between the processor and the NIC. This is mostly done with the common PCI[12], or one of the newer PCI-X, PCI-Express or CSA standards which are usually connected through one or more additional bridges[13] to the CPU. The following table compares transfer rates of different bus types:

type	Clock	Width	Bandwidth
-	**MHz**	**bit**	**Gbit/s**
PCI/32 2.0	33	32	1,03
PCI/32 ab 2.1	66	32	2,06
PCI/64 ab 2.0	33	64	2,06
PCI/64 ab 2.1	66	64	4,13
PCI-X 66	66	64	4,13
PCI-X 133	133	64	8,31
PCI-X 266	266	64	16,60
PCI-X 533	533	64	33,31
PCI-X 1066	1066	64	66,62
PCI-Express	-	-	128,00
CSA	-	-	2,08

Remarks:

- the bandwidth are maximum values

- PCI/32 2.0 is too slow for full duplex Gigabit Ethernet

- the throughput was calculated without considering the overhead ($throughput = clock * width$).

- PCI-Express does not fit into this table, because it offers a serial point to point link with bandwidths from 4 Gbit/s up to 128 GBit/s.

- CSA[14] does not fit into this table, because the adapter is directly connected to the MCH[15] (For further information on CSA see [4])

2.1.2 Evaluation Criteria

The available cards are grouped by the built-in connector chips (MAC) because this has a significant impact on the performance and the driver support for Linux. Then the following facts are used as evaluation criterias:

Only separate cards (no LOM[16] cards) have been analyzed!

- **bus** - available bus type for connection (PCI, CSA ...)

- **throughput** - throughput with different MTU's (1500, 3000, 6000, 9000) - if there are different values in different studys, the highest value is taken (because misconfiguration can only lead to performance loss) - all throughput values are to be seen very imprecise, but they can give a clue

---

[12]Periphal Component Interconnect
[13]Interface Controller Hub, Memory Controller Hub
[14]Communication Streaming Architecture
[15]Memory Controller Hub
[16]LAN on Motherboard

- **price** - the cheapest price (EUR, in Germany) found in the internet
- the latency was not analyzed because only one study offers latency values and these seem to be very imprecise

This study fully depends on the other studies, if there are any wrong measurements (e.g. misconfigured cards, bus problems, chipset problems ...) in the source-studies, they will be also wrong in this paper. Additional features as mentioned in 1.4 are not analyzed because it would have been to costly to ask every manufacturer for the NIC's different features (and most salesmen simply do not know this in the needed particular detail).

2.2 BCM5701 Based Cards

This chip was tested in four different studys ([5], [6], [7], [9]). It seems to work with bcm5700 or tg3 driver shipped with the vanilla kernel. The peak throughput seems to depend strongly on the used mainboard and the measured performance in a 66MHz slot was lower than in a 33MHz slot (see [5]).

bus configuration:	64Bit/33Mhz
throughput (1500):	739 Mbit/s
throughput (3000):	747 Mbit/s
throughput (6000):	786 Mbit/s
throughput (9000):	620 Mbit/s

Available products:

Name	Price
3COM 3C996B-T NIC (PCI/64)	95

2.3 3C2000 Based Cards

No published test to this chip could be found, thus no performance values are available and it is even not clear if it works with Linux.

bus configuration:	-
throughput (1500):	-
throughput (3000):	-
throughput (6000):	-
throughput (9000):	-

Available products:

Name	Price
3COM 3C2000-T (PCI/32)	-

2.4 AC100x Based Cards

This chip was analyzed by the c't newspaper and different other studies, see [9], [6] (Card: Hardlink HA-64G) and [7] (Card: Hardlink HA-64G). The AC100x based cards work with the tg3 driver included in the Linux kernel.

bus configuration:	64Bit/66MHz
throughput (1500):	516 Mbit/s
throughput (3000):	470 Mbit/s
throughput (6000):	375 Mbit/s
throughput (9000):	420 Mbit/s

Available products:

Name	Price
ALLIED Telesyn AT-2915T (PCI/32)	27
NETGEAR GA302T (PCI/64)	29
Hardlink HA-64G (PCI/64)	-
COREGA PCI-GT (PCI/32)	30

2.5   DP8382x Based Cards

This chip was analyzed by the c't newspaper and different other studies, see [9], [5] (Cards: ARK SOHO 2000T, ARK SOHO 2500T, Asante GigaNIX, D-LINK DGE-500T), [6] (Cards: SMC SMC9462TX, TRENDnet TEG-PCITX2) and [8] (Cards: LNIC-1000T, Hardlink HA-32G). This chip is supported by the generic ns83820 driver included in the Linux kernel. The throughput rate is highly dependend of the used driver version, it differs at MTU=1500 in more than 150Mbit/s (see [6]). There seem to be some driver problems, so the MTU could not be set higher than 8192.

bus configuration:	64Bit/66MHz
throughput (1500):	524 Mbit/s
throughput (3000):	525 Mbit/s
throughput (6000):	640 Mbit/s
throughput (9000):	-

Available products:

Name	Price
ALLNET ALL0124 (PCI/32)	39
NETGEAR GA622T (PCI/64)	108
ARK SOHO 2000T (PCI/64)	-
SMC SMC9462TX (PCI/64)	76
ARK SOHO 2500T (PCI/32)	-
Asante GigaNIX (PCI/64)	-
CNET GNIC2000(A) (PCI/32)	35
D-LINK DGE-500T (PCI/32)	40
TRENDnet TEG-PCITX2 (PCI/64)	-
LNIC-1000T (PCI/32)	-
Hardlink HA-32G (PCI/32)	-

2.6   Marvell Chip Based Cards

This chip was also analyzed by the c't newspaper and different other studies, see [9], [5] (Card: Syskonnect SK-9821) and [7] (Cards: Syskonnect SK-9821, Syskonnect SK-9822). The Syskonnect SK-9822 is the two-port version of SK-9821. All Cards seem to work with the vanilla kernel's sk98lin driver.

bus configuration:	64Bit/66MHz
throughput (1500):	784 Mbit/s
throughput (3000):	925 Mbit/s
throughput (6000):	939 Mbit/s
throughput (9000):	939 Mbit/s

Available products:

Name	Price
Syskonnect SK-9521 (PCI/32)	22
Syskonnect SK-9821 (PCI/64)	75
Syskonnect SK-9822 (PCI/64)	574
D-LINK DGE-550T (PCI/64)	62

2.7   DL-200x Chip Based Cards

This chip was also analyzed by the c't newspaper, see [9]. Thus the only available measured MTU value is 1500. It may work with higher MTU's but it is not proved. Linux support is also available.

bus configuration:	32Bit/33MHz
throughput (1500):	492 Mbit/s
throughput (3000):	-
throughput (6000):	-
throughput (9000):	-

Available products:

Name	Price
D-LINK DGE-510T (PCI/32)	-

2.8   ZX1701 Chip Based Cards

This chip was analyzed in study [7]. It seems to be quiet complicated to operate this card with Linux. There have been different problems with MTU values bigger than 3000, thus no measurement data is avialable for this values. So it is not recommended for Linux Cluster usage. Nevertheless the following results were achieved:

bus configuration:	64Bit/66MHz
throughput (1500):	270 Mbit/s
throughput (3000):	295 Mbit/s
throughput (6000):	-
throughput (9000):	-

Available products:

Name	Price
ZyXEL Omni Lan PCI G1 (PCI/64)	-

2.9   Intel Chip Based Cards

Intel has a lot of different chips and a lot of revisions of them on the market. So it was not possible to investigate all different chips, and the test results have to be seen generalized. Different chips have been analyzed in [9], [5], [7] and [8]. The best achieved values are listed below:

bus configuration:	64Bit/66MHz
throughput (1500):	650 Mbit/s
throughput (3000):	533 Mbit/s
throughput (6000):	620 Mbit/s
throughput (9000):	743 Mbit/s

Available products:

Name	Price
INTEL Express PRO/1000MT (PWLA8390MT) (PCI/64)	41
INTEL Express PRO/1000MT (PWLA8391MT) (PCI/64)	37
INTEL Express PRO/1000MT (PWLA8490MT) (PCI/64)	111
INTEL Express PRO/1000MT (PWLA8492MT) (PCI/64)	153

2.10   RTL8169 Chip Based Cards

This chip was also analyzed by the c't newspaper (see [9]) and [7] (Card: CNet ProG2000L). There is a driver available in the Linux standard (vanilla) kernel (r8169), but it does not allow to set the MTU value higher than 1500. Thus no jumbo packet throughput is available.

bus configuration:	64Bit/66MHz
throughput (1500):	370 Mbit/s
throughput (3000):	-
throughput (6000):	-
throughput (9000):	-

Available products:

Name	Price
CNet ProG2000L	-
LINDY Gigabit Ethernet Karte 32bit (PCI/32)	25
LINDY Gigabit Ethernet Karte 64bit (PCI/64)	50

2.11   TMI TC9201A Based Cards

This chip was only analyzed in the c't newspaper (see [9]). This card seems not to work with Linux. No measurements are available.

bus configuration:	32Bit/33MHz
throughput (1500):	-
throughput (3000):	-
throughput (6000):	-
throughput (9000):	-

Available products:

Name	Price
LONGSHINE LCS-8039TX (PCI/64)	49
Longshine LCS-8037TX (PCI/32)	15

2.12   Cards With Unknown Chipset

There are also some cards, which were not mentioned in any study. Thus the used MAC chip is unknown, but it is most likely that one of the chips mentioned above is used. This list can be used for further price comparison:

Available products:

Name	Price
NETGEAR GA311 (PCI/64)	21
ALLNET ALL0125 (PCI/64)	-
Q-TEC 596NG PCI Network Card (PCI/32)	35
SMC SMC9452TX (PCI/32)	19
ALLIED Telesyn AT-2930T (PCI/64)	72
ALLIED Telesyn AT-2970T (PCI/64)	530
ALLIED Telesyn AT-2970T2 (PCI/64)	1169

2.13   Conclusion

There are many different cards with many different chips available on the market. The most interesting chip is the Marvell chip. Also the Intel chips and the Broadcom chips seem to be suitable for high performance computing. But however, these studies are partially very old and the performance of the different cards is highly dependent on the driver. We should make our own tests with different cards (at least Intel, Broadcom and Marvell) to achieve a usable result for throughput and latency values.

3   Interconnection Elements

3.1   Introduction

The interconnection network technology is comparable to ethernet or fast ethernet. A switch is is used for the address oriented connection of several nodes. A so called full duplex repeater is used for shared communication without considering the node's addresses. A third possible element, the so called buffered distributer is rather the same but it may buffer one ore more packets before redistributing them. But only the active switch technology is analyzed because the other two passive approaches use shared bandwidth which is not suitable for high performance computing.

3.2   Forwarding Rate

The forwarding rate F is one of three fundamental parameters of active gigabit ethernet connection elements. It specifies the amount of packets which can pass a switch per second. This is intuitively indirect proportional to the packet size S.

$$F \sim \frac{1}{S} \tag{1}$$

To be more precise, the forwarding rate can be calculated in the following way with use of the following parameters (IEEE 802.3):

1. minimal packetsize: $S_{p,min} = 64$ (14 bytes header, 46 bytes payload, 4 bytes FCS[17])

2. 8 bytes preamble: $S_{pre} = 8$

3. interframe gap for full duplex mode: $S_{gap} = 12byte$

4. gigabit bandwidth: $B_{max} = 10^9 \ Mbit/s = \frac{10^9}{8} \ MB/s$

$$S \ = \ S_p + S_{pre} + S_{gap} \tag{2}$$

$$F_{max} \ = \ \frac{B_{max}}{S} \tag{3}$$

Thus, the theoretical maximal forwarding rate $F_{max,th}$ can be reached by using the smallest possible packetsize of 64 byte:

$$F_{max,th} \ = \ \frac{B_{max}}{S_{p,min} + S_{pre} + S_{gap}} \tag{4}$$

$$F_{max,th} \ = \ \frac{10^9}{(64 + 8 + 12) * 8} \tag{5}$$

$$F_{max,th} \ = \ 1488095,2 \tag{6}$$

As you can see the theoretical maximum forwarding rate for gigabit is around 1,488 Mio packets/s.

3.3   Bandwidth

The maximum bandwidth B_{max} is set to 1 Gb/s[18]. But the usable bandwidth B_{use} is smaller because of preamble ($S_{pre} = 8byte$), some necessary header fields ($S_h = 14bytes$), the checksum ($S_{FCS} = 4bytes$) and the inter frame gap ($S_{gap} = 12bytes$) attached to every packet. The maximum theoretical usable bandwidth depends on the packets size. The packetsize S_{pkt} is considered in the ethernet standard without the preamble and the frame gap. The size of the payload data is given as S_{pay} this size can be limited by the operating systems MTU[19] value, which is 1500 bytes for the traditional ethernet standard. Thus, the packet size of traditional ethernet is limited to 1518 bytes.

$$S_{pkt} = S_h + S_{pay} + S_{FCS} \tag{7}$$

---

[17]Frame Check Sequence
[18]4 cable pairs, 250Mb/s per cable pair
[19]Maximum Transfer Unit

The regarding usable bandwidth can be calculated in the following way:

$$B_{use} \quad = \quad \frac{S_{pay}}{S_{pkt} + S_{pre} + S_{gap}} * B_{max} \tag{8}$$

$$S_{overhead} \quad = \quad S_h + S_{FCS} + S_{pre} + S_{gap} \tag{9}$$

$$S_{overhead} \quad = \quad 14 + 4 + 8 + 12 \tag{10}$$

$$S_{overhead} \quad = \quad 38 \tag{11}$$

$$B_{use} \quad = \quad \frac{S_{pay}}{S_{pay} + S_{overhead}} * B_{max} \tag{12}$$

and some are prepared in the following table:

S_{pay}	S_{pkt}	B_{use}	utilization
46	84	547,6 Mb/s	54,76%
100	138	724,6 Mb/s	72,46%
500	538	929,4 Mb/s	92,94%
1000	1038	963,4 Mb/s	96,34%
1500	1538	975,3 Mb/s	97,53%
3000	3038	987,5 Mb/s	98,75%
6000	6038	993,7 Mb/s	99,37%
9000	9038	995,8 Mb/s	99,58%
16000	16038	997,6 Mb/s	99,76%

3.4   Latency

The latency is defined as the time needed to send one packet from host A to host B through an interconnection network. There are many parameters which influence the latency, for example the operating system and hardware on the sender and receiver side, the link cable and/or switch between them and the actual traffic on the wire. Thus the latency times cannot be easily determined. However, the latency time increases with packet size and is therefore mostly measured by sending the smallest possible frame (64 byte) from host A to host B. The fastets gigabit capable switch seems to need $6\mu s$ to forward a packet(as claimed by the vendor). The overall latency for a packet should be between 20 and $60\mu s$

3.5   Evaluation Criteria

There are only very few studies about switches publicly available in the internet. So, I decided to ask the manufacturers for details on the switch parameters. The following sections are grouped by the switch's port count because this is the most necessary parameter for cluster computing. Most of the given switch performance values are supplied by the manufacturer.

Following parameters are evaluated:

- **layer** - which functionality offers the switch (layer 2/3/4)
- **managed** - is it manageable?
- **backbone speed** - the switch backbone speed, for full duplex operation ideally: $port\_count * 2GBit/s$ (full bisectional bandwidth, non blocking)
- **forwarding rate** - packet forwarding rate in pps[20] - for full duplex operation ideally: $port\_count * 2 * 1,488\,Mio\,pps$

---

[20]packets per second

- **latency** - latency measured in μs (ws means wire speed and is said to mean less than 1 μs)

- **maximal framesize** - jumbo frame support? (frames >1500 kB)

- **price** - price

Some switches offer stacking, but mostly the bandwidth is reduced by this operation, so stacking will be not considered here.

3.6   24 Port Switches

Many manufacturers offer these comparatively small switches as a single chip solution, non blocking with full bisectional bandwidth for a reasonable price. The ideal forwarding and throughput values are listed below, the actual values of different switches can be found in table 1 on page 14.

ideal forwarding rate	71,424 Mio pps
ideal backbone speed	48 Gbit/s

3.7   48 Up To 100 Port Switches

These bigger switches are harder to produce because there is no single chip solution available. An overview of today's switches which offer 24 up to 100 ports can be found in table 2 on page 15. The ideal forwarding rate and bandwidth for 48 ports is listed below:

ideal forwarding rate	142,848 Mio pps
ideal backbone speed	96 Gbit/s

3.8   More Than 100 Port Switches

Only some manufacturers offer these very big switches, they are mainly used for supercomputing and clustering. A list of different switches with more than 100 ports can be found in table 3 on page 15. The ideal forwarding rate and bandwidth for 120 ports is listed below:

ideal forwarding rate	357,12 Mio pps
ideal backbone speed	240 Gbit/s

4   Summary and Conclusion

The currently available solutions for gigabit ethernet differ a lot in terms of bandwidth and latency values. Especially switches, but also differennt network interface cards have a very low performance and are not suitable for the high demands of cluster computing. Before purchasing a gigabit ethernet solution for a cluster, one should do a comparative study. Available studies can be used but the data should be proven on different conditions (regarding architecture and topology).

product	layer	manage	Gbit/s	Mio pps	μs	framesize	EUR
SMC8624T	-	yes	88	-	-	-	-
SMC8524T	-	no	-	-	-	9216	-
LevelOne GSW-2451T	-	yes	48	1,49	-	1500	645
LevelOne GSW-2450T	2	no	48	1,49	-	1500	600
LinkSys SR2024	2	-	-	-	-	-	-
Corega COR-GSW-24	-	no	-	-	-	-	580
Dell Power Connect 2624	2	no	48	35,7	-	-	699
Dell Power Connect 5224	2/3	yes	48	35,7	-	-	1499
AllNet ALL4706W	-	-	-	-	-	-	-
AllNet ALL4724	-	-	-	-	-	-	-
NetGear GS524T	2	no	20	1,48	30	1500	922
NetGear GSM7324	2/3	yes	40	1,48	20	9216	3689
NetGear GSM7224	2	yes	48	1,48	20	9216	1801
D-Link DGS-3224TGR	2/3/4	yes	48	-	10	9216	-
D-Link DGS-3324SR	2/3	yes	88	32	ws[21]	9216	-
D-Link DGS-3324SRI	2/3	yes	160	35,7	ws	9216	-
Nortel Bay Stack 380	-	-	48	-	-	9216	-
AT-GS924i-50	2	no	-	1,48	-	-	-
Cisco Catalyst 3750G - 24T	2/3	yes	35,7	38,7	-	9018	-
Cisco Catalyst 3750G - 24TS	-	yes	38,7	38,7	-	9018	-
Foundry EdgeIron 24G	2/3	yes	48	35,7	10	9216	-
3COM SuperStack 3 Switch 4924	2/3	yes	56	41,6	ws	1500	-
3COM SuperStack 3 Switch 3824	2	yes	48	35,7	ws	1500	-
3COM Baseline Switch 2824	2	no	-	-	-	9192	-
Extreme Summit7i (!28 Port!)	2/3	yes	48	64	-	-	-
Extreme Alpine 3802 (!20 Port!)	-	-	16	12	-	9192	-

Table 1: 24 Port switches

product	ports	layer	manage	Gbit/s	Mio pps	μs	framesize	EUR
D-Link DES-6000	48	2/3	yes	21,3	-	10	-	-
D-Link DES-6300	72	2/3	yes	32	-	10	-	-
D-Link DES-6500	96	2/3/4	yes	160	-	10	9216	-
Nortel Bay Stack 5510	48	2/3	yes	160	-	-	-	-
AT-SB4104	96	2/3/4	-	64	-	-	-	-
Cisco Catalyst 6503 chassis	48	-	yes	20	30	-	-	-
Cisco Catalyst 4503 Chassis	96	-	yes	28	21	-	-	-
Foundry FastIron 3208RGC	48	2/3	yes	128	48	6	-	-
Foundry BigIron 4000	56	2/3	yes	128	101	6	-	-
Foundry FastIron 400	56	2/3	yes	128	101	6	-	-
Extreme Summit 400-48t	48	2/3	yes	160	101	-	-	-
Extreme Alpine 3804	64	-	-	32	24	-	9216	-
Extreme BlackDiamond 6804	96	-	yes	256	48	-	9216	-

Table 2: 48 - 100 Port switches

product	ports	layer	manage	Gbit/s	Mio pps	μs	framesize	EUR
AT-SB4108	192	2/3/4	-	128	-	-	-	-
Cisco Catalyst 6506 chassis	120	-	-	20	30	-	9216	-
Cisco Catalyst 6509 chassis	192	-	-	20	30	-	9216	-
Cisco Catalyst 4506 chassis	240	-	-	64	48	-	-	-
Cisco Catalyst 4507R chassis	240	-	-	68	51	-	-	-
Cisco Catalyst 6513 chassis	288	-	-	20	30	-	-	-
Cisco Catalyst 4510R Chassis	384	-	-	96	72	-	-	-
Foundry FastIron 800	120	2/3	yes	256	220	6	-	-
Foundry BigIron 8000	120	2/3	yes	256	220	6	-	-
Foundry FastIron 1500	232	2/3	yes	480	429	6	-	-
Foundry BigIron 15000	232	2/3	yes	480	429	6	-	-
Foundry BigIron MG8	480	2/3	yes	1280	480	7	9192	-
Extreme Alpine 3808	128	-	yes	64	48	-	9192	-
Extreme BlackDiamond 6808	168	-	yes	384	96	-	9192	-
Extreme BlackDiamond 6816	360	-	yes	768	192	-	9192	-
Extreme BlackDiamond 10808	480	2/3	yes	1600	-	9	9216	-
3COM Switch 7700	120	2	yes	96	-	-	9192	-

Table 3: > 100 Port switches

References

[1] IEEE COMPUTER SOCIETY: *802.3 IEEE Standard for Information technology, Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*

[2] CHRISTIAN KURMANN, FELIX RAUCH AND THOMAS M. STRICKER: *Speculative Defragmentation Leading Gigabit Ethernet to True Zero-Copy Communication*

[3] PCI-SIG: *http://www.pci-sig.com*

[4] INTEL: *Communication Streaming Architecture*

[5] ANTHONY BETZ, PAUL GRAY: *02.04.2002 - Gigabit Over Copper Evaluation*

[6] EVGENIY ZAITSEV: *02.02.2003 - Gigabit Network Adapters on 64bit PCI Bus Platform (AMD760MPX)*

[7] EVGENIY ZAITSEV: *12.05.2004 - Gigabit Network Adapters on Platform TYAN Trinity GC-SL. Part One: 32bit PCI Interface*

[8] EVGENIY ZAITSEV: *32Bit 1Gbit Network Adapters Roundup*

[9] ERNST AHLERS: *01.04.2004 - c't 04/2004 - Hochdruck Anschluss, Gigabit Ethernet Karten zum Nachruesten*