MARK A. LAUER (Bar No. 163756)
THOMAS W. LATHRAM (Bar No. 59639)
T. LESTER WALLACE (Bar No. 159967)
SILICON EDGE LAW GROUP, LLP
6601 Koll Center Parkway
Suite 245
Pleasanton, California 94566
Telephone: 925-621-2110
Facsimile: 925-621-2119

Attorneys for Plaintiff
Alacritech, Inc.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

ALACRITECH, INC., Plaintiff, v. MICROSOFT CORPORATION, Defendant.	Case No.: C04 03284 JSW **DECLARATION OF DR. KEVIN ALMEROTH IN SUPPORT OF ALACRITECH'S MOTION FOR PRELIMINARY INJUNCTION OF MICROSOFT'S INFRINGEMENT OF CLAIM 1 OF U.S. PATENT 6,697,868** The Honorable Jeffrey S. White

I, Dr. Kevin Almeroth, declare and state as follows:

Exhibit 16
Almeroth
(WITNESS) (DATE)
5.4.18
Holly Thuman, CSR 6834

Qualifications

1. I am over eighteen years old and am fully competent to make this declaration. I make this declaration in support of Alacritech's Motion For Preliminary Injunction of Microsoft's Infringement Of Claim 1 Of U.S. Patent No. 6,697,868. Except as indicated

herein, the facts stated herein are stated of my own personal knowledge and I could and would competently testify as to such facts if called upon to do so.

2. I have a Ph.D. in Computer Science from the Georgia Institute of Technology.

3. I am currently the Vice Chair and an Associate Professor in the Department of Computer Science at the University of California in Santa Barbara. My main professional research interests include computer networks and protocols, including the TCP and IP protocols.

4. My employment history and field of expertise are described in my curriculum vitae attached as Exhibit A to this declaration.

Publications

5. A list of the publications authored by me is attached to my curriculum vitae.

Consultation Fee

6. My fee for legal consultation in this case is $400.00/hour. My compensation is not conditioned on the outcome of this case.

Testimony And Deposition In Other Recent Legal Actions

7. I have provided assistance as an expert in the following cases: *ACTV v. Disney (00-CV-9622 SDNY)*; *Toddlerwatch.com v. Motorola (01-12187-REK Mass)*; *Bond Holders of @Home v. AT&T (02-5442-CRB NDCA)*; and *Two-Way Media v. AOL (C-04-089 SDTX)*. In *ACTV v. Disney*, I prepared an expert report, had my testimony taken in a deposition, and testified at a Markman Hearing in October 2001. In *Toddlerwatch.com v. Motorola*, I prepared an expert report in February 2003. In *Bond Holders of @Home v. AT&T*, I commented on claim construction issues in September 2003.

Patent Materials Reviewed

8. I have reviewed patent documents including the following:

a. U.S. Patent No. 6,697,868, and its file wrapper (the file wrapper of U.S. patent application serial number 10/208,093).

b. The CD Appendix of U.S. Patent No. 6,697,868 (CD Appendix A, CD Appendix B and CD Appendix C).

c. U.S. Patent No. 6,427,171, and its file wrapper (the file wrapper of U.S. patent application serial number 09/514,425).

d. U.S. Patent No. 6,389,479, and its file wrapper (the file wrapper of U.S. patent application serial number 09/141,713).

e. Provisional Patent Application No. 60/098,296.

f. U.S. Patent No. 6,226,680, and its file wrapper (the file wrapper of U.S. patent application serial number 09/067,544).

g. Provisional Patent Application No. 60/061,809.

h. U.S. Patent No. 6,434,620, and its file wrapper (the file wrapper of U.S. patent application serial number 09/384,792).

i. The CD Appendix of U.S. Patent No. 6,434,620 (CD Appendix A, CD Appendix B, CD Appendix C, and CD Appendix D).

j. U.S. Patent No. 6,427,173, and its file wrapper (the file wrapper of U.S. patent application serial number 09/464,283).

k. U.S. Patent No. 6,247,060, and its file wrapper (the file wrapper of U.S. patent application serial number 09/439,603).

Exhibits To This Declaration

9. Below is a list of the Exhibits to this declaration:

Exhibit A – Curriculum vitae of Dr. Kevin C. Almeroth.

Exhibit B - Compact Disc labeled "Microsoft WinHEC 2004 Products & Tools DVD 3 – Microsoft Windows Code Name Longhorn ISO Images."

Exhibit C - Photocopy of the front side of the Compact Disc of Exhibit B.

Exhibit D - Compact Disc labeled "Microsoft WinHEC 2004 Products & Tools DVD 2 – Microsoft Windows Code Name Longhorn ISO Images."

Exhibit E - Photocopy of the front side of the Compact Disc of Exhibit D.

Exhibit F – Copy of the "readme" file, titled "Readme for the Preliminary Release of Microsoft Windows Code Name "Longhorn" ," from the compact disc of Exhibit D.

Exhibit G – Compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD1".

Exhibit H - Photocopy of the front side of the compact disc of Exhibit G.

Exhibit I – Compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD2".

Exhibit J - Photocopy of the front side of the compact disc of Exhibit I.

Exhibit K – Slides of the WinHEC 2004 presentation entitled "Windows Architecture And Roadmap For Scalable Networking" (submitted as Exhibits K1 and K2).

Exhibit L – Slides of the WinHEC 2004 presentation entitled "Designing Quality Advanced Ethernet Adapters And Drivers."

Exhibit M – Slides of the WinHEC 2004 presentation entitled "Writing NDIS Drivers For TCP Offload Engine NICs."

Exhibit N – White paper entitled "Microsoft Windows Scalable Networking Initiative – WinHEC 2004 Version – April 13, 2004."

Exhibit O - White paper entitled "Scalable Networking: Network Protocol Offload – Introducing TCP Chimney."

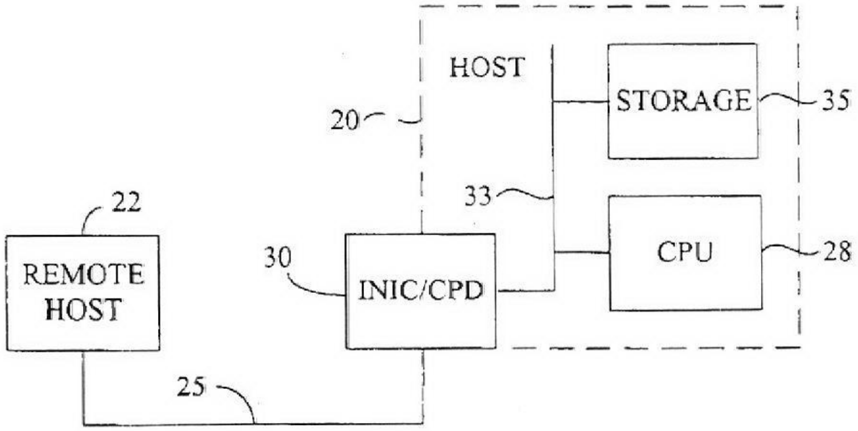
Exhibit P - TCP specification (RFC793).

Exhibit Q - Figure 24.15 of the book entitled "TCP/IP Illustrated," Volume 2 (7th edition, 1999), by Gary R. Wright and W. Richard Stevens, that is incorporated by reference into the '868 patent.

Exhibit R – Documentation entitled "Full TCP Offload" found in the Longhorn Development Kit.

U.S. Patent No. 6,697,868

10. U.S. Patent No. 6,697,868 (the '868 Patent) discloses and teaches in Figure 1 and the corresponding text a host computer 20. Figure 1 is replicated below.



As illustrated in Figure 1, host computer 20 is coupled to a remote host 22 by a network 25. The host computer 20 includes a central processing unit (CPU) 28, storage 35, and an intelligent network interface card (INIC) 30.

11. In the networking and computer arts to which the '868 patent pertains, a network interface card is often called a "NIC".

12. As set forth in the '868 patent in places such as column 6, lines 1-2, the INIC/CPD 30 appearing in Figure 1 of the '868 patent has an ability to perform TCP protocol processing functions. The INIC/CPD 30 appearing in Figure 1 of the '868 patent is therefore

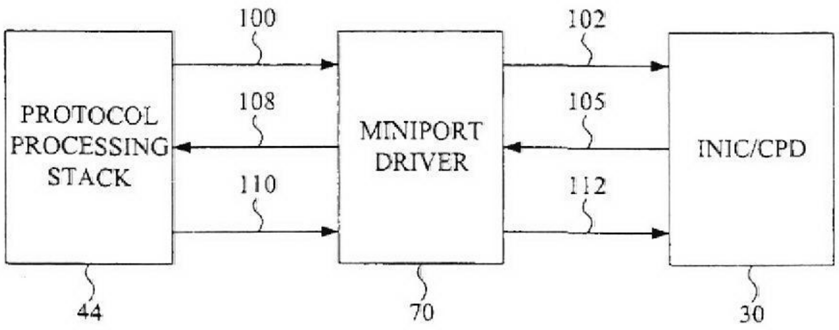
a type of NIC, and because it can perform TCP protocol processing it is called an "intelligent" network interface card, or "INIC."

13. To install a NIC on a host computer, a piece of software commonly referred to as a "driver" is loaded onto the host computer. CD Appendix C of the '868 patent contains a copy of source code for such a "driver".

14. The driver executes on the CPU of the host computer. The host computer also typically runs operating system software. The operating system typically includes software called a "protocol processing stack." CD Appendix C of the '868 patent contains a copy of source code for such a protocol processing stack.

15. The driver provides an interface between the NIC card and the protocol processing stack of the operating system of the computer.

16. Figure 4 of the '868 patent illustrates protocol processing stack software 44 as well as driver software 70. Figure 4 is replicated below.



As illustrated in Figure 4, the driver software 70 provides an interface between the INIC 30 and the protocol processing stack 44. Both the driver software 70 and the protocol processing stack 44 are programs, or sets of instructions, that are executed by the CPU 28 of host computer 20. Column 3, lines 23-24 of the '868 patent, for example, states that CPU 28 runs the protocol processing stack 44 of instructions. Column 8, lines 61-62 of the '868 patent, for example, states that driver 70 is installed on host computer 20.

17. When a host computer communicates in conventional fashion with a remote host using the TCP and IP protocols, the CPU of the host computer performs a substantial amount of protocol processing on the information. As set forth in the background section of the '868 patent, a large portion of the CPU's computing resources may be consumed by such protocol processing, and this may interfere with the CPU's ability to perform other tasks.

18. The '868 patent discloses and teaches the offloading of time consuming protocol processing tasks from the CPU of a host computer to an intelligent network interface card (INIC).

19. The '868 patent discloses and teaches a "set of instructions" that are executable on the CPU 28 of host computer 20. This "set of instructions" includes protocol processing stack 44.

20. The '868 patent discloses and teaches that the set of instructions are executed by CPU 28 such that a TCP connection is established between host computer 20 and remote host 22. Once the TCP connection has been established, then messages can be communicated via the TCP connection between host computer 20 and remote host 22.

21. The '868 patent discloses in column 9, lines 50-55, that "the protocol processing stack 44 on the host has responsibility for deciding when a connection is to be handed out to the INIC/CPD 30. A connection can be handed out to the INIC/CPD 30 as soon as the connection is fully ***established***." (emphasis added).

22. CD Appendix C of the '868 patent includes a set of instructions (in source code format) for a protocol processing stack that runs on a host computer. The set of instructions includes instructions for putting a TCP connection into the ***established*** state. (emphasis added).

23. The '868 patent discloses and teaches that the set of instructions, once the TCP connection is established, can cause the TCP connection to be "offloaded" from the CPU 28.

This offloading allows some of the most time consuming protocol processing associated with the TCP connection to be performed by the INIC 22 rather than the CPU 28 of the host computer 20.

24. “Offloading” a TCP connection is also referred to as “handing out” a TCP connection or “connection handout”. The ‘868 patent explains in column 5, lines 60-62 that a Communication Control Block or “CCB” “contains the set of variables used to represent the state of a given TCP connection.” The ‘868 patent in lines 47-48 explains that a CCB defines a particular TCP connection. The ‘868 patent then explains in column 5, lines 62-65 that “transfer of a CCB from the host to the INIC/CPD is termed a connection handout...”

25. The ‘868 patent discloses in column 6, lines 1-2, that “once a connection handout occurs, the INIC/CPD handles all TCP processing...”. The ‘868 patent discloses in column 5, lines 35-38, that “...the present invention improves system performance by offloading TCP/IP data processing from the host protocol stack to the INIC/CPD.”

26. The set of protocol processing stack instructions (in source code format) on CD Appendix C includes instructions for offloading a TCP connection from a host computer to an INIC.

Microsoft’s “TCP Chimney” Software
Distributed at the WinHEC 2004 Conference

27. Attached as Exhibit B is a first compact disc that contains “a set of instructions” made by Microsoft.

28. Attached as Exhibit C is a photocopy of the face of the compact disc of Exhibit B. The disc is labeled “Microsoft WinHEC 2004 Products & Tools DVD 3 – Microsoft Windows Code Name Longhorn ISO Images.”

29. Attached as Exhibit D is a second compact disc that was distributed along with the first compact disc of Exhibit B during WinHEC 2004.

30. Attached as Exhibit E is a photocopy of the face of the compact disc of Exhibit D. The disc is labeled "Microsoft WinHEC 2004 Products & Tools DVD 2 – Microsoft Windows Code Name Longhorn ISO Images."

31. The compact discs of Exhibits B and D were received from Microsoft during a conference called "WinHEC 2004". WinHEC stands for "Windows Hardware Engineering Conference". The WinHEC 2004 conference was hosted by Microsoft and took place in Redmond, WA in May, 2004.

32. I am informed that compact discs identical to the compact discs of Exhibits B and D were made by Microsoft and were then distributed by Microsoft during the WinHEC 2004 conference to numerous attendees of the conference.

33. A "readme" file on the compact disc of Exhibit D is titled "Readme for the Preliminary Release of Microsoft Windows Code Name "Longhorn." A copy of this "readme" file is attached as Exhibit F. After the title, a caption reads "Distributed at the Windows Hardware Engineering Conference May 2004". The "readme" file explains how to install the Longhorn operating system on "x86-based systems", "Itanium-based systems" and "64-bit extended systems".

34. The "set of instructions" that is found on the compact disc of Exhibit B infringes Claim 1 of the '868 patent. The "set of instructions" is a Microsoft operating system code named "Longhorn".

35. The "set of instructions" is found in the form of a file named "lh_usa_4074_x86_pro-dvd.iso." This file is an iso image file usable to create an installation compact disc. When the resulting installation compact disc is loaded onto an x86-based computer, a set-up program automatically executes. A sequence of pop-up screens is then displayed on the computer. These pop-up screens lead the user through a sequence of steps to load the Longhorn operating system onto the x86-based computer. When installed on an x86-based system, the x86 processor of the x86-based computer executes the Longhorn "set of instructions".

36. A Pentium microprocessor is an example of an x86 processor. An instruction set defines the various instructions that an x86 processor can execute. The instructions of the Longhorn "set of instructions" are instructions of the x86 instruction set.

37. The version of the Longhorn "set of instructions" is identified by a build number. The build number of the Longhorn operating system "set of instructions" derived from the iso file is build 4074. I therefore refer to this version as the Longhorn build 4074 "set of instructions".

38. The Longhorn build 4074 "set of instructions" includes a subset of instructions that implements a TCP offload capability. Microsoft calls the TCP offload capability "TCP Chimney". In this declaration, I refer to the instructions that implement TCP Chimney as "the TCP Chimney software".

39. At the WinHEC 2004 conference, several oral presentations were made that describe the "TCP Chimney software" and how it operates. Those presentations include:

a. A presentation entitled "Windows Architecture And Roadmap For Scalable Networking".

b. A presentation entitled "Designing Quality Advanced Ethernet Adapters And Drivers".

c. A presentation entitled "Writing NDIS Drivers For TCP Offload Engine NICs".

40. I ordered from Microsoft and received in the mail compact discs including two discs entitled "Microsoft WinHEC 2004 – Event In a Box – DVD1" and "Microsoft WinHEC 2004 – Event In a Box – DVD2", respectively. These compact discs contain video (in the form of files of digital information) of the three presentations mentioned above. The compact discs also contain electronic copies of overhead slides that were presented at the conference contemporaneously with the presentations. The electronic copies are in the form of PowerPoint files (.ppt files). In addition to the two compact discs, the "Event In a Box" included conference proceedings.

41. Attached as Exhibit G is the first compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD1".

42. Attached as Exhibit H is a photocopy of the front side of the first compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD1".

43. Attached as Exhibit I is the second compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD2".

44. Attached as Exhibit J is a photocopy of the front side of the second compact disc entitled "Microsoft WinHEC 2004 – Event In a Box – DVD2".

45. Video of the presentation entitled "Windows Architecture And Roadmap For Scalable Networking" is found on the first compact disc labeled "DVD1." The video can be viewed by clicking on the file "TW04074.htm." The slides presented with the presentation are found in the file "TW04074.ppt." A paper copy of the slides is attached as Exhibit K.

46. Video of the presentation entitled "Designing Quality Advanced Ethernet Adapters And Drivers" is found on the first compact disc labeled "DVD1." The video can be viewed by clicking on the file "TW04012.htm." The slides presented with the presentation are found in the file "TW04012.ppt." A paper copy of the slides is attached as Exhibit L.

47. Video of the presentation entitled "Writing NDIS Drivers For TCP Offload Engine NICs" is found on the first compact disc labeled "DVD1." The video can be viewed by clicking on the file "TW04086.htm." The slides presented with the presentation are found in the file "TW04086.ppt." A paper copy of the slides is attached as Exhibit M.

48. Slide 37 of the presentation "Windows Architecture And Roadmap For Scalable Networking" of Exhibit K directs the viewer and attendees of the WinHEC 2004 conference to the following white papers:

a. "Microsoft Windows Scalable Networking Initiative"; and

b. "Scalable Networking: Network Protocol Offload – Introducing TCP Chimney".

49. A copy of the white paper entitled "Microsoft Windows Scalable Networking Initiative – WinHEC 2004 Version – April 13, 2004" is attached as Exhibit N. The face of

the white paper states "The current version of this paper is maintained on the Web at: http://www.microsoft.com/whdc/".

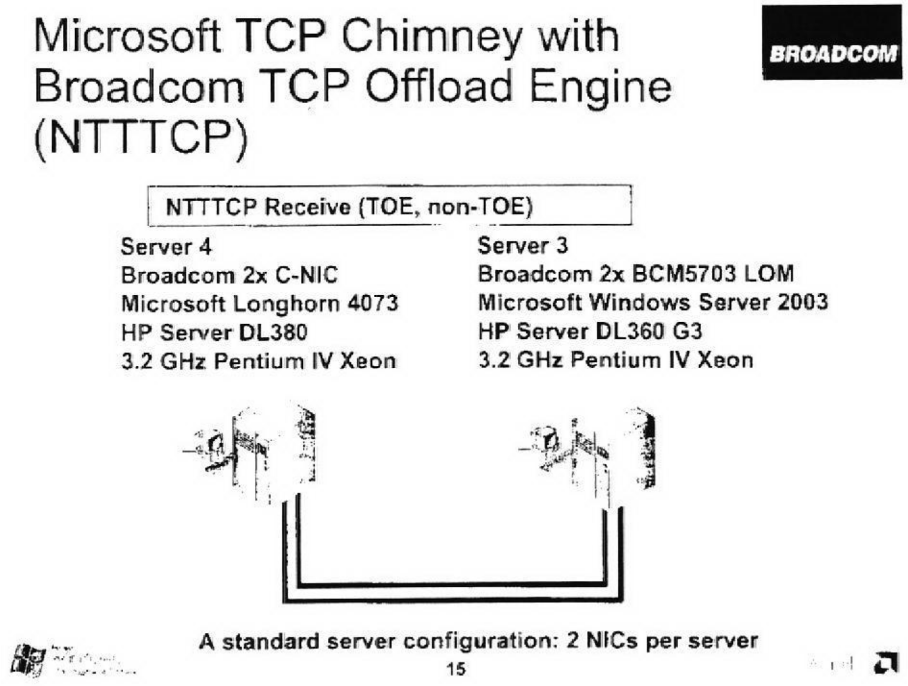
50. A copy of the white paper entitled "Scalable Networking: Network Protocol Offload – Introducing TCP Chimney" is attached as Exhibit O. The face of the white paper states "The current version of this paper is maintained on the Web at: http://www.microsoft.com/whdc/".

51. I have reviewed the video of the presentations of Exhibits G and I, the slides of Exhibits K, L and M, and the white papers of Exhibits N and O. The video, slides and white papers constitute evidence that the "TCP Chimney software" that is part of Longhorn operating system build 4074 that was distributed to attendees of the WinHEC 2004 conference embodies each and every recitation of Claim 1 of the '868 patent and therefore infringes Claim 1 of the '868 patent. A detailed explanation of where each recitation of Claim 1 is found in the infringing software is set forth below in the section entitled "Infringement of Claim 1 of the '868 Patent."

Demonstration of "TCP Chimney" Software at the WinHEC 2004 Conference

52. A live technical demonstration of the operation of a version of the Longhorn operating system, Longhorn build 4073, was made at the WinHEC 2004 conference. Video of the demonstration is found between time 25:45 and time 41:25 when the TW04074.htm video file on the compact disc of Exhibit G is viewed. Slides presented along with the demonstration are slides 13-19 of the presentation of Exhibit K.

53. The demonstration was performed on a system illustrated on slide 15 of Exhibit K. Slide 15 is replicated below.



Microsoft TCP Chimney with Broadcom TCP Offload Engine (NTTTCP)

BROADCOM

NTTTCP Receive (TOE, non-TOE)

Server 4
Broadcom 2x C-NIC
Microsoft Longhorn 4073
HP Server DL380
3.2 GHz Pentium IV Xeon

Server 3
Broadcom 2x BCM5703 LOM
Microsoft Windows Server 2003
HP Server DL360 G3
3.2 GHz Pentium IV Xeon

A standard server configuration: 2 NICs per server

15

As illustrated in slide 15, the system involved a first computer (server 4) and a second computer (server 3). Server 4 was an HP Server DL380. The processor of server 4 was a Pentium IV Xeon. Microsoft Longhorn operating system build 4073 was executing on the processor of server 4. The "TCP Chimney software," which is part of the Longhorn operating system build 4073, was being executed by the Pentium processor of server 4 as part of the execution of the remainder of the Longhorn operating system. At time 28:39 of the video of Exhibit G, the statement is made "we're running TCP Chimney". Server 4 included a Broadcom C-NIC. The C-NIC is, as the title of slide 15 indicates, a "TCP Offload Engine".

Server 3 was an HP Server DL360 G3. During the demonstration, a set of instructions executing on the Pentium processor of server 4 caused a TCP connection to be established. The TCP connection was between server 4 and server 3. Once the TCP connection had been established, the TCP connection was "offloaded" such that the C-NIC of server 4 was performing protocol processing tasks. At time 35:40 of the video, the statement is made that "the connection we're using is actually offloaded". At time 36:00 of the video, the statement is made that "the connection that is being used on the receiving side, the side that is retrieving the files, is actually an ***offloaded connection***" (emphasis added).

54. Slide 16 entitled "TCP Offload Engine (TOE) Preliminary User Mode Performance" contains performance data for the demonstration. This performance data is evidence that protocol processing tasks were offloaded from the "CPU" of server 4 to a C-NIC of server 4.

55. The C-NIC of server 4 is part of an "intelligent TCP offload mechanism".

56. During the demonstration, Longhorn build 4073 executed on a processor, caused a TCP connection to be established, and then offloaded the TCP connection from the processor to an intelligent TCP offload mechanism.

Interpretation of Terms in Claim 1 Of The '868 Patent

57. I interpret Claim 1 of the '868 patent and terms that appear in Claim 1 of the '868 patent as one of ordinary skill in the art (at the time the invention was made) would have interpreted the claim and the terms after reviewing: the text and Figures of the '868 patent (including the CD appendix of the '868 patent and the documents incorporated by reference into the '868 patent), the other claims in the '868 patent, the prosecution history of the '868 patent, and the patent documents that are listed in the "Cross Reference To Related Applications" section of the '868 patent.

58. The phrase "set of instructions executable on a processor" in Claim 1 of the '868 patent means "code such as software that can be run by a processor". A Pentium microprocessor is an example of a "processor" as the term "processor" is used in Claim 1 of the '868 patent. An example of a "set of instructions" in the '868 patent is an operating system that includes a protocol processing stack.

59. The term "TCP connection" in Claim 1 of the '868 patent means "a combination of information that identifies a process on a local host and a process on a remote host that wish to communicate using TCP, describes the status of TCP communication between those

processes, and can be employed to send data between those processes using TCP". Support for this interpretation is found in numerous places including the TCP specification (RFC793) itself. A copy of the TCP specification (RFC793) entitled "TCP standard (rfc793), Transmission Control Protocol" is attached as Exhibit P. Page 5 of the TCP specification, lines 15-24, defines "Connections: The reliability and flow control mechanisms described above require that TCPs initialize and maintain certain status information for each data stream. ***The combination of this information***, including sockets, sequence numbers, and window sizes, ***is called a connection***…When two processes wish to communicate, their TCP's must first establish a connection (initialize the status information on each side)" (emphasis added).

60. The phrase "establishing a TCP connection" in Claim 1 of the '868 patent means "putting a TCP connection into the "ESTABLISHED" state". The ESTABLISHED state is a state of a TCP connection. The ESTABLISHED state is defined by the TCP specification (RFC793).

61. The phrase "the TCP connection being at least in part identified by a TCP source port, TCP destination port, IP source address, and IP destination address" in Claim 1 of the '868 patent is part of the definition of a "TCP connection." The words "being at least in part identified by" are a definitional recitation for the preceding term "TCP connection" within the context of the claim. A particular "TCP connection" can be uniquely identified by its TCP source and destination ports, and its IP source and destination addresses. This is being spelled out by the "being at least in part identified by" phrase in Claim 1.

62. The phrase "offloading the TCP connection from the processor to an intelligent TCP offload mechanism" in Claim 1 of the '868 patent means "transferring the TCP connection from the processor to an intelligent TCP offload mechanism". An example in the '868 patent of "offloading the TCP connection from the processor to an intelligent TCP offload mechanism" is the passing of a CCB from stack 44 to driver 70 during the connection

handout process as set forth in the '868 patent, col. 9, lines 23-26. The "CCB" (Communication Control Block) includes at least in part the TCP source port, the TCP destination port, the IP source address, and the IP destination address as set forth in the '868 patent, col. 6, lines 7-10. As set forth in col. 9, lines 25-30, when the driver receives the CCB from the stack 44, the driver 70 in turn forwards the CCB on to the INIC 30.

63. The term "intelligent TCP offload mechanism" in Claim 1 of the '868 patent means "a network interface for the processor, which is capable of processing the established TCP connection". An example in the '868 Patent of an "intelligent TCP offload mechanism" is the combination of INIC 30 and driver software 70.

64. Claim 2 depends from Claim 1. Claim 2 recites that "the TCP offload mechanism" of Claim 1 "is a network interface card (NIC)". Claim 1 has a scope that is broader than, and also encompasses, the scope of Claim 2. I interpret the term "intelligent TCP offload mechanism" in Claim 1 consistent with the specification of the '868 patent to encompass a combination of an intelligent NIC and its driver software, whereas I interpret Claim 2 as more specifically defining that the intelligent TCP offload mechanism is a "network interface card (NIC)" without its associated software driver. Because the scope of Claim 1 encompasses the scope of Claim 2, a "network interface card (NIC)" alone would also satisfy the meaning of "intelligent TCP offload mechanism" for purposes of Claim 1.

Prosecution History of Claim 1 Of The '868 Patent

65. Claim 1 of the '868 patent was originally filed as Claim 1 of U.S. Patent Application Serial No. 10/208,093 on July 29, 2002. The claim as filed was exactly the same as Claim 1 in the issued '868 patent, but for the insertion of the term "intelligent" before the term "TCP offload mechanism." The term "TCP offload mechanism" at the end of the claim was therefore changed to "intelligent TCP offload mechanism." No office action rejecting any claim was ever made by the Examiner, rather the first action from the Examiner was a

"Notice of Allowance" and "Notice of Allowability" dated August 1, 2003. In the "Notice of Allowability", the Examiner made an "Examiner's Amendment" that changed the term "TCP offload mechanism" in Claim 1 to "intelligent TCP offload mechanism". The prosecution history provides no explicit written reason for why the Examiner's Amendment was made. The application, including Claim 1 as amended by the Examiner, then issued as the '868 patent on February 24, 2004.

66. I interpret the terms of Claim 1 of the '868 patent so that the meaning of each term is consistent with the prosecution history of the '868 patent. The '868 patent claims priority from U.S. Patent Application Serial No. 09/384,792 (now U.S. Patent No. 6,434,620). As set forth in the "Cross Reference To Related Applications" section of the '868 patent, the subject matter of U.S. Patent Application Serial No. 09/384,792 (now U.S. Patent No. 6,434,620) is incorporated by reference into the '868 patent. The four inventors listed on the face of the '868 patent are also listed as inventors on U.S. Patent Application Serial No. 09/384,792 (now U.S. Patent No. 6,434,620). In the text of U.S. Patent Application Serial No. 09/384,792, the inventors repeatedly refer to a conventional network interface card (conventional NIC) as a "dumb NIC", and thereby distinguish their "intelligent" NIC (INIC) set forth in U.S. Patent Application Serial No. 09/384,792 from conventional "dumb" NICs. I note the following five occurrences of the term "dumb" NIC in U.S. Patent No. 6,434,620: 1) col. 7, line 14; 2) col. 11, line 26; 3) col. 12, line 24; 4) col. 18, line 40; and 5) col. 20, line 56.

67. To give meaning to the prosecution history, I interpret the term "intelligent" in the term "intelligent TCP offload mechanism" of Claim 1 of the '868 patent to have a meaning that distinguishes pre-existing TCP offload mechanisms involving relatively "dumb" network interface cards ("dumb" NICs) that were not capable of processing a TCP connection as the INIC set forth in the '868 patent is.

Infringement of Claim 1 Of The '868 Patent

68. Claim 1 of the '868 patent is replicated in the claim chart below. Evidence of each recitation of Claim 1 being embodied in the Longhorn software is set forth in the right hand column of the chart:

1. A set of instructions executable on a processor, the set of instructions being for performing steps comprising:	Microsoft's Longhorn operating system build 4074 is a "set of instructions". This set of instructions executes on the CPU (central processing unit) of a computer. A CPU is a "processor". For example, Longhorn build 4074 includes binary executable instructions for execution on a x86 processor. A Pentium is an x86 processor. Microsoft's Longhorn operating system 4073 is a "set of instructions". This set of instructions executes on the CPU of a computer. A CPU is a "processor". In the case of the demonstration of the operation of Longhorn build 4073 of Exhibits G and K, the processor was a "3.2 GHz Pentium IV Xeon" processor. For additional details, see paragraphs 69 and 70 below.
establishing a TCP connection, the TCP connection being at least in part identified by a TCP source port, TCP destination port, IP source address, and IP destination address; and	Execution of Longhorn build 4074 causes a "TCP connection" to be put into the "established" state. Microsoft documentation states "connection established on host" (Exh. K, slide 10). Execution of Longhorn build 4073 causes a 'TCP connection" to be put into the "established" state. In the case of the demonstration of Exhibits G and K, Longhorn build 4073 caused a TCP connection to be established. The TCP connection allowed information to move between server 3 and server 4. The phrase "at least in part identified by a TCP source port, TCP destination port, IP source address, and IP destination address" is a statement of what a "TCP connection" is for purposes of the claim. It is a definitional recitation of the term "TCP connection" for purposes of the claim. For additional details, see paragraphs 71-75 below.

offloading the TCP connection from the processor to an intelligent TCP offload mechanism.	Execution of Longhorn build 4074 causes the "TCP connection" to be offloaded from the "processor" to an "intelligent TCP offload mechanism." Execution of Longhorn build 4073 causes the "TCP connection" to be offloaded from the "processor" to an "intelligent TCP offload mechanism." In the case of the demonstration of Exhibits G and K, the "C-NIC" along with its driver constitute an "intelligent TCP offload mechanism". Slides 16 and 19 of Exhibit K presented contemporaneously with the demonstration confirm that CPU utilization (the CPU is a "processor") was reduced due to the offloading, thereby evidencing that the TCP connection was offloaded from the CPU to something that could reduce the load on the CPU, namely "an intelligent TCP offload mechanism." For additional details, see paragraphs 76-91 below.

69. Microsoft's Longhorn operating system build 4074 is a "set of instructions". A copy of the Longhorn build 4074 "set of instructions" is on the compact disc of Exhibit B. Longhorn build 4074 includes a "set of instructions" in executable binary form that are executable on a x86 CPU (central processing unit) of a computer. An x86 CPU is a "processor".

70. Microsoft's Longhorn operating system build 4073 is a "set of instructions". This set of instructions executes on the CPU of a computer. A CPU is a "processor". In the case of the demonstration of the operation of Longhorn build 4073 of Exhibits G and K, the CPU of server 4 executed the "Microsoft Longhorn 4073" set of instructions, and the CPU was a "3.2 GHz Pentium IV Xeon" processor. Slide 15 of Exhibit K is a diagram of the system on which the demonstration took place.

71. Execution of the "set of instructions" of a Longhorn operating system[1] causes two steps to occur. First, execution of the "set of instructions" causes a "TCP connection" to be

[1] Either Longhorn build 4074 or Longhorn build 4073.

"established". Second, after the TCP connection has been established, execution of the "set of instructions" causes the TCP connection to be "offloaded."

72. Evidence that execution of the Longhorn "set of instructions" causes a TCP connection to be established before the TCP connection is offloaded appears in numerous places in Microsoft documentation and presentations. For example, page 1 of the Microsoft white paper entitled "Microsoft Windows Scalable Networking Initiative" (Exhibit N) states that the information in the paper "applies to the Microsoft Windows operating system codenamed "Longhorn." " Page 8 of the document contains an "Overview of the TCP Chimney". In this "Overview of the TCP Chimney" section, the document states "TCP Chimney offload can occur on any ***existing*** TCP connection that is in the ***ESTABLISHED*** state..." (emphasis added). Due to the reference to an "existing" TCP connection, it is clear that the TCP connection being referred to was established before it was offloaded. The document states further down on page 8 that "TCP connection setup,...., is still done by the host stack". TCP connection setup here means putting the TCP connection into the ESTABLISHED state. The "ESTABLISHED" state is a state defined by the TCP protocol as described in RFC793. Consistent with the "Overview of the TCP Chimney" section of the document of Exhibit N, the presentation entitled "Windows Architecture And Roadmap For Scalable Networking" (Exhibit K) makes it clear that TCP connections are not "established" by the offload target, but rather are established by the host. At time 11:09 of the video of the corresponding presentation (viewable by clicking on the file TW04074.htm on the compact disc of Exhibit G), it is stated that "TCP Chimney does not offload connection setup." Slide 9 of the corresponding presentation slide of Exhibit K states "Connection setup, ...on host...". Similarly, the next slide, slide 10, reads "Chimney initialized after connection established on host". Consistent with this, the presentation entitled "Writing NDIS Drivers For TCP Offload Engine NICs" indicates at time 27:43 that "connection offload will have to wait until the connection goes to the established state". This video can be viewed by clicking on the file TW04086.htm on the compact disc of Exhibit G. Due to these many different pieces of

evidence, it is clear that the Longhorn "set of instructions" causes "a TCP connection" to be "established", and that this establishing of the TCP connection occurs before the TCP connection is "offloaded".

73. "Establishing" a TCP connection means putting the TCP connection into the ESTABLISHED state as defined by the TCP protocol.

74. The TCP protocol defines several "states". One of the states is called the "ESTABLISHED" state. The "ESTABLISHED" state of the TCP protocol is illustrated in the diagram of Exhibit Q. The diagram of Exhibit Q is labeled "TCP state transition diagram." The diagram of Exhibit Q is Figure 24.15 from page 806 of the book entitled "TCP/IP Illustrated," Volume 2 (7th edition, 1999), by Gary R. Wright and W. Richard Stevens. The Wright and Stevens book is incorporated by reference into the '868 patent as indicated by column 4, lines 10-15 of the '868 patent.

75. Claim 1 contains the recitation "the TCP connection being at least in part identified by a TCP source port, TCP destination port, IP source address, and IP destination address". This claim recitation is a statement of what a "TCP connection" is for purposes of Claim 1. It is not a recitation of an act to be performed. It is a definitional recitation for the preceding term "TCP connection" within the context of the claim.

76. In addition to establishing the TCP connection, the Longhorn operating system set of instructions of Exhibit B causes the "offloading" of the TCP connection from the processor to an intelligent TCP offload mechanism.

77. Evidence that execution of the Longhorn set of instructions causes the TCP connection to be "offloaded" appears in numerous places in the Microsoft documentation. For example, the "Overview of the TCP Chimney" section on page 8 of the white paper entitled "Microsoft Windows Scalable Networking Initiative" (Exhibit N) states that "multiple TCP connections can be ***offloaded***" (emphasis added). Page 9 of the same white paper continues on stating "TCP Chimney ***offload*** does not reduce the overhead associated with completing an I/O to a user-mode application. It simply ***offloads*** all TCP and IP

network processing to the offload target" (emphasis added). This description of offloaded connections follows the statement on page 8 that "TCP Chimney Offload can occur on any existing TCP connection that is in the ESTABLISHED state". It is therefore clear that it is an already established TCP connection that has been offloaded.

Consistent with the "Overview of the TCP Chimney" section of the white paper of Exhibit N, is the reference to offloaded connections in the slides of Exhibit K presented along with the presentation entitled "Windows Architecture And Roadmap For Scalable Networking." After the statement "Connection setup,...on host" on slide 9 (Exhibit K) appears the statement "Multiple connections can be ***offloaded***/uploaded at one time" (emphasis added). The connections referred to as "offloaded" therefore must have been connections that were previously established.

78. Claim 1 of the '868 patent requires that the TCP connection be offloaded "from the processor to an intelligent TCP offload mechanism".

79. Evidence that execution of Longhorn build 4074 causes the "TCP connection" to be offloaded "from the processor to an intelligent TCP offload mechanism" appears in numerous places in the Microsoft documentation. For example, page 4 of the white paper "Scalable Networking: Network Protocol Offload – Introducing TCP Chimney" (Exhibit O) states that "TCP Chimney ***offloads*** the TCP protocol stack ***to a Network Interface Card (NIC)***" (emphasis added).

80. A NIC that is capable of TCP processing an offloaded TCP connection, together with its driver software, is an example of an "intelligent TCP network mechanism".

81. Evidence that execution of Longhorn build 4074 causes the "TCP connection" to be offloaded "from the processor to an intelligent TCP offload mechanism" is present in the form of multiple statements in Microsoft presentations and documentation that Longhorn is to operate with TOE NICs. "TOE" stands for "TCP Offload Engine". A "TOE" device is a hardware device that can offload a host CPU of TCP protocol processing functions. A TOE NIC is such a device embodied on a network interface card (NIC).

82. A "TOE NIC", together with its driver, is an example of a "TCP offload mechanism" because a "TOE NIC" is capable of TCP processing an offloaded TCP connection for a host computer to which the TOE NIC is attached.

83. A Longhorn Development Kit (LDK) is found on the first compact disc of Exhibit D. "LDK" stands for Longhorn Development Kit. This kit includes a software testing program for testing the proper operation of a TOE NIC when a TOE NIC is used with Longhorn. Slide 11 of the presentation "Designing Quality Advanced Ethernet Adapters And Drivers" is entitled "Testing TCP Offload Engine NICs". Slide 14 shows a "System Under Test" that includes a "TOE" NIC. Slide 17 mentions "System Under Test – TOE NIC". Slide 19 shows a screen shot of operation of an LDK utility program. The screen shot contains a selection of "Manual Tests" that can be performed, and two of those tests are titled "TCP/IP offload engine setup" and "TCP/IP offload engine test". Slide 22 is entitled 'Test Kits – LDK". The slide then states "Please refer to LDK Kit installation guidelines". Slide 26 is entitled "Test Kits". On the slide appears a heading "LDK". Below the heading appears the statement "Tests are functional now!". A complete other presentation was given at WinHEC 2004 entitled "Writing NDIS Drivers For TCP Offload Engine NICs." Slides from this presentation are attached as Exhibit M. This presentation was giving instructions on how to write a driver program to interface a "TOE NIC" to Longhorn. Slide 3 entitled "Session Outline" contains a heading entitled "TOE (TCP Offload Engine) support". The next slide, slide 4, is entitled "Initialization: Exchanging Handlers In NDIS 6.0 ("Longhorn")." As evidenced from these many different pieces of evidence, Microsoft is teaching how to cause the Longhorn "set of instructions" to "offload" processing associated with "a TCP connection" from a CPU to a "TOE NIC". A "TOE NIC", together with its driver, is an example of a "TCP offload mechanism".

84. In addition to the software testing program, the LDK (Longhorn Development Kit) includes documentation that explains capabilities of the chimney offload architecture of the Longhorn "set of instructions". Attached as Exhibit R is a printout of a screen of

documentation found in the LDK. The LDK documentation of Exhibit R is entitled "Full TCP Offload". The LDK documentation explains that "NDIS 6.0 supports a new architecture that enables full TCP offload... This architecture is called a 'chimney offload' architecture because it provides a direct connection, called a 'chimney,' between applications and ***an offload-capable NIC***." (emphasis added) The LDK documentation of Exhibit R goes on to explain that "Depending on the offload features supported by the NIC, the ***chimney enables the NIC to perform all TCP... processing for offloaded connections***, including maintaining the protocol state." (emphasis added). In my opinion, an "offload-capable NIC" that can "perform all TCP processing for an offloaded connection" is an example of a "TCP offload mechanism". The LDK documentation of Exhibit R therefore constitutes additional evidence that the Longhorn "set of instructions" enables the offloading of a "TCP connection" from the processor to a "TCP offload mechanism."

85. Microsoft calls the combination of a TOE NIC and its associated driver an "Offload Target". Microsoft offloads a TCP connection by moving a "state structure" from the Chimney software to the driver portion of the "offload target", wherein the state structure includes the TCP source port, the TCP destination port, the IP source address, and the IP destination address. For example, Microsoft explains on page 5 of Exhibit O that initiating an offload involves "moving a host-based state structure from the host stack to the Offload Target". This state structure, as evidenced by page 7 of Exhibit O, includes the TCP source port and the TCP destination port. This state structure, as evidenced by page 9 of Exhibit O, includes the IP source address and the IP destination address.

86. Microsoft's "set of instructions" offloads a TCP connection by moving a "state structure" from the Chimney software to the driver portion of the "Offload Target". The '868 patent discloses offloading a TCP connection by moving a "CCB" from the stack 44 to the driver 70 (see '868 patent, column 9, lines 23-27). Microsoft's moving of the "state structure" is analogous to the '868 patent's moving of the "CCB". In both cases, the TCP

source port, the TCP destination port, the IP source address, and the IP destination address is being passed to the driver of an "intelligent TCP offload mechanism".

87. The demonstration of Exhibits G and K constitutes evidence that execution of Longhorn build 4073 causes a "TCP connection" to be offloaded "from the processor to an intelligent TCP offload mechanism." Slides 16 and 19 (Exhibit K) presented contemporaneously with the demonstration of Exhibits G and K confirm that "CPU utilization" (processor utilization) of the host was reduced due to the offloading, thereby evidencing that the TCP connection was offloaded to something that could reduce the load on the CPU, namely "an intelligent TCP offload mechanism."

88. Because "CPU utilization" was reduced in the demonstration of Exhibits G and K due to the offloading, the TCP connection could not simply have been offloaded to driver software executing on the host, but rather had to have been offloaded to the Broadcom "C-NIC".

89. Slide 15 of the "Windows Architecture And Roadmap For Scalable Networking" (Exhibit K) refers to the Broadcom "C-NIC" of the demonstration as a "Broadcom TCP Offload Engine". Slide 19 refers to the Broadcom "C-NIC" of the demonstration as a "Broadcom TOE".

90. The "C-NIC" in the demonstration of Exhibits G and K is a TOE NIC. It is a network interface card (NIC) that TCP processed an offloaded TCP connection.

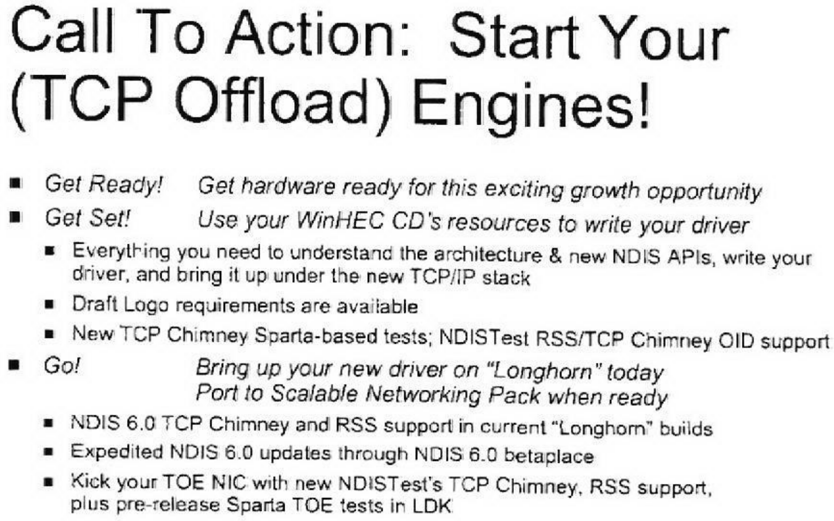
91. The "C-NIC" in the demonstration of Exhibits G and K, together with its associated driver, is an example of "an intelligent TCP offload mechanism".

Induced Infringement of Claim 1 Of The '868 Patent

92. Microsoft has induced others, including attendees of the WinHEC 2004 conference, to use a "set of instructions" that meets every limitation of Claim 1 of the '868 patent. In particular, Microsoft supplied Longhorn build 4074 to the attendees of the WinHEC 2004 conference on the compact disc of Exhibit B as set forth above. Microsoft

also induced the attendees of the WinHEC 2004 conference to write drivers for interfacing Longhorn build 4074 to TCP offload engine NICs. Microsoft also induced attendees of the WinHEC 2004 conference to use the resulting drivers and TOE NICs with Longhorn build 4074 in such a way that a "set of instructions" was present that satisfied all the limitations of Claim 1 of the '868 patent.

93. Microsoft's inducing attendees of the WinHEC 2004 conference to write drivers to interface Longhorn build 4074 to TOE NICs is evidenced in the presentation made at WinHEC 2004 entitled "Writing NDIS ***Drivers*** For TCP Offload Engine NICs" (emphasis added). Slides for this presentation are attached as Exhibit M. Further evidence of Microsoft inducing attendees of the WinHEC 2004 conference to write drivers to interface Longhorn build 4074 to TOE NICs is found in the presentation made by Microsoft at WinHEC 2004 entitled "Designing Quality Advanced Ethernet Adapters And ***Drivers***" (emphasis added). Slides for this presentation are attached as Exhibit L. Slide 30 of Exhibit L is reproduced in part below.



Call To Action: Start Your (TCP Offload) Engines!

- *Get Ready!* *Get hardware ready for this exciting growth opportunity*
- *Get Set!* *Use your WinHEC CD's resources to write your driver*
  - Everything you need to understand the architecture & new NDIS APIs, write your driver, and bring it up under the new TCP/IP stack
  - Draft Logo requirements are available
  - New TCP Chimney Sparta-based tests; NDISTest RSS/TCP Chimney OID support
- *Go!* *Bring up your new driver on "Longhorn" today*
  Port to Scalable Networking Pack when ready
  - NDIS 6.0 TCP Chimney and RSS support in current "Longhorn" builds
  - Expedited NDIS 6.0 updates through NDIS 6.0 betaplace
  - Kick your TOE NIC with new NDISTest's TCP Chimney, RSS support, plus pre-release Sparta TOE tests in LDK

Slide 30 reads, in very large letters, "Call To Action: Start Your (TCP Offload) Engines !" Slide 30 instructs the attendees of the WinHEC 2004 conference to "Use your WinHEC CD's resources to ***write your driver***" (emphasis added). The next bullet of the slide states that the

CDs contain "Everything you need to understand the architecture & new NDIS APIs, ***write your driver***, and bring it up under the new TCP/IP stack" (emphasis added). The slide then instructs the WinHEC 2004 attendees to "***Bring up your new driver*** on "Longhorn" today" (emphasis added).

94. Microsoft is inducing attendees of the WinHEC 2004 conference to not only write a driver and interface a TOE NIC to Longhorn build 4074, but Microsoft is further inducing the attendees to use the resulting system, thereby causing a CPU to run the Longhorn "set of instructions" as well as the driver instructions. Slide 30 of the presentation of Exhibit L instructs the attendees to "Kick your TOE NIC with new NDISTest's TCP Chimney, RSS support, plus pre-release Sparta TOE tests in LDK".

95. By inducing attendees of the WinHEC 2004 conference to write drivers to interface Longhorn build 4074 to "TOE NICs", and by telling attendees of the WinHEC 2004 to "Start Your (TCP Offload) Engines !" and to "Kick your TOE NIC", Microsoft is inducing attendees of the WinHEC 2004 conference to use Longhorn build 4074 in combination with a TOE NIC and its associated driver.

96. A combination of a TOE NIC and its associated driver is an example of an "intelligent TCP offload mechanism" as that term is used in Claim 1 of the '868 patent. According to numerous Microsoft documents, including pages 8 and 9 of the document entitled "Microsoft Windows Scalable Networking Initiative" (Exhibit N), the TCP Chimney functionality of Longhorn causes a TCP connection to be established, and then "offloads all TCP and IP network processing to the offload target". Microsoft is therefore inducing attendees of the WinHEC 2004 to infringe Claim 1 of the '868 patent.

97. The demonstration of Exhibits G and K set forth above is one example of infringement by a company that was induced by Microsoft to infringe Claim 1 of the '868 patent. Personnel from that company, Broadcom Inc., are seen in the video using a "set of instructions" that meets all the limitations of the Claim 1 of the '868 patent.

98. Microsoft states that the compact discs of Exhibits B and D that were distributed at WinHEC 2004 contain "everything needed to understand the architecture" of TCP Chimney. Slide 38 of the WinHEC 2004 presentation entitled "Windows Architecture And Roadmap For Scalable Networking" (Exhibit K) states "your WinHEC CD's resources" contain "everything you need to understand the architecture". Slide 30 of the WinHEC 2004 presentation entitled "Designing Quality Advanced Ethernet Adapters And Drivers" (Exhibit L) also states "your WinHEC CD's resources" contain "everything you need to understand the architecture".

I have personal knowledge of the matters set forth in this declaration and could and would so testify if I were called as a witness. I declare under penalty of perjury under the laws of the State of California and the United States that the foregoing is true and correct.

Dated: November 18, 2004

By: Kevin Almeroth

Dr. Kevin Almeroth