

# **SCSI - Überblick**

von

**Wolfgang Wiese**

**xwolf@xwolf.com**

**<http://www.xwolf.com/>**

## **Inhalt:**

- 1. Einführung**
- 2. Geschichte**
- 3. Grundlagen**
- 4. Zusammenfassung**

## **1. Einführung**

Die Computerschnittstelle SCSI (Small Computer System Interface) hat sich innerhalb der letzten Jahre in nahezu allen Bereichen durchgesetzt. Insbesondere bei RAID-Anwendungen, Servern, Workstations und high-end PCs.

Grund genug um eine Zusammenfassung über das Thema zu geben. Diese Arbeit wird aufgrund des Umfangs die Details nur anreissen, aber dennoch versuchen einen groben Überblick über SCSI zu geben. Angefangen bei der Geschichte, über die Grundlagen der Hardware und des Protokolls und dann eingehend auf einige technische Besonderheiten. Zum Schluß wird noch eine Zusammenfassung gegeben.

Anmerkung: Ursprüngliches Ziel dieser Arbeit war die Zusammenfassung des Artikels „The SCSI Bus“ von Brett Glass in „Under the Hood“. Da dieser Artikel jedoch nur bis ins Jahre 1990 reichte, die Entwicklung aber inzwischen viele Fortschritte gemacht hat, hab ich es mir erlaubt, noch einige andere Quellen hinzuzuziehen um die Entwicklung bis heute darzulegen.

## 2. Geschichte

Im August 1981 wurde der erste Vorläufer von SCSI von der Shugart Ass. vorgestellt: SASI (Shugart Ass. System Interface). Die Entwicklung von SASI begründete sich damals hauptsächlich darin, ein flexibles Produkt für I/O-Operationen auf parallelen Bussen herauszubringen, welches die Vorteile vom bestehenden OEM Channel verbesserte und mindestens ebenso gut war, wie das Intelligent Peripheral Interface (IPI).

1982 begann das ANSI X3T9.2 Komitee mit der Ausarbeitung der Spezifikationen für die nun in SCSI umgenannte Schnittstelle. (Zu diesem Zeitpunkt war dieser recht herstellerspezifische Bus bereits in eine mehr allgemein gehaltene Busschnittstelle umgewandelt worden.) 1984 wurde die Spezifikation fertiggestellt und zwei Jahre später nochmals verbessert. Schon davor wurde SCSI von vielen Geräteherstellern übernommen. Heute bezeichnet man die 1986 fertiggestellte Version als SCSI-1.

Ab Juni 1986 bis Februar 1990 wurde an der Entwicklung von SCSI-2 gearbeitet. Die Unterschiede zu SCSI-1 lagen in höheren Datentransferraten, einen umfangreicheren Statusbit-Satz und einer größeren Kompatibilität zu anderen SCSI-Schnittstellen. (SCSI-2 ist auch abwärtskompatibel zu SCSI-1.)

Neben der Verbesserung der Schnittstelle innerhalb der Hardware wurde ab 1986 auch deutlich, daß eine Normierung auf Seiten der Protokoll-Software erfolgen mußte, so daß das CAM-Komitee gebildet wurde (Common Access Method), welches die vorher recht lockeren Protokoll-Spezifikationen festlegte und einen genormten Befehlssatz herausbrachte.

Ab 1990 begann die Entwicklung von SCSI-3. Dieses unterscheidet sich von SCSI-2 durch Neuerungen am Protokoll, bei den Schnittstellen und bei der Dokumentationsaufteilung. Die Zahl der Geräte am Bus wurde auf 32 erhöht, genauso wie die Zahl der logischen Einheiten auf 32. Zur Zeit werden bestehende alte SCSI-Versionen auf SCSI-3 upgedated.

Neben den Verbesserungen des bestehenden SCSI-Busses und des Protokolls wurde auch die physikalische Schnittstelle erweitert um serielle Übertragungsmöglichkeiten. Schlagwörter hierzu sind FireWire und Fibre Channel.

## 3. Grundlagen

### 3.1. Zielsetzungen von SCSI

Mit SCSI sind insbesondere 3 Zielsetzungen verbunden:

- Einem Rechner werden geräteunabhängige Geräteklassen zur Verfügung gestellt. So können unterschiedliche Festplatten, Bandgeräte, Scanner oder andere Geräte ohne Änderung an allgemeiner Hard- und Software betrieben werden.

- Kompatibilität zwischen Geräten die nach SCSI-1/-2 oder -3-Spezifikation arbeiten. Solche Geräte können am gleichen parallelen SCSI-Bus betrieben werden.
- Gerätespezifische Eigenschaften verbleiben transparent bei den Geräten. Das Betriebssystem kann die technischen Daten und veränderbaren Parameter des Geräts abfragen. Idee des „Plug-und-Play“.

Diese Zielsetzungen werden durch die SCSI-Spezifikation erfüllt. Zu bemerken ist, daß SCSI die Idee des „Plug-und-Play“ schon realisiert hatte, als es diesen Begriff noch garnicht gab. Lediglich die Umsetzung des technisch machbaren (und vorgesehenen) wurde bei vielen Herstellern lange Zeit „verschlafen“.

### 3.2. Prinzipien

Bei den OEM Channel wurden zwei 8bit breite simplex Datenpfade verwendet um das Problem der zu hohen „Channel turnaround time“ im Griff zu bekommen. Ein einziger Pfad, bei dem wechselseitig die Daten in beide Richtungen fließen würden, war nach der damaligen Technik nicht möglich, da der Prozessor zu viel Zeit für die Richtungsänderung benötigte. Bei SCSI wurde dies Problem dadurch gelöst, daß man logische Kommandos einfuhrte, welche nicht mehr nur die Adresse, sondern auch noch die Instruktion mit übergaben.

Neben den logischen Kommandos, ist SCSI in der Lage mehrere Aufträge zur selben Zeit zu erledigen. Dies wird dadurch ermöglicht, daß der Host, der ein Kommando abgibt, gleich nach der Auftragsabsendung die Verbindung auf den Bus wieder beendet. Der Bus wird sobald die Auftrag erledigt wurde, die Verbindung zum Auftragsgeber neu aufbauen und eine positive oder negative Antwort liefern. Während der Auftrag vom Bus bearbeitet wird, kann der Host dann andere Peripheriegeräte ansprechen oder andere Geräte können den Bus benutzen.

Eine weitere Besonderheit an SCSI ist die Verwaltung der angeschlossenen Geräte. Typische Controller beanspruchen die CPU um für Bad Blocks eine „File allocation table“ aufzustellen. Bei SCSI dagegen merkt die CPU davon nichts: Der SCSI Controller kümmert sich selbst darum, so daß die CPU im Falle einer Festplatte keine Bad Blocks sieht.

Eine besonders wichtiges Feature von SCSI ist die Möglichkeit, Befehlswarteschlangen zu bilden: Befehle, die von (verschiedenen oder gleichen) Geräten an ein Device gehen, können an diesem Device in eine Warteschlange einsortiert werden, welche dann später (ggf. optimiert) verarbeitet wird.

### 3.3. Initiator & Target

Der SCSI(-1)-Bus kann bis zu 8 Geräte unterstützen. Diese wiederum können jeweils 8 logische Einheiten haben, welche wiederum je 256 logische Subunits haben können. Jedes Gerät kann ein „Initiator“ sein, d.h. ein Gerät, welches Befehle in Auftrag gibt oder ein „Target“, d.h. ein Gerät, welches Befehle verarbeiten kann.

Nach der Initiierung eines Befehls geht die Busaktivität auf den Target über! (Also nicht wie bei anderen Systemen mit „Master“ und „Slave“.) Der Targer führt dann den Befehl aus und meldet die Erfüllung (ob erfolgreich oder nicht) zum Anschluß an den Initiator zurück. Wie bei der Zielsetzung bereits angegeben, kann der Initiator, während der Target arbeit, andere Dinge bearbeiten. (Dies wirkt sich aber hauptsächlich nur bei Betriebssystemen aus, bei denen richtiges Multitasking implementiert ist.)

Da ein Target auch aus mehr als einer einzigen logischen Einheit bestehen kann, gibt es LUNs (Logical Unit Numbers). Dies ist ähnlich der Unterscheidung zwischen Gerät und Partition einer Festplatte. Eine Datenübertragung geschieht also eigentlich zwischen dem Initiator und einer LUN auf einem bestimmten Target. Die Identifizierung erfolgt über die

IDENTIFY-Message. Daten werden also im Target als logische Blöcke auf logischen Einheiten abgelegt oder von dort geholt. Dieses Konzept vereinfacht die Arbeit der Systementwickler und erleichtert die Ansteuerung verschiedener Geräte derselbe Geräteklasse.

### 3.4. Die Betriebsphasen

Im SCSI-Bus findet sich kein Bus-Takt. Die Signalzeiten werden als Minimal- oder Maximalzeit definiert. Jedes Gerät darf so schnell es kann (innerhalb der Tolleranzen). Das System wird nicht auf das langsamste Gerät abgebremst.

Die zeitliche Abfolge auf dem Bus wird in Zeitabschnitte eingeteilt, den Bus Phasen:

Bus Free (Bus frei), Command (Befehl), Data (In/Out) (Daten), Message (In/Out) (Meldung), Reselection (Wiederauswahl), Selection (Auswahl), Status (Zustand).

Es würde jetzt zu weit führen, die Automaten des Phasenverhaltens zu beschreiben. Ich verweise auf die Quellen. Es sei aber bemerkt, daß bei Uniprozessorsystemen mit einem einzigen angeschlossenen Gerät keine Buszuteilung erfolgen muß, während bei einem System mit mehreren Geräten der Automat erweitert werden muß derart, daß die einzelnen Geräte selbst Verbindungsab-/aufbau machen können. (Durch die RESELECTION-Phase).

### 3.5. Die Geräteklassen

Das SCSI-Protokoll kennt keine individuellen Geräte sondern Klassen von Geräten. Dabei sind Geräte mit gemeinsamen Eigenschaften zusammengefasst. Innerhalb dieser Klassen werden die Geräte gleich behandelt.

Folgende Geräteklassen sind in SCSI-2 definiert:

0	All Device Types
1	Direct-Access Devices
2	Sequential-Devices
3	Printer Devices
4	Processor Devices
5	Write-Once Devices
6	CD-ROM Devices
7	Scanner Devices
8	Optical Memory Devices
9	Medium-Changer Devices
10	Communication Devices

Die SCSI-Befehle sind für die unterschiedlichen Geräteklassen entsprechend definiert. Es wird ein logisches Gerätemodell zugrundegelegt.

### 3.6. CAM

Im Zuge der Verbreitung von SCSI war es nötig, die Protokoll-Spezifikationen genauer zu regeln: In der Spezifikation SCSI-1 wurden zwar viele Befehlsformate eingebaut, aber nur wenige wurden wirklich genutzt. Die Folge war, daß viele Hersteller unterschiedliche Interpretationen der Befehle umsetzten.

Das CAM-Komitee erstellte daraufhin einen allgemeingültigen Befehlssatz, welches Standard werden sollte. Ein Ziel von CAM ist es, gleiche Interfaces für unterschiedliche Geräte zur Verfügung zu stellen, so daß es nicht nötig ist für jedes Gerät eigene Treiber zu erstellen. CAM arbeitet indem es den direkten Zugriff von der Software auf das Gerät unterbindet. Stattdessen sendet die Software, oder das Gerät, auf die Kontroll-Ebene von CAM zu. Diese wiederum greift transparent für das Gerät auf ein Modul zu, welches dann

auf den Host Bus Adapter zugreift um von diesem gemäß der Geräteklasse den Auftrag erfüllen zu lassen.

### 3.7. Kabelforschung

Nachdem 1995 FAST-SCSI herauskam wurde deutlich, daß es einen dringenden Nachholbedarf in der Erforschung von Kabelimpedanz- und Kabelmaterial-Probleme gab. Für die sichere und störungsfreie Signalübertragung auf dem SCSI-Bus ist es nötig, die Leitungen an beiden Enden mit einem Wellenwiderstand (Impedanz) abzuschließen. Damit werden kurzzeitige Spannungsüberhöhungen oder -einbrüche weitgehend unterbunden. Somit sind die wichtigsten Parameter für eine Übertragungstrecke Impedanz, Kapazität, Flankensteilheit und Busabschluß (Termination). Die Impedanz ist mit den Eckwerten 90 Ohm (unsymmetrisch) und 132 Ohm (symmetrisch) definiert. Wichtig ist dabei nicht die Höhe, sondern die Gleichmäßigkeit auf der gesamten Strecke, inklusiv all ihren Bestandteilen (Treiber, Leiterbahnen, Steckverbindungen, Kabel und Terminierungen). Gering gehalten werden müssen die Kapazitäten. Diese sind mit die größten Geschwindigkeitsbremsen. Mit der ersten SCSI-Spezifikation wurde 25pF / Teilnehmer zugelassen; moderne Geräte haben heute Werte um 15 pF. Die Flankensteilheit hat ebenfalls Einfluß auf die Übertragung, da zu lange Zeiten die Erkennung der Schaltschwelle erschweren, zu kurze aber hochfrequente Energie erzeugen, welche stört. Außerdem besteht die Gefahr der Doppeltriggerung durch Reflektionen. Neben den Besonderheiten bei der Kabeltechnik ist es auch wichtig die Kabellängen und die Übertragungsmethoden zu beachten: Die Leitungsimpedanzen am physikalisch identischen Kabel fallen so unterschiedlich aus, je nachdem ob das Kabel für symmetrischen oder unsymmetrischen Betrieb eingesetzt wird.

## 4. Zusammenfassung

In seinen Abschußsatz zeichnete Brett Glass eine rosige Zukunft für SCSI. Und obwohl seitdem schon 8 Jahre vergangen sind, scheint er auch heute noch recht zu haben. Schon mehrfach totgesagt hat sich SCSI auf professionellen Geräten durchgesetzt und es ist eine stärkere Verbreitung zu erwarten.

Dennoch darf dies nicht darüber hinwegtäuschen, daß das SCSI von vor 15 Jahren nicht mehr dasselbe ist, wie es heute ist. Es hat eine lange Entwicklungsphase hinter sich und wird noch weiter verändert werden. Entwicklungen wie FireWire, welche auf das SCSI-Protokoll aufsetzen, zeigen wohin es geht. Betrachtet man zusätzlich noch die gestiegene Verbreitung von Multitasking-fähigen Betriebssystemen, ist im Moment auch keine Trendwende zu sehen. Die Zukunft dürfte für SCSI noch eine Weile lang rosig bleiben.