



Doctoral Thesis

Multimedia communication subsystems: architecture, interfaces and implementation

Author(s):

Rütsche, Erich Marcel

Publication Date:

1993

Permanent Link:

<https://doi.org/10.3929/ethz-a-000943652> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH Nr. 10228

Multimedia Communication Subsystems: Architecture, Interfaces and Implementation

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of

DOCTOR OF TECHNICAL SCIENCES

presented by

Erich Marcel Rütscbe
Dipl. El.-Ing. ETH

born May 7, 1963
citizen of Kirchberg SG



CatE

accepted on the recommendation of

Prof. Dr. B. Plattner, examiner
Prof. Dr. H. Burkhart, Universität Basel, co-examiner

1993

Zusammenfassung

Diese Arbeit untersucht die Architektur, die Schnittstellen und die Implementation von Kommunikationssystemen. Es wird eine neue Hardwarearchitektur vorgeschlagen, die sich besonders für die Übertragung multimedialer Daten eignet. Entwurfsziel ist die Behandlung von isochronen Multimediaprotokollen bei einer Netzwerkdatenrate von 622 Mb/s. Konventionelle Transportprotokolle werden jedoch ebenfalls unterstützt. Am Beispiel von TCP/IP lässt sich eine Leistung von mehr als 35000 TCP/IP Segmente/s vorhersagen.

Die neue Architektur wurde aus Experimenten mit der *Parallel Protocol Engine* (PPE) entwickelt. Anhand von TCP/IP wurde die parallele Implementation eines Transportprotokolls auf dem PPE untersucht. Die Parallelität zwischen Sender und Empfänger konnte gut ausgenutzt werden, während die Parallelität zwischen Protokollschichten wegen der unterschiedlichen Rechenzeitanforderungen der Schichten (TCP und IP) nur sehr unzureichend ausgenutzt werden konnte. Die Schnittstellen zwischen Kommunikationssystem und Anwendung auf dem Hostrechner erwiesen sich als der grösste Engpass.

Multimedia-Anwendungen erfordern die Echtzeitbehandlung von Protokollen, damit ihre Anforderungen an die Dienstqualität erfüllt werden können. Um die Echtzeitbehandlung von multimedialen Datenströmen zu garantieren, wurde der *Multimedia Protocol Adapter* (MPA) entwickelt, der isochrome Datenströme getrennt von asynchronem Verkehr behandelt. Diese Trennung wird mit dem Protokollfilter erreicht. Der Protokollfilter extrahiert die Verbindungsinformation eines Paketkopfes und generiert eine eindeutige Verbindungsnummer. Multimediale Daten werden anhand dieser Nummer erkannt und ohne Eingriff des Protokollprozessors in ein besonderes *Multimedia-FIFO* geschrieben. Das Multimedia-FIFO bildet die Schnittstelle zu einem Multimediaibus des Hostsystems. Datenpakete asynchroner Verbindungen werden aufgetrennt in ihren Kopf- und Datenteil in getrennte Speicher geschrieben. Der Prozessor bearbeitet die Protokollinformation im Paketkopf und gibt die Daten über einen programmierbaren Buscontroller an den Host weiter. Die Realisierung des Protokollfilters, der Prüfsummeneinheit und der DMA-Einheit in Hardware erlaubt die überlappende Abarbeitung dieser Funktionen. Die Parallelität zwischen Sender und Empfänger wird im MPA durch den Einsatz von zwei dedizierten Prozessoren für die Protokollbearbeitung ausgenutzt.

Die analytische Untersuchung der Busschnittstelle zum Hostsystem zeigt, dass der konventionelle E/A-Bus und die traditionelle Anwendungsschnittstelle für multimediale Daten ungeeignet sind. Die Untersuchung belegt die Annahme, dass besondere Schnittstellen für Multimedia-kommunikation sinnvoll und notwendig sind.

Das grundlegende Konzept des MPA, die Trennung von isochronen und asynchronen Datenströmen durch die hardwareunterstützte Extraktion der Verbindungsinformation aus dem Datenpa-

ket ist auch bei einfacher aufgebauten Systemen anwendbar. Als Beispiel dafür wird der *Light-Weight Multimedia Adapter* vorgestellt, der keinen Protokollprozessor besitzt. Für den asynchronen Datenverkehr unterstützt diese Architektur die Prüfsummenberechnung und die allfällige Datenkodierung und -dekodierung. Die eigentlichen Protokolle werden aber auf dem Host behandelt.

Abstract

This thesis investigates the architecture, interfaces and implementations of communication subsystems and proposes a new architecture which is specially suited for multimedia. The new architecture supports processing of isochronous multimedia protocols at the network speed of 622 Mbit/second. Conventional transport protocols are also supported by the architecture. The calculated performance is more than 35000 TCP/IP segments/second.

The new architecture is based on experiments with the Parallel Protocol Engine (PPE). We implemented TCP/IP to investigate the parallel implementation of a standard transport protocol stack on the PPE. Parallelism between sender and receiver was exploited optimally, whereas parallelism between protocol layers was difficult to implement because of the different processing requirements of the layers. The interface between communication subsystem and the application on the host proved to be the major performance bottleneck.

Multimedia applications require realtime processing of protocols to satisfy their quality of service. To guarantee realtime processing of multimedia data streams, we developed the Multimedia Protocol Adapter (MPA) which separates isochronous multimedia traffic from the asynchronous traffic. The separation is performed with the assistance of the Protocol Filter. The Protocol Filter extracts the connection information of a packet header and generates a unique connection number. Multimedia data are determined by this number and are written to a multimedia FIFO, bypassing the protocol processor. The multimedia FIFO is the interface to a multimedia bus of the host system. Asynchronous data are split into header and data part and are written to separated header and data memories. The processor performs the protocol processing on the header and forwards the data via a programmable bus controller to the host. The hardware implementation of the protocol filter, checksumming unit and DMA unit allows for packet processing in a hardware pipeline. Parallelism is exploited between the sender and receiver part of the multimedia adapter and in the pipelined execution of processing-intensive functions in hardware. The queueing analysis of the bus interface showed that the I/O bus and the socket interface are not suited for multimedia communication. The analysis proved our assumption that a separate I/O interface is required for multimedia communication.

The main concept of the MPA architecture, the separation of isochronous and asynchronous data by the extraction of the connection information of a packet, can also be used in simpler architectures. We propose the Light-weight Multimedia Adapter (LMA), which supports multimedia on B-ISDN networks. The LMA provides checksumming and coding/decoding but protocol processing is performed on the host.