Architekturen für Ethernet-basierte Teilnehmerzugangsnetzwerke und deren Umsetzung in Hardware

Dissertation

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock

vorgelegt von
Kubisch, Stephan, geb. am 26.11.1978 in Rostock
aus Rostock
Rostock, 09.03.2009
URN: **urn:nbn:de:gbv:28-diss2009-0148-9**

Betreuer:

- Prof. Dr.-Ing. Dirk Timmermann
  Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik

Gutachter:

- Prof. Dr.-Ing. Uwe Schwiegelshohn
  Technische Universität Dortmund, Institut für Roboterforschung, Abteilung Informationstechnik
- Prof. Dr.-Ing. habil. Djamshid Tavangarian
  Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Informatik, Lehrstuhl für Rechnerarchitektur

Tag der Einreichung: 09.03.2009
Tag der Verteidigung: 17.07.2009

Keine Schuld ist dringender, als die, Danke zu sagen.
(Marcus Tullius Cicero, röm. Politiker, Anwalt & Philosoph)

Diese wissenschaftliche Arbeit entstand während meiner Tätigkeit am Institut für Angewandte Mikroelektronik und Datentechnik der Universität Rostock.

Großer Dank gilt in erster Linie dem Betreuer dieser Arbeit, Prof. Dr.-Ing. Dirk Timmermann, welcher durch die Bereitstellung der nötigen Rahmenbedingungen diese Arbeit erst ermöglicht hat. Gerade in den entscheidenden Phasen hat er für die notwendige Motivation gesorgt, die zur erfolgreichen Fertigstellung der vorliegenden Dissertation geführt hat.

Ich bedanke mich bei meinen Kommilitonen und Kollegen Frank & Harry für die angenehme Zeit während des gesamten Studiums in Rostock. Ihnen und allen anderen Reviewern, allen voran Prof. Dr.-Ing. Hartmut Pfüller, vielen Dank für das eifrige Korrekturlesen. Ich danke zudem dem Team des Instituts für die angenehme Arbeitsatmosphäre, insbesondere den Kollegen der Verwaltung und Administration, die einem den Arbeitsalltag oft erleichterten.

Ich möchte mich weiterhin beim gesamten Team der Nokia Siemens Networks GmbH & Co. KG für die harmonische und fruchtbare Kooperation der vergangenen Jahre bedanken, allen voran Herrn Dipl.-Ing. (FH) Daniel Duchow und Herrn Dipl.-Inform. Thomas Bahls.

Ein großes Dankeschön geht an die vielen Studenten, durch deren Arbeiten diese Dissertation an Fleisch gewonnen hat. Dies sind die Damen und Herren Andy Strzeletz, Martin Siemroth, Stephan Spies, Sara Hernández Burgos, Lukasz Krukowski, Enrico Heinrich, Peter Danielis, Christian Lange, Oliver Röwer, Tino Demuth, Grit Rhinow, Enrico Daum und Ansgar Waschki.

Meiner Familie gilt ein spezieller Dank für die beständige Unterstützung während meiner Studienzeit. Sorgenfrei konnte ich mich auf die Arbeit konzentrieren. Zudem danke ich Sabrina, die mir trotz großer Entfernung und rarer Zweisamkeit, den nötigen Raum zum Schreiben der Arbeit gab. Viele Passagen der Arbeit entstanden im Zug zwischen Rostock und Hamburg.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungen & Symbole

Thematisch ist diese Arbeit u. a. im Bereich der Telekommunikation angesiedelt. Dieses Gebiet ist für gewöhnlich durch eine Vielzahl von Fachbegriffen und Abkürzungen geprägt, welche in der folgenden Übersicht aufgeführt sind.

Verwendete Abkürzungen

A

AAA ..... Authentication Authorization Accounting
ACIP ..... Access Control and Information Protocol; generisches Verwaltungsprotokoll
Ack ..... Acknowledge; Steuersignal zur Flusskontrolle
ADSL ..... Asymmetric DSL; asymmetrische DSL-Variante
AH ..... Authentication Header; Sicherheitsprotokoll für IPsec
AIA ..... Additional Information Adder; Komponente des IPclip-Prototyps
AIR ..... Additional Information Remover; Komponente des IPclip-Prototyps
AMBA ..... Advanced Microcontroller Bus Architecture; Busspezifikation
APCG ..... Application Characterization Graph; Darstellungsform systeminterner Abhängigkeiten
ARCG ..... Architecture Characterization Graph; formale Notation einer NoC-Architektur
ARP ..... Address Resolution Protocol
ASIC ..... Application Specific IC; anwendungsspezifischer integrierter Schaltkreis
ATM ..... Asynchronous Transfer Mode; Datenübertragungstechnik

B

BEAM ..... Border-Enhanced Mesh; alternative NoC-Topologie
BFM ..... Bus Functional Model; funktionales Schnittstellenmodell in einer Testharness
BITKOM ..... Bundesverband Informationswirtschaft, Telekommunikation und neue Medien
BRAS ..... Broadband Remote Access Server; erster Router im Kernnetz

C

CDC ..... Clock Domain Crossing; Übergang in andere Taktdomänen
chaddr ..... Client Hardware Address; Feld im DHCP-Header
CLIP ..... Calling Line Identification Presentation; ISDN-Feature
CMAC ..... Customer MAC; MAC-Adresse eines Teilnehmers
CPE ..... Customer Premises Equipment; Endgeräte beim/des Kunden
CPU ..... Central Processing Unit
CRC ..... Cyclic Redundancy Check; Verfahren zur Fehlererkennung
CRR ..... Core-to-Router Ratio; Anzahl der IP-Cores pro Router einer NoC-Topologie
CSMA/CA .... Carrier Sense Multiple Access/Collision Avoidance; Medien-Zugriffsverfahren
CSMA/CD .... Carrier Sense Multiple Access/Collision Detection; Medien-Zugriffsverfahren

CTG ..... Communication Task Graph; Notation zur Darstellung systeminterner Abhängigkeiten
CuPON ..... Copper (Cu)-PON; sinnbildliche Bezeichnung für Kupfer im Gigabit-Bereich

D

DDoS ..... (Distributed) Denial-of-Service Attack; Kompromittierung der Dienstverfügbarkeit
DHCP ..... Dynamic Host Configuration Protocol; Protokoll zur Vergabe von IP-Adressen
DOR ..... Dimension-ordered Routing; Klasse von Routing-Algorithmen
DPG ..... Design Productivity Gap
DSL ..... Digital Subscriber Line; digitaler Teilnehmer-Anschluss für hohe Datenraten
DSL-AC ..... DSL-Access Concentrator; erster DSL-Knotenpunkt im Kernnetz
DSLAM ..... DSL Access Multiplexer; Komponente eines TZN
DST ..... Destination; Datensenke
DTAG ..... Deutsche Telekom AG; Marktführer Telekommunikation Deutschland
DUID ..... DHCP Unique Identifier; eindeutiger Hostbezeichner bei DHCPv6
DUT ..... Device-under-Test; zu verifizierende Entität in einer Simulationsumgebung
DVB-H ..... Digital Video Broadcasting-Handhelds; digitaler Videorundfunk f. tragbare Geräte

E

EFM ..... Ethernet-in-the-first-Mile; Einsatz von Ethernet im Kundenanschlussbereich
EMAC ..... Embedded Medium Access Controller; Ethernet-Schnittstelle in Xilinx FPGAs
ENUM ..... Telefon Number Mapping
EoF ..... End-of-Frame; letztes Bit/Byte einer zu übertragenden PDU
ESP ..... Encapsulating Security Payload; Sicherheitsprotokoll für IPsec

F

FCS ..... Frame Check Sequence; CRC Prüfsumme in Ethernet-Frames
FDB ..... Filtering/Forwarding Database; Adresstabelle eines Switches
FF ..... Flipflop; Register, Speicherelement
FIFO ..... First-In-First-Out
FLIT ..... Flow Control Digit; Flusskontrolleinheit auf Schicht 2
FM ..... Funktionsmodul; funktional abgeschlossene(r) Entität/Block
FPGA ..... Field Programmable Gate Array; (feld)programmierbarer Halbleiterbaustein
FTP ..... File Transfer Protocol; Dateiübertragungsverfahren/-protokoll
FTTx ..... Fiber-to-the-x; versch. Ausprägungen optischer Verkabelung

G

GALS ..... Global Asynchron Lokal Synchron; Taktschema digitaler, eingebetteter Systeme
GAN ..... Global Area Network; Weitverkehrsnetz mit globalen Ausmaßen
GbE ..... Gigabit Ethernet; Kabelgebundene Übertragungstechnologie für Datennetze
GDSL ..... Gigabit DSL; Gigabit DSL
GLI ..... Geospatial Location Information
GPON ..... (Gigabit) Passives Optisches Netz; Übertragungstechnik mittels Glasfaser
GPS ..... Global Positioning System
GUI ..... Graphical User Interface; graphische Schnittstelle f. Mensch-Maschine-Interaktion
GWHS ..... Getaktetes Wormhole Switching; Flusskontrollverfahren

H

HC ..... Hop-Counter; Zähler für HSM innerhalb eines NoC-Router-Ports
HDSL ..... High Data Rate DSL; symmetrische DSL-Variante
HDSPA ..... High Speed Downlink Packet Access; Übertragungsverfahren innerhalb von UMTS
HOL ..... Head-of-Line-Blocking; Blockierung von PDUs am Kopf einer Warteschlange
HSM ..... Hybrider Switching-Mechanismus; Flusskontrollverfahren
HTTP ..... Hypertext Transfer Protocol; Datenübertragungsprotokoll der Anwendungsschicht
HVt ..... Hauptverteiler

I

IANA ..... Internet Assigned Numbers Authority; Internet-Organisation
IC ..... Integrated Circuit; integrierter (elektronischer) Schaltkreis
ICANN ..... Internet Corporation for Assigned Names and Numbers; Internet-Organisation
ICMP ..... Internet Control Message Protocol
IDS ..... Intrusion Detection System; System zur Angriffserkennung
IEEE ..... Institute of Electrical and Electronics Engineers; internationaler Ingenieursverband
IETF ..... Internet Engineering Task Force; Internet-Organisation
IHL ..... Internet Header Length
IKE ..... Internet Key Exchange; Schlüsselaustauschprotokoll
IP(v4/v6) ..... Internet Protocol (Version 4/6); verbindungsloses Basisprotokoll im Internet
IP-Core ..... Intellectual Property Core; (lizenzpflichtiges) Funktionsmodul, Teilmodul eines SoCs
IPclip ..... Internet Protocol-Calling Line Identification Presentation
IPoE ..... IP-over-Ethernet; Netzwerkprotokoll zum Verbindungsaufbau
IPsec ..... Internet Protocol Security; Sicherheitsarchitektur für IP
IPTV ..... IP Television; Internetfernsehen
ISDN ..... Integrated Services Digital Network; digitales Telekommunikationsnetzwerk
ISO ..... International Organization for Standardization; Standardisierungsorganisation
ISP ..... Internet Service Provider; Dienstanbieter im Internet

K

KWHS ..... Kombinatorisches Wormhole Switching; Flusskontrollverfahren

L

LAN ..... Local Area Network; lokales Netzwerk
LCM ..... Local Control Module; lokale Verwaltungseinheit
LDP ..... Label Distribution Protocol; Protokoll zur Verbreitung von MPLS-Labels
LER ..... Label Edge Router; MPLS-Randrouter
LI ..... Location Information; Kombination versch. Ortsinformationen in einer IPclip-Option
LLC ..... Logical Link Control; Schicht in IEEE 802
LSB ..... Least Significant Bit; niederwertigstes Bit
LSP ..... Label Switched Path; MPLS-Pfad durch ein Netzwerk
LSR ..... Label Switched Router; MPLS-Kernrouter

M

MAC ..... Medium Access Control; Medienzugriffskontrolle
MAM ..... MTU Adaptation Module; Komponente des IPclip-Prototyps
MAN ..... Metropolitan Area Network; Stadtbereichsnetz, regionales Netzwerk
MAS ..... MAC Address Stacking; Kapselungsschema bei Ethernet

MAT ..... MAC Address Translation; Ersetzungsmechanismus für MAC-Adressen
MATMUNI ... MAT, TM und MPLS-UNI; Name eines paketverarbeitenden Systems
MiM ..... MAC-in-MAC; Bezeichnung für IEEE Std 802.1ah
MPLS ..... Multi Protocol Label Switching; verbindungsorientiertes Vermittlungsverfahren
MPSoC ..... Multi Prozessor SoC; SoC auf Basis homogener Prozessoren
MSB ..... Most Significant Bit; hochwertigstes Bit
MTA ..... Mail Transfer Agent; zentrale E-Mail-Software/Server eines ISPs
MTBF ..... Mean Time Between Failures; durchschn. Zeit bis zum Auftreten eines Fehlers
MTU ..... Maximum Transmission Unit; maximale PDU-Größe eines Protokolls
MUX ..... Multiplexer; Selektionsschaltnetz der analogen Elektronik- und Digitaltechnik

N

NAT ..... Network Address Translation; Adressumsetzung auf der Vermittlungsschicht
NIC ..... Network Interface Card; Schnittstelle zwischen Rechner und Übertragungsmedium
NMEA ..... National Marine Electronics Association
NoC(s) ..... Network(s)-on-Chip; Kommunikationsinfrastruktur für integrierte Systeme

O

OAM ..... Operation Administration Maintenance
OCIN(s) ..... On-Chip-Interconnection-Network(s); Chip-interne Verdrahtungsnetze
OSI ..... Open Systems Interconnection
OVM ..... Option Verification Module; Komponente des IPclip-Prototyps

P

P2P ..... Peer-to-Peer; Rechner-zu-Rechner-Verbindung
PAM ..... PPPoE MTU Adaptation Module; Komponente des IPclip-Prototyps
PDU ..... Protocol Data Unit; abgeschlossene Dateneinheit eines Protokolls
PF ..... Peering Flag; Status-Bit einer IPclip-Option
PGP ..... Pretty Good Privacy; Sicherheitstool auf Basis von Public-Key-Verfahren
PHIT ..... Physical Transfer Digit; kleinste Informationseinheit auf Schicht 1
PMAC ..... Provider MAC; vom ISP definierte MAC-Adresse
PMTU ..... Path MTU; bestimmte MTU eines Datenpfades
PMTUD ..... Path MTU Discovery; Anpassung der MTU bei IPoE
PoP ..... Point-of-Presence; Knotenpunkt in Datennetzen
POTS ..... Plain Old Telephone Service; klassische, analoge Telefondienste
PPP ..... Point-to-Point-Protocol; Netzwerkprotokoll zum Verbindungsaufbau
PPPoE ..... PPP-over-Ethernet; Netzwerkprotokoll zum Verbindungsaufbau
PSTN ..... Public Switched Telephone Network; leitungsvermitteltes Telefonnetz

Q

QiQ ..... Q-Tag-in-Q-Tag; Bezeichnung für verschachtelte VLANs nach IEEE Std 802.1ad
QoE ..... Quality of (User) Experience; Subjektive Zufriedenheit eines Nutzers
QoS ..... Quality of Service; Dienstgüte in der Telekommunikation

R

RARP ..... Reverse Address Resolution Protocol
RDNI ..... Resource Dependent Network Interface; Ressourcen-Schnittstelle des RNI

Req ..... Request; Steuersignal zur Flusskontrolle
RF ..... Removal Flag; Status-Bit einer IPclip-Option
RFC ..... Request for Comments; Diskussions- & Standardisierungsvorschlag
RINI ..... Resource Independent Network Interface; NoC-Schnittstelle des RNI
RISC ..... Reduced Instruction Set Computing; Bez. f. Prozessoren mit reduziertem Befehlssatz
RNI ..... Resource Network Interface; Brückenmodul in einem NoC

S

S/MIME ..... Secure/Multipurpose Internet Mail Extensions; (Sicherheits-)Standard für E-Mails
SAFS ..... Store-and-Forward Switching; Flusskontrollverfahren
SAN ..... Storage Area Network; Speichernetzwerk
SCA ..... Subscriber Catchment Area; Einzugsgebiet eines Zugangsknotens
SDSL ..... Symmetrical Data Rate DSL; symmetrische DSL-Variante
SF ..... Source Flag; Status-Bit einer IPclip-Option
SID ..... Service-ID
SIP ..... Session Initiation Protocol
SLA ..... Service Level Agreement; Dienstgütevereinbarung
sMAT ..... simplified MAC Address Translation; Ersetzungsmechanismus für MAC-Adressen
SMTP ..... Simple Mail Transfer PProtocol; Protokoll zum E-Mail-Versand
SNMP ..... Simple Network Management Protocol; Netzwerk-Verwaltungsprotokoll
SoC(s) ..... System(s)-on-Chip; modulares integriertes System
SoF ..... Start-of-Frame; erstes Bit/Byte einer zu übertragenden PDU
SONET ..... Synchronous Optical Network; Übertragungstechnik für synchrone Datenströme
Spam ..... Spiced Ham; störende, unverlangte Massennachrichten per E-Mail
SRC ..... Source; Datenquelle
SSL ..... Secure Sockets Layer; Sicherheitsprotokoll der Transportschicht

T

TAE ..... Telekommunikationsanschlusseinheit; 'Telekom-Steckdose'
TAL ..... Teilnehmeranschlussleitung
TBA ..... Trust-by-Authentication
TBW ..... Trust-by-Wire
TCP ..... Transmission Control Protocol; verbindungsorientiertes Transportprotokoll
TF ..... Trustability Flag; Status-Bit einer IPclip-Option
TKG ..... Telekommunikationsgesetz
TLS ..... Transport Layer Security; Sicherheitsprotokoll der Transportschicht
TM ..... Traffic Manager; Mechanismus zur Regelung des Datenvolumens in Netzwerken
TZN ..... Teilnehmerzugangsnetzwerk; Teil des Gesamtnetzes zur Anbindung von Teilnehmern

U

UDP ..... User Datagram Protocol; verbindungsloses Transportprotokoll
UMTS ..... Universal Mobile Telecommunication System; Mobilfunkstandard, 3. Generation
UNI ..... User-to-Network-Interface; Schnittstelle zwischen Nutzer und Netzwerk
URI ..... Uniform Resource Identifier
URL ..... Uniform Resource Locator; allg. Bezeichnung für Adressen von Webseiten

UTC ..... Universal Time (coordinated); koordinierte Weltzeit

V

VCTS ..... Virtual cut-through Switching; Flusskontrollverfahren
VDSL ..... Very High Data Rate DSL; DSL-Variante
VDT ..... Valid Data Toggle; Steuersignal des HSM
VHDL ..... Very High Speed Integrated Circuit Hardware Description Language; Hardwarebeschreibungssprache
VID ..... VLAN-ID; VLAN-Bezeichner im Q-Tag
VLAN ..... Virtual Local Area Network; virtuelles LAN
VoIP ..... Voice-over-IP; Internettelefonie
VPLS ..... Virtual Private LAN Services; VPN-Dienste
VPN ..... Virtual Private Network; logisches, meist verschlüsseltes Overlay-Netzwerk

W

WAN ..... Wide Area Network; Weitverkehrsnetz
WGS84 ..... World Geodetic System 1984; globales Koordinatensystem
WHS ..... Wormhole Switching; Flusskontrollverfahren
WiMAX ..... Worldwide Interoperability for Microwave Access; IEEE Std 802.16
WLAN ..... Wireless LAN; lokales Funknetzwerk
WWW ..... World Wide Web; abrufbares Hypertextsystem als eine Nutzungsform des Internets

X

xDSL ..... Sammelbegriff für DSL-Varianten
XML ..... Extensible Markup Language; erweiterbare Auszeichnungssprache

Symbole & Formelzeichen

α ..... (Alpha) Aktivität, Wahrscheinlichkeit eines 0-1-Pegelwechsels
Δ_φ ..... (Delta) Phasendifferenz/-unterschied [°]
Δ_k ..... (Delta) Differenz der Kantenlänge zweier verschiedener Gitter-Topologien
$\mathcal{G}$ ..... Graph
$\mathcal{M}$ ..... Abbildung bzw. Mapping
$\mathcal{Z}$ ..... Optimierungsziel
Θ_S ..... (Theta) Sättigungsdurchsatz [b/s] bzw. [%]
Θ_S(k) ..... (Theta) Sättigungsdurchsatz einer NoC-Topologie mit Kantenlänge k [b/s] bzw. [%]
Θ_{Eth} ..... effektiver Durchsatz von Ethernet [b/s] bzw. [%]
φ ..... (Phi) Phase [°]
B ..... Bisektion einer Netzwerktopologie [# Kanäle]
BW ..... Bandbreite [b/s]
BW_B ..... Bisektionsbandbreite eines NoC [b/s]
$BW_{Channel}$ .... Bandbreite eines unidirektionalen Übertragungskanals im NoC [b/s]
BW_{Eth} ..... Bandbreite einer Ethernet-Variante [b/s]
BW_{NoC} ..... Gesamtbandbreite eines NoC [b/s]
BW_{Router} ..... interne Gesamtbandbreite eines NoC-Routers [b/s]
C_0 ..... Grundkapazität pro Längeneinheit [F]

C_L ..... Leitungskapazität [F]
D ..... Menge aller Abhängigkeiten $d_{i,j}$ einer Anwendung
d ..... Länge eines Übertragungspfades im NoC [Hops]
D' ..... Menge aller Kommunikationsprozesse $d'_{i,j}$ einer Anwendung
$d'_{i,j}$ ..... Kommunikationsprozess zwischen p'_i und p'_j
d_{avrg} ..... durchschnittliche Pfadlänge einer NoC-Topologie [Hops]
$d_{i,j}$ ..... Daten- oder Steuerabhängigkeit zwischen p_i und p_j
d_{max} ..... maximale Pfadlänge bzw. Durchmesser in einer NoC-Topologie [Hops]
d_{min} ..... minimale Pfadlänge in einer NoC-Topologie [Hops]
f ..... Frequenz [Hz]
K ..... Anzahl bzw. Menge aller unidirektionalen Kanäle $k_{i,j}$ in einem NoC
$k_{(x,y)}$ ..... Kantenlänge eines k-fachen 2-Würfels (der X- bzw. Y-Dimension)
$k_{i,j}$ ..... physikalischer Datenkanal zwischen n_i und n_j
l ..... Leitungslänge [m]
L_{bit} ..... Länge einer PDU [b]
L_{Frame} ..... Länge bzw. Größe eines Ethernet-Frames [B]
L_{IP-Opt} ..... Länge bzw. Größe einer IP-Option [B]
L_{Paket} ..... Länge bzw. Größe eines IP-Paketes [B]
MTU_{config} .... konfigurierter Wert der MTU des IPclip-Systems [B]
MTU_{IPclip} .... von IPclip modifizierter Wert für die PMTU [B]
MTU_x ..... Bezeichner f. eine beliebige MTU [B]
N ..... Anzahl bzw. Menge der IP-Cores bzw. Verarbeitungselemente n_i in einem NoC
n, m ..... ganzzahlige Variable
$n_{blocked}$ ..... Wartezeit eines Pakets aufgrund von Blockierungen im NoC [Takte]
n_{buf} ..... Wartezeit eines Frames im Puffer bis zur Arbitrierung [Takte]
n_{flits} ..... Größe bzw. Länge eines Pakets [Flits]
n_{func} ..... inhärente Verzögerung von Funktionsmodulen [Takte]
n_{hc} ..... Zähler für Routing-Schritte [Hops]
n_i ..... IP-Core bzw. Verarbeitungselement in einem NoC
n_{mem} ..... Dauer der Suche im Speicher [Takte]
n_{route} ..... interne Verzögerung eines NoC-Routers [Takte]
$n_{signaling}$ ..... Verzögerung durch interne Signalisierungen im MATMUNI-System [Takte]
n_{sync} ..... Synchronisationsverzögerung einer CDC-Schaltung für ein Handshake [Takte]
P ..... Menge aller Prozesse bzw. Aufgaben p_i einer Anwendung
P' ..... Menge aller unabhängigen Funktionsmodule p'_i einer partitionierten Anwendung
p'_i ..... unabhängiges Funktionsmodul in einer partitionierten Anwendung
P_V ..... Verlustleistung [W]
p_i ..... Aufgabe, Prozess bzw. Teilfunktion innerhalb einer Anwendung
R ..... Anzahl bzw. Menge der Router in einem NoC
R_0 ..... Grundwiderstand pro Längeneinheit [Ω]
R_L ..... Leitungswiderstand [Ω]
r_i ..... Knoten bzw. Router in einem NoC
t ..... Zeitdauer, Latenz, Verzögerung [s]
$t(k)$ ..... Latenz in einer NoC-Topologie mit Kantenlänge k [s]
t_0 ..... lastfreie Latenz (Zero-Load Latency) [s]

t_D	Leitungsverzögerung (Wire Delay) [s]
$t_{0,avrg}$	durchschnittliche lastfreie Latenz [s]
$t_{0,min}$	minimale lastfreie Latenz [s]
t_{avrg}	durchschnittliche Latenz [s]
T_{clk}	Taktperiode [s]
V_{dd}	Versorgungsspannung [V]
W_{bit}	Kanalbreite [b]
x,y	Koordinaten im zweidimensionalen Gitter
x_{max},y_{max}	maximale Ausdehnung eines zweidimensionalen Gitters

Teil I.

Entwicklungen in der Telekommunikationsbranche & Grundlagen

Discovery consists of seeing what everybody has seen and thinking what nobody has thought.
(Albert v. Szent-Györgyi Nagyrápolt, ungar. Nobelpreisträger)

Kapitel 1.

Einleitung

Die moderne Informationsgesellschaft ist sowohl durch den ubiquitären Charakter verschiedenster Netzwerke als auch durch eine hohe und zunehmende Wertschöpfung auf Basis elektronischer Datenverarbeitung gekennzeichnet [Kle08]. Das noch immer gültige Mooresche Gesetz von 1965 [Moo65] hat mit Beginn des Informationszeitalters in den 70-iger & 80-iger Jahren ein ebenbürtiges Pendant bekommen [CO01]: Die Verkehrsstatistiken des Deutschen Internet-Zentralknotens in Frankfurt [DCX] belegen eine jährliche Verdopplung des Datenaufkommens im Internet, welches somit ein noch stärkeres Wachstum als das der Integrationsdichte digitaler Schaltkreise nach Moore aufweist. Die Gründe dafür sind der Erfolg und das rasante Wachstum des Internets [Mil07] und der damit einhergehende Paradigmenwandel [HC03, SS06]. Abbildung 1.1 veranschaulicht diesen Wandel anhand der Definition des Nutzwertes von Netzwerken in Abhängigkeit von der Anzahl der Teilnehmer (n). Während in klassischen Rundfunknetzen der Nutzwert eines Netzwerks nach Sarnoff proportional zu n ist, führte die rasante Adaption des Internets, des sogenannten Web 2.0 und darauf aufbauender Dienste dazu, dass sich durch Ausprägung vieler unabhängiger Teilnetze innerhalb eines Gesamtnetzes dessen Nutzwert darüber hinaus enorm steigert. Dieser exponentielle Zusammenhang ist im Reedschen Gesetz festgehalten ($\approx 2^n$). Konnektivität – die Verfügbarkeit eines möglichst direkten und schnellen „Drahtes“ zum gewünschten Kommunikationsendpunkt – ist somit der primäre Treiber für die weitere Entwicklung der Informationsgesellschaft [Odl01a]. Nortel Networks beschreibt den Zustand dieser umfassenden Vernetzung als Hyperkonnektivität [Nor07a].

Zielstellung Die vorliegende Dissertationsschrift beschäftigt sich mit abgeleiteten Problemstellungen und Auswirkungen des o. g. Paradigmenwechsels. Die Arbeit behandelt zwei verwandte Themenstränge, in denen die Begriffe Komplexität und Skalierbarkeit eine entscheidende Rolle

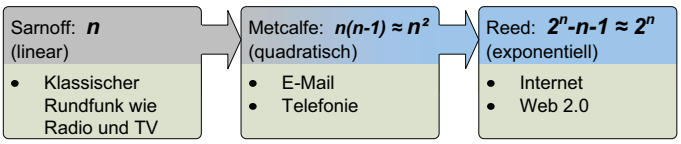


Abbildung 1.1.: Nutzwert von Netzwerken am Beispiel des Paradigmenwandels in der Telekommunikation

spielen. Auf der einen Seite werden Entwicklungen im Bereich der Telekommunikation und des Internets untersucht und neue Lösungsansätze vorgestellt. Andererseits setzt sich die Arbeit mit aktuellen Trends und Problemen bzgl. des Entwurfs digitaler integrierter Systeme mit dem Ziel einer flexiblen und skalierbaren Chip-internen Kommunikationsinfrastruktur auseinander, da derartige Systeme in der Telekommunikation verbreitet zum Einsatz kommen.

Telekommunikationsnetze wachsen zusehends zusammen. Das Internet bietet dafür die gemeinsame Basis. Es erfordert jedoch in vielen Bereichen zusätzliche Anpassungen, um der neuen Rolle als Massenmedium genügen zu können. Die Konvergenz der Medien führt zu einem komplexen Zustandsraum im Internet, in welchem sich gleichermaßen das Reedsche Gesetz widerspiegelt (siehe Abbildung 1.1). Teilnehmerzugangsnetze repräsentieren dabei die zunehmend intelligentere Schnittstelle zum Internet, um die Heterogenität der Dienste und Technologien sowie die hohe Anzahl der Nutzer im homogenen Kernbereich des Internets zu vereinen. Deshalb werden neue Konzepte und Methoden im Gebiet Ethernet-basierter Teilnehmerzugangsnetze für das Internet untersucht und vorgestellt, um den veränderten Anforderungen an das aktuelle und zukünftige Internet entsprechen zu können.

Eine ähnliche Entwicklung in ihrer Komplexität zeigen digitale integrierte Schaltkreise und Systeme. Integrationsdichte und Taktraten steigen. Die integrierte Funktionalität wird umfangreicher. Jedoch steigt die Produktivität der Entwurfswerkzeuge und Verfahren nicht in gleichem Maße, wodurch das Leistungspotential derartig hoch integrierter Systeme nicht mehr ausgeschöpft wird und die Realisierung nicht mehr in ökonomischer Weise durchgeführt werden kann. Dieses Missverhältnis wird in der International Technology Roadmap for Semiconductors [ITRS] als Design-Produktivitätslücke (Design Productivity Gap, DPG) bezeichnet: „[...] Cost (of design) is the greatest threat to continuation of the semi-conductor roadmap [...] the number of available transistors grows faster than the ability to meaningfully design them [...]“. Aus diesem Grund werden neue Ansätze zum Umgang mit Komplexität im Entwurf digitaler Schaltkreise untersucht. Diese Arbeit fokussiert sich dabei insbesondere auf Chip-interne Kommunikationsinfrastrukturen.

Beide Themengebiete der Arbeit stehen unterdessen in direktem Zusammenhang miteinander.

Die Entwicklungen in der Telekommunikationsbranche fallen direkt auf elektronische Datenverarbeitung zurück und führen zu einem immensen Bedarf an leistungsstarker paketverarbeitender Hardware. Hyperkonnektivität und wachsende Datenraten fordern ihren Tribut. Aus diesem Grund werden in vergleichender Weise ausgewählte Mechanismen des ersten Themenstrangs exemplarisch auf eine im zweiten Themenbereich entwickelte Network-on-Chip-Architektur abgebildet.

Überblick über das Dokument Die vorliegende Arbeit ist in 4 Abschnitte unterteilt. Diese Struktur ist in Abbildung 1.2 gezeigt. Blau hervorgehobene Kapitel enthalten wesentliche eigene Beiträge.

- Abschnitt I dient der Einführung in die Thematik. Kapitel 2 rekapituliert dazu notwendige Grundlagen im Bereich der Netzwerke, Paketverarbeitung und Protokolle, welche im weiteren Verlauf der Arbeit zur Anwendung kommen. Kapitel 3 bereitet den aktuellen Ist-Zustand in der Telekommunikationsbranche auf, diskutiert Problemstellungen und leitet Anforderungen für weitere Entwicklungen ab, um die Notwendigkeit der in den Abschnitten II und III vorgestellten eigenen Beiträge und Ansätze zu begründen.

- Abschnitt II beschäftigt sich mit neuen Mechanismen im Bereich der Teilnehmerzugangsnetze. Kapitel 4 stellt einen Mechanismus zur Umsetzung von Ethernet-MAC-Adressen vor, welcher vor allem die Probleme der Skalierbarkeit und Sicherheit in Ethernet-basierten Teilnehmerzugangsnetzen adressiert. Eine prototypische Realisierung des Systems wird bzgl. seiner Leistung bewertet. Im Gegensatz dazu widmet sich der neuartige IPclip-Mechanismus in Kapitel 5 den Problemen der Vertrauenswürdigkeit allgemein und der Identifikation der physikalischen Teilnehmerleitung im Internet.

- Getrieben durch den Bedarf nach performanten und vor allem auch skalierbaren Hardwarelösungen für die Paketverarbeitung bei steigenden Datenraten und zunehmender Systemkomplexität werden in Abschnitt III Aspekte von on-Chip Kommunikations- und Verbindungsstrukturen für digitale integrierte Schaltungen behandelt. Kapitel 6 diskutiert verschiedene Ansätze und schlägt eine ressourcensparende und skalierbare Network-on-Chip-Infrastruktur vor. In Kapitel 7 wird die Anwendbarkeit dieser Kommunikationsarchitektur anhand der in Kapitel 4 vorgestellten Beispielanwendung diskutiert.

- Abschnitt IV schließt die Arbeit ab. In den Kapiteln 8 und 9 werden die wesentlichen Ergebnisse der Arbeit zusammengefasst, bestehende Probleme diskutiert und ein Ausblick auf zukünftige Forschungsrichtungen gegeben.

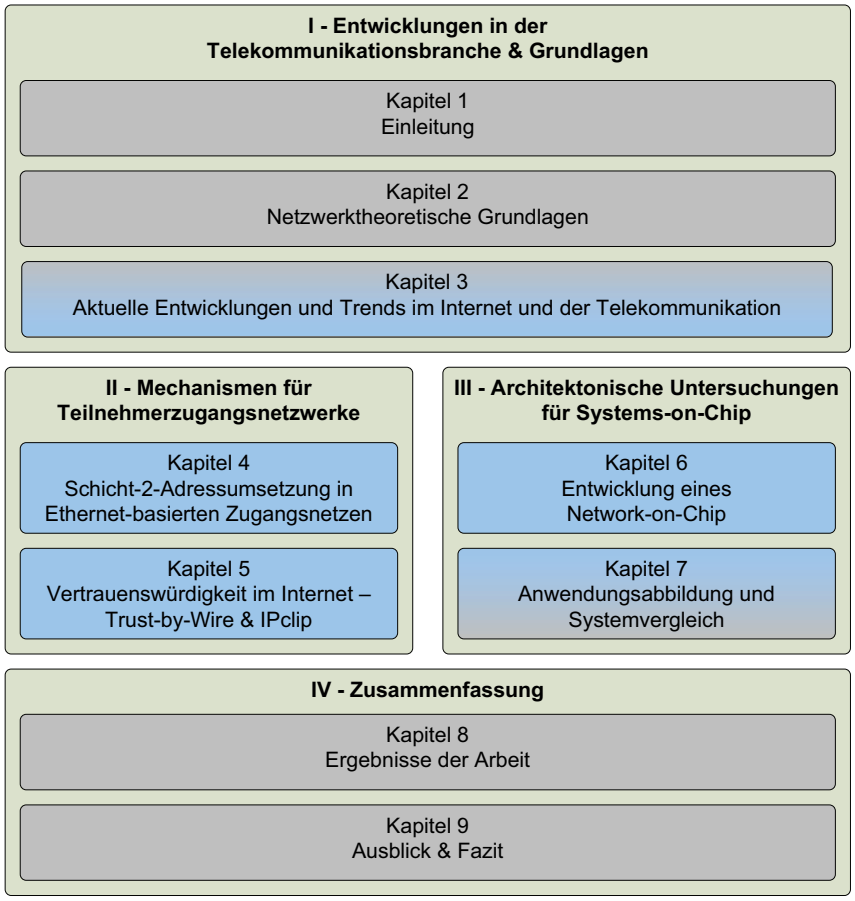


Abbildung 1.2.: Aufbau und Struktur der Arbeit: Abschnitte, die wesentliche eigene Beiträge enthalten, sind blau hervorgehoben.

Copy from one, it's plagiarism;
copy from two, it's research.
(Wilson Mizner, amerik. Dramaturg & Erzähler)

Kapitel 2.

Netzwerktheoretische Grundlagen

Kapitelstruktur

Angefangen vom „Netz der Netze", dem Internet, bis zu vergleichsweise winzigen integrierten Networks-on-Chip werden in dieser Arbeit Kommunikationsnetze verschiedenster Art behandelt. Dieses Kapitel gibt einen Überblick über wesentliche netzwerktheoretische Grundlagen sowie weiterführende Literatur.

2.1. Informationsverarbeitende Systeme

Im privaten sowie beruflichen Alltag existieren verschiedenste Formen von Netzwerken. Um globale Interoperabilität zwischen heterogenen als auch innerhalb lokaler Netzwerke zu gewährleisten, sind gemeinsame Richtlinien und Spezifikationen für den Kommunikationsablauf und die technischen Aspekte der Informationsübertragung notwendig. Dabei geht es um eindeutige Festlegungen bezüglich der Art und Struktur von Informationen sowie der Rolle von Zeit und Raum, welche nach [Bre02] die Fragen des *Was?*, *Wann?* und *Wo?* in Netzwerken aufwerfen:

Was? Wie sehen Informationen aus, die kommuniziert werden, und wie müssen sie für eine einheitliche Kommunikation strukturiert sein?

Wann? Zu welchem Zeitpunkt, wie schnell und vor allem in welcher Abfolge findet eine Kommunikation statt?

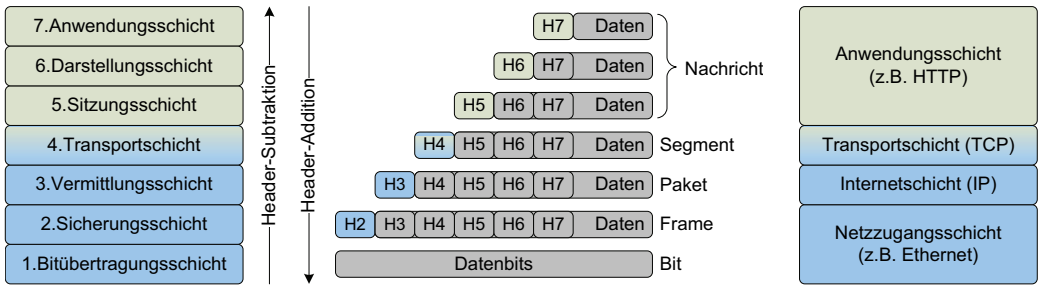
Wo? Welche Geräte, Kanäle und Medien sind an einer Kommunikation beteiligt, um bestimmte Distanzen sowohl effektiv als auch auf effiziente Art und Weise zu überbrücken?

Über mehrere Entwicklungsstufen [Met73, Zim80] wurde deshalb eine Rahmenstruktur vorgeschlagen, welche von der Internationalen Organisation für Normung (ISO) spezifiziert wurde [ISO94]. Das *Open Systems Interconnection (OSI) Basic Reference Model* stellt keinen industriellen Standard dar, sondern beschreibt in Form von Schichten formale Richtlinien für den Aufbau und die Funktionsweise von Kommunikationsnetzen. Eine Schicht bietet der jeweils übergeordneten Schicht Dienste an und nutzt Dienste der untergeordneten Schicht. Abbildung 2.1a zeigt die Anordnung der Schichten. Die für die Arbeit relevanten Schichten sind kurz erläutert.

Schicht 2 – Sicherungsschicht In der Sicherungsschicht werden Netzwerkteilnehmer anhand physikalischer Adressen identifiziert. Weitere Aufgaben sind die Datenvermittlung und Flusskontrolle, Fehlererkennung und -behebung.

Schicht 3 – Vermittlungsschicht Diese Schicht verwendet logische Adressen. Primäre Aufgabe ist die Wegewahl zwischen Kommunikationsendpunkten über das gesamte Netzwerk hinweg.

Schicht 4 – Transportschicht Aufgaben der Transportschicht sind die Ende-zu-Ende-Kontrolle zwischen logischen Kommunikationsendpunkten und die Bereitstellung einer einheitlichen Schnittstelle zu den höheren anwendungsorientierten Schichten.



(a) Schichten des OSI Referenzmodells, Kapselung von Nutzdaten und Protokoll-Headern **(b)** TCP/IP Protokollstack

Abbildung 2.1.: OSI Referenzmodell und TCP/IP Protokollstack (transportorientierte Schichten sind blau, anwendungsorientierte beige hinterlegt)

Informationen durchlaufen jede einzelne Schicht in Form abgeschlossener Dateneinheiten (Protocol Data Unit, PDU), welche definierte Strukturen aufweisen. Im Allgemeinen bestehen PDUs aus einem Kopf (Header), Nutzdaten (Payload) und in einigen Fällen einem zusätzlichen Anhang (Trailer). Inhalte von Header und Trailer sind abhängig vom jeweiligen Protokoll und werden als Protokoll-Overhead bezeichnet. Im Gegensatz zum OSI Referenzmodell, welches nur ein konzeptuelles Gerüst bereitstellt, definieren Protokolle die genauen Verfahrensweisen und Methoden zur Regelung der Kommunikation und definieren die Struktur der PDUs [Jav07]. Header, Nutzdaten und Trailer eines Protokolls auf Schicht n werden dabei in den Nutzdaten auf Schicht $n-1$ gekapselt. Abbildung 2.1a zeigt diese Hierarchie.

Eine bedeutende Implementierung eines OSI-konformen Netzwerkes ist die TCP/IP Protokollfamilie, zu der mittlerweile ca. 500 Protokolle für Kommunikations- und Verwaltungsaufgaben im Internet gehören. Das Internet Protokoll (IP) und das Transmission Control Protocol (TCP) sind die Basisprotokolle des Internets. Abbildung 2.1b stellt die 4 Schichten des TCP/IP Referenzmodells dem allgemeinen OSI Referenzmodell gegenüber. Insbesondere werden die höheren, anwendungsbezogenen Schichten in einer einzigen Anwendungsschicht zusammengefasst.

Für zusätzliche Informationen zum OSI Referenzmodell sei auf die genannte Literatur als auch auf [PD04], [HB05] und [Hal05] verwiesen.

2.2. Computernetzwerke

Computernetzwerke verbinden autarke, elektronische Systeme miteinander und ermöglichen den Austausch von Daten und Informationen. Das OSI Referenzmodell bietet dafür den geeigneten Rahmen. Nach [PD04] sind die Hauptaufgaben eines Computernetzes, Konnektivität zwischen einzelnen Geräten bereitzustellen, die Nutzung gemeinsamer Ressourcen zu ermöglichen und zu kontrollieren sowie Dienste und einheitliche Schnittstellen bereitzustellen.

In der Praxis haben sich verschiedene Ausprägungen von Netzwerken herauskristallisiert. Dazu zeigt Abbildung 2.2 eine Einordnung der verschiedenen Netzwerkklassen anhand ihrer typischen geographischen Ausdehnung und Datenraten[1].

Lokale Netze Local Area Networks (LAN) sind die verbreitetste Form von Computernetzwerken. Ein LAN verwendet üblicherweise nur eine Übertragungstechnologie und besteht weitestgehend aus Netzwerkendpunkten.

Nicht-lokale Netze Zu den nicht-lokalen Netzwerken werden alle Weitverkehrsnetze wie Metropolitan, Wide und Global Area Networks (MAN, WAN, GAN) gezählt. Sie bestehen aus

[1] Im weiteren Verlaufe der Arbeit wird diese Grafik noch erweitert werden.

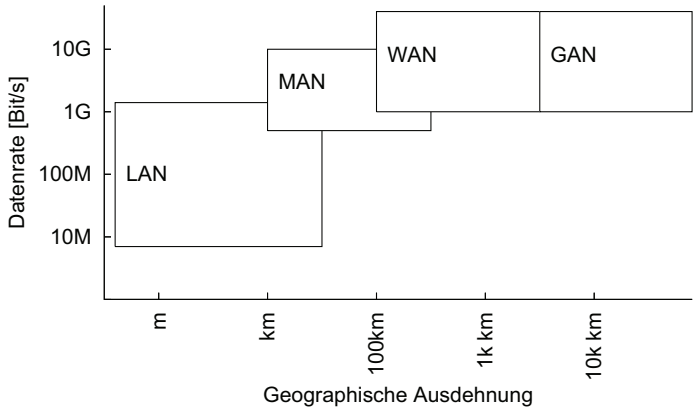


Abbildung 2.2.: Einordnung unterschiedlicher Netzwerkklassen bezüglich ihrer geographischen Dimensionen und typischer Datenraten

Netzwerkknotenpunkten zur Verknüpfung hierarchisch untergeordneter oder lokaler Netze. Ein MAN erstreckt sich über einen kompletten städtischen Raum. WANs hingegen verbinden mehrere MANs miteinander und können sich landesweit oder auch über Kontinente erstrecken. Die oberste Hierarchiestufe stellen GANs oder sogenannte Internetzwerke dar, z. B. das BGAN von Inmarsat [Inm], die auf globaler Ebene mehrere WANs zusammenführen. Ein Internetzwerk verbindet unabhängig voneinander verwaltete Teilnetze, Netze verschiedener Betreiber oder Netze unterschiedlicher Basistechnologien miteinander.

Spezialformen Besondere Formen von Rechnernetzen sind Virtuelle LANs (VLANs) und Virtuelle Private Netze (VPNs). Dies sind logische Netze, welche über der tatsächlichen Netzwerkinfrastruktur aufgespannt sind, um z. B. Netzwerkendpunkte logisch zu gruppieren, den Datenverkehr effizienter zu steuern oder auch um geographisch entfernte Teilnetze zu einem logischen Gesamtnetz zusammenzufassen. Weitere Spezialformen von Netzwerken sind das *Internet*, *Teilnehmerzugangsnetzwerke* (TZN) und auch *Networks-on-Chip* (NoCs). Die zuletzt Genannten werden in den entsprechenden Kapiteln dieser Arbeit noch detailliert betrachtet.

In [Cis01, PD04, Tan03, Jav07] können weitere Informationen rund um das Thema der Computernetzwerke nachgeschlagen werden.

2.3. Paketverarbeitung in Netzwerken

Im Bereich der Kommunikations- und Netzwerktechnik werden unter dem Begriff *Paketverarbeitung* (Packet Processing bzw. Protocol Processing) alle Prozesse, Aktionen und Maßnahmen

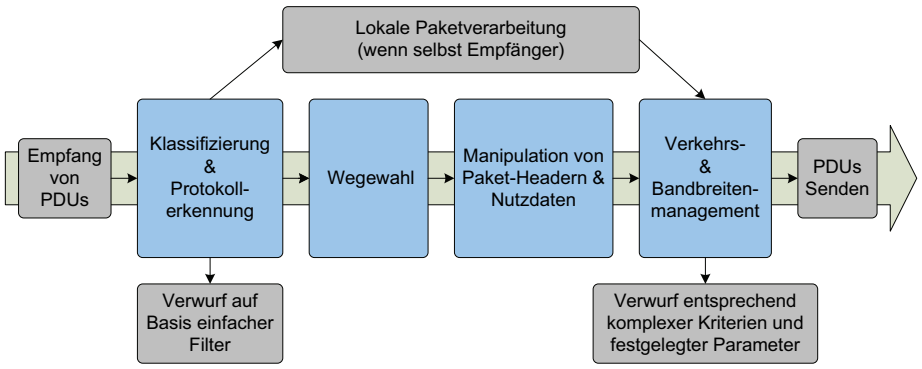


Abbildung 2.3.: Typische, logische Abfolge von Prozessen der Paketverarbeitung

zusammengefasst, welche in Netzwerkend- und Netzwerkknotenpunkten ausgeführt werden, um Daten zu senden, zu empfangen und zu verarbeiten.

Abbildung 2.3 zeigt die typische, logische Abfolge von Aufgaben in der Paketverarbeitung. Nachdem PDUs einen Klassifizierungsprozess durchlaufen haben, wird das Klassifizierungsergebnis zur weiteren Wegewahl genutzt, wenn die PDUs nicht bereits durch erste einfache Filter verworfen worden sind. Teile der PDUs können dabei verändert werden. Bevor sie dann über den gewählten Ausgangskanal weitergesendet werden, finden verschiedene Mechanismen zur Gewährleistung vereinbarter Dienstgüten Anwendung. Die für den weiteren Verlauf der Arbeit interessanten Schritte sollen kurz erläutert werden.

Paketklassifizierung & Protokollerkennung Die Klassifizierung eines Paketes und der gekapselten Protokolle ist der erste Schritt in der Paketverarbeitung [GM99, GM01]. Dazu werden relevante Informationen, sogenannte Schlüssel (Keys), aus den PDUs extrahiert. Diese Tupel werden zur Suche nach Regeln und Richtlinien (Rules & Policies) genutzt. Weiterführende Aufgaben, z. B. Wegewahl oder Bandbreitenmanagement, benötigen das Ergebnis des Klassifizierungsvorgangs als Entscheidungskriterium (Trigger). Die Klassifizierung war bis vor kurzem nicht im jetzigen Umfang erforderlich, da im Internet und anderen Netzwerken ausschließlich minimalistische Dienstgüten zugesichert wurden, die unter *Best-Effort* bekannt waren. Zur Bereitstellung individueller Dienste und abgestufter Dienstgüten, was mit *Quality-of-Service* (QoS) bezeichnet wird, ist jedoch die Extraktion komplexer Tupel an Informationen auf allen Protokollschichten notwendig.

Wegewahl Die Wegewahl (Routing & Switching) beschreibt die Ermittlung des Ausgangskanals für eine PDU in Richtung des Zielknotens. Dies ist die wichtigste Funktion in einem Kommunikationsnetzwerk und dient dem korrekten Weiterleiten von Daten. Es wird zwischen

Switching und Routing unterschieden. Reine Switche nutzen die physikalischen Adressen der Sicherungsschicht zum Weiterleiten von z. B. Ethernet-Frames innerhalb *desselben* Netzwerks. Router hingegen nutzen die logischen Adressen der Vermittlungsschicht, um z. B. IP-Pakete *zwischen verschiedenen* Netzwerken zu vermitteln, welche zudem auch auf unterschiedlichen Übertragungstechnologien basieren können. Auf beiden Schichten werden üblicher Weise Adresstabellen gepflegt, die für jede Quelladresse oder Gruppen von Quelladressen den entsprechenden Ausgangskanal enthalten. Als Beispiel zeigt Abbildung 2.4 den Aufbau einer einfachen IP Routing-Tabelle. Ein ähnliches Ergebnis liefert u. a. der Befehl `route PRINT` der Kommandozeile von Windows Betriebssystemen. Mithilfe von Routing-Algorithmen werden derartige Tabellen erstellt. Die Einträge werden dann zum eigentlichen Weiterleiten genutzt.

```
 Destination   Address          Next-Hop         Interface
 Address       Mask             Address          Number
-------------------------------------------------------------
 192.5.48.0    255.255.255.0    128.210.30.5     2
 128.10.0.0    255.255.0.0      128.210.141.12   1
 0.0.0.0       0.0.0.0          128.210.30.5     2
 ...
```

Abbildung 2.4.: Beispiel einer einfachen IP-Routing-Tabelle

Verkehrs- & Bandbreitenmanagement Individuelle Dienstgüten bzw. QoS, wie z. B. zur Verfügung gestellte minimale oder maximale Bandbreiten, sind vertraglich zwischen Endkunden und Dienstanbietern geregelt. Zudem erfordern viele Dienste die Einhaltung von Randbedingungen und Parametern, um zu funktionieren, z. B. bestimmte zulässige Verzögerungszeiten. Diese Regeln werden auch als Service-Level-Agreements (SLAs) bezeichnet. Unter Verkehrs- und Bandbreitenmanagement werden Mechanismen zur Gewährleistung von SLAs und zur Einhaltung notwendiger Verkehrsparameter verstanden. Dazu zählen das Messen des Datenverkehrs (Metering), das Vergleichen mit definierten SLAs und entsprechendes Markieren von PDUs (Policing) sowie die Kontrolle des Datenflusses (Shaping) in Bezug auf die Ergebnisse des Policing. Zur Datenflusskontrolle zählen das Puffern (Buffering) sowie das Verwerfen von PDUs (Dropping). Dazu werden in den Speichern meist mehrere logische Warteschlangen verwaltet (Queueing), um separate Datenströme zu priorisieren oder zu blockieren. Umfassende theoretische Grundlagen zu QoS bietet [BT04, Par05].

Bezüglich der Paketverarbeitung werden heutzutage strikte Anforderungen an die verwendeten Geräte gestellt. Waren vor einigen Jahren noch die Bandbreiten der Übertragungstechnologien der limitierende Faktor im Internet, so sind es heute oft die Netzwerkknotenpunkte selbst, da diese eine Vielzahl physikalischer Übertragungskanäle bündeln und gleichzeitig tausende logische

Kommunikationsverbindungen verwalten müssen. Daher lassen sich allgemeine Anforderungen ableiten, durch die moderne Netzwerkgeräte charakterisiert sind. Primäre Attribute sind kurz aufgeführt. Zusätzliche Informationen zu Aspekten der Paketverarbeitung können z. B. [Gri01, Cha03, Foa04, Var05] oder [PM04] entnommen werden.

Flexibilität bezeichnet vor allem die Programmierbarkeit und Skalierbarkeit eines Geräts, z. B. die Ausrichtung existierender Funktionen an Kundenwünschen oder die Möglichkeiten der Integration gänzlich neuer Features.

Blockierungsfreiheit bedeutet, dass trotz der hohen und wachsenden Datenvolumina (siehe Kapitel 1) keine PDUs verworfen werden oder zu hohe Latenzen entstehen.

Verfügbarkeit und Zuverlässigkeit sind notwendige Eigenschaften der Netzwerkinfrastruktur, da Netzwerke integraler Bestandteil von Wirtschaft und Gesellschaft sind.

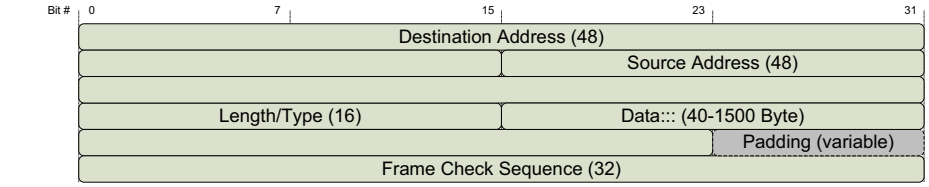
2.4. Ethernet

In den letzten Jahren hat sich die Übertragungstechnologie Ethernet zur vorherrschenden und erfolgreichsten Technologie entwickelt. Ethernet findet sowohl in lokalen und flächendeckenden Netzwerken als auch im Internet Einsatz. Der Begriff Ethernet steht mittlerweile nicht mehr nur für eine einzige Übertragungstechnologie, sondern fasst eine Vielzahl von Ausprägungen, Erweiterungen und Spezifikationen zusammen. Die aktuelle Version des Standards von 2005 ist in IEEE Std 802.3 [IEE05] als Teil der Arbeitsgruppe IEEE 802 für lokale Netz [IEE02] verfügbar. Der IEEE Std 802.3 definiert Teile der Bitübertragungsschicht und Sicherungsschicht des OSI Referenzmodells. Details zu den Aufgaben der einzelnen Unterschichten können den Standards [IEE98, IEE05] entnommen werden.

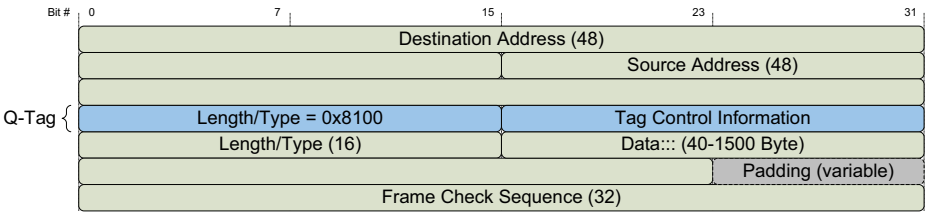
Ethernet-Frames Die allgemeine Struktur einer Ethernet-PDU, eines Frames bzw. Rahmens, ist in allen Ethernet-Varianten gleich. Abbildung 2.5a zeigt den typischen Aufbau.

Destination & Source Address Die physikalischen Schnittstellen von Netzwerkteilnehmern werden mithilfe von 48 Bit breiten MAC-Adressen identifiziert. Sie müssen innerhalb eines abgeschlossenen Netzes eindeutig sein.

Length/Type Dieses 2-Byte Feld gibt die Länge des Frames an, wenn der Wert kleiner gleich `0x0600` (hexadezimal) ist. Ist der Wert größer als `0x0600`, identifiziert dieses Feld das gekapselte Schicht-3-Protokoll. Ein typischer Wert ist `0x0800`, welcher die Nutzdaten als Internet Protokoll charakterisiert.



(a) Standard Ethernet-Frame nach IEEE Std 802.3 [IEE05]



(b) Tagged Ethernet-Frame nach IEEE Std 802.1Q [IEE06a]

Abbildung 2.5.: Zusammensetzung und Struktur von Ethernet-Frames

Data & Padding Das Datenfeld beinhaltet die Nutzinformationen, die sogenannte Payload des Frames. Um eine minimale Framelänge von 64 Byte zu gewährleisten, ist in einigen Fällen ein Auffüllen mit Nullen (Padding) notwendig.

Frame Check Sequence Die Frame Check Sequence (FCS) umfasst 4 Byte und dient der Fehlererkennung. Die FCS ist eine 32-Bit Prüfsumme, welche mittels Cyclic Redundancy Check (CRC) von der Zieladresse bis zum Ende des Datenfeldes berechnet wird.

Abbildung 2.5b zeigt eine besondere Form von Ethernet-Frames. Sie werden als Tagged Ethernet-Frames bezeichnet, da ein zusätzliches Feld, das sogenannte Q-Tag, eingefügt wird. Q-Tags sind in IEEE Std 802.1Q [IEE06a] standardisiert und ermöglichen die Nutzung mehrerer logischer, virtueller Netze (VLANs) innerhalb desselben Netzwerkes. Mit einem Q-Tag können $2^{12} = 4096$ VLANs differenziert werden. Innerhalb von TZNs finden insbesondere verschachtelte Q-Tags (Q-in-Q, QiQ) nach IEEE Std 802.1ad [IEE06b] Anwendung, um die Zahl möglicher VLANs zu erhöhen. Ein Q-Tag wird durch ein Typfeld mit dem Wert `0x8100` angezeigt. Nach dem letzten Q-Tag folgt das eigentliche Length/Type-Feld des Ethernet-Frames.

Ausführliche Informationen zu IEEE Std 802.3 können in den Standarddokumenten selbst als auch in [Spu00, Hel03, Axe03] recherchiert werden.

Eigenschaften von Ethernet-Netzwerken Aufgrund seiner vielen Vorzüge hat sich Ethernet zur Übertragungstechnik der Wahl entwickelt. Trotz dieser Vorteile hat Ethernet aber auch Grenzen und Nachteile. Das Gesamtpaket Ethernet bietet jedoch im Vergleich mit alternativen

Technologien wie ATM (Asynchronous Transfer Mode) den besseren Kompromiss. Im Folgenden sind die wichtigsten Eigenschaften beschrieben.

Ethernet ist vielseitig. Ethernet bietet mit seinen vielen Ausprägungen zugeschnittene Lösungen für verschiedene Anwendungsgebiete. Mittlerweile sind Bandbreiten von 10 Mbit/s bis zu 10 Gbit/s möglich. Zurzeit erfolgt die Standardisierung von 100 Gbit/s-Ethernet [IEEb, Por07].

Ethernet ist einfach. Es ist keine umfangreiche Konfiguration erforderlich, um einen Rechner an ein Ethernet anzuschließen. Der Zugriff auf das jeweilige Medium wird alleine durch die Hardware gesteuert, welche durch Bitübertragungs- und Sicherungsschicht einheitlich spezifiziert ist. Durch seine Einfachheit ist Ethernet zudem sehr verlässlich.

Ethernet ist kostengünstig. Eine breite Palette an Produkten verschiedener Hersteller ist verfügbar. Zudem sind die Herstellungskosten für Kabel oder Netzwerkkarten gering. Insbesondere im Bereich der Telekommunikation ist ein geringer Preis pro Port entscheidend. Für lokale Netzwerke existieren durchaus günstigere Technologien, z. B. Universal Serial Bus (USB) oder I^2C, jedoch sind diese auf bestimmte Anwendungsgebiete spezialisiert. Ethernet kann in einem deutlich größeren Rahmen eingesetzt werden.

Ethernet überbrückt große Distanzen. Verdrillte Kupferdrahtleitungen ermöglichen Entfernungen bis ca. 100 m. Optische und drahtlose Medien und Technologien sowie spezielle Netzwerkelemente können die Ausdehnung eines Ethernets zusätzlich erweitern.

Ethernet ist nicht echtzeitfähig. Bestimmte Anwendungen erfordern Echtzeitfähigkeit. Sie verlassen sich auf ein garantiertes Zeit- und Sendeverhalten. Dies ist bei nativem Ethernet aufgrund des CSMA/CD-Mechanismus *nicht* gegeben. Allerdings sind bei geringer Kanalauslastung und nur wenigen Kollisionen die Verzögerungen relativ gering. Insbesondere für industrielle Anwendungsgebiete bieten auf Ethernet aufbauende Ansätze wie EtherCAT oder Ethernet-Powerlink eine Lösung.

Ethernet ist ineffizient. Besonders bei minimalen und kleinen Frames weist Ethernet ein ineffizientes Verhältnis zwischen Nutzinformationen und Steuerinformationen auf. Pro Ethernet-Frame werden zusätzlich bereits 20 Byte für Inter Frame Gap, Präambel und Start Frame Delimiter benötigt [IEE05] (siehe auch Anhang C.3). Bei normaler Auslastung des Netzes – dies bedeutet bei Ethernet für gewöhnlich weniger als 30 % [PD04] – ist dieser Overhead aufgrund der Geschwindigkeit des Mediums jedoch vernachlässigbar. Bei höherer Auslastung steigt die Zahl der Kollisionen, was dazu führt, dass viele Frames erst nach mehrmaligem Versuch vollständig gesendet werden können. Damit sinkt die Effizienz wieder.

Ethernet hat einen hohen Energieverbrauch. IEEE Std 802.3 definiert keine Energiesparmodi für die Ethernet-Schnittstellen. Somit wird im Ruhebetrieb die gleiche Menge an Energie verbraucht wie während einer Übertragung [PP08]. Aufgrund des verbreiteten Einsatzes der Ethernet-Technologie werden so immense Kosten verursacht. Die Energy Efficient Ethernet (EEE) Study Group [IEEa] befasst sich mit der Entwicklung von Mechanismen zur Reduzierung des Energieverbrauchs während Perioden geringer Leitungsaktivität, z. B. durch erneute Aushandlung einer geringeren, angemessenen Übertragungsgeschwindigkeit. In [GCNS08, NPI+08] sind dazu Ansätze zur adaptiven Anpassung der Datenraten vorgestellt.

2.5. Internet Protokoll

Wie der Name bereits impliziert ist das Internet Protokoll (IP) [RFC0791] das zentrale Protokoll des Internets. IP ordnet sich auf der Vermittlungsschicht des OSI Referenzmodells ein bzw. repräsentiert alleine die Internetschicht im TCP/IP Protokollstack, wie Abbildung 2.6 zeigt. Diese Darstellung wird auch als IP-Stundenglas bezeichnet und verdeutlicht die zentrale Stellung von IP. Verschiedenste Protokolle und Dienste werden innerhalb IP gekapselt während IP gleichzeitig unabhängig von der zugrunde liegenden Übertragungstechnologie ist. Die wichtigsten Aufgaben von IP sind dabei die Verknüpfung unabhängiger Netze sowie das Weiterleiten von IP-Paketen innerhalb eines Netzes und zwischen verschiedenen Netzen, die Bereitstellung von (logischen) Adressen der Vermittlungsschicht, sowie die Zustellung von Daten auf Basis pauschaler, minimalistischer Dienstgüten (Best-Effort). Deshalb wird das zustandslose IP netzwerk- und technologieübergreifend zur Bereitstellung von Ende-zu-Ende-Konnektivität zwischen Kommunikationsendpunkten eingesetzt. In IP spiegeln sich die grundlegenden Prinzipien und Charakteristika des Internets wieder, welche u. a. in [RFC1958] beschrieben sind. Um weiteren Erklärungen zuvorzukommen, soll an dieser Stelle ein Auszug daraus helfen: „[...] in very general terms, the community believes that the goal is connectivity, the tool is the Internet Protocol, and the intelligence is end to end rather than hidden in the network."

IP-Datagramme Abbildung 2.7a stellt die Struktur eines IPv4-Datagramms dar. Im Folgenden wird die üblichere Bezeichnung IP-Paket genutzt. Für die Arbeit interessante Felder des IPv4-Headers sind kurz erläutert. Weitere Details können [RFC0791] und [Jav07] entnommen werden.

VER Dieses Feld gibt die IP-Version wieder (4 = IPv4, 6 = IPv6).

IHL Die Internet Header Length (IHL) beschreibt die Länge des IP Headers in 32-Bit-Worten.

Protocol Number Dieses Feld identifiziert das Schicht-4-Protokoll, z. B. TCP = `6`.

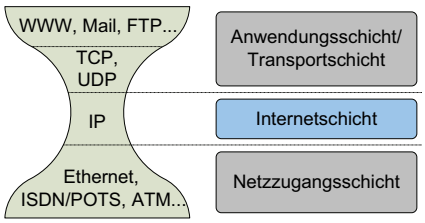
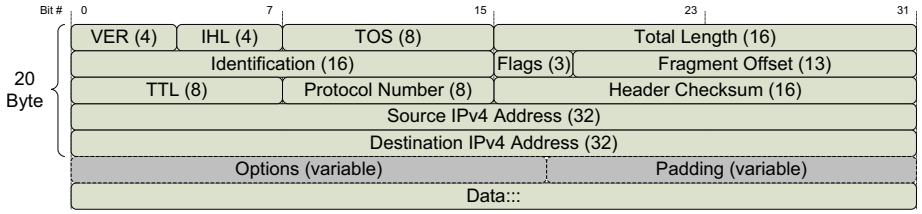
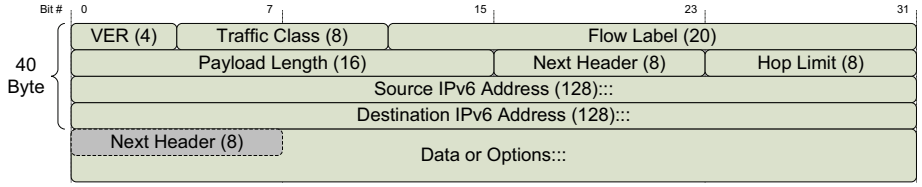


Abbildung 2.6.: Das Internet Protokoll als Bindeglied im sogenannten IP-Stundenglas sowie Gegenüberstellung mit dem TCP/IP Protokollstack (vgl. Abschnitt 2.1)



(a) Aufbau eines IPv4-Pakets



(b) Aufbau eines IPv6-Pakets

Abbildung 2.7.: IP-Paketformate

Source & Destination Diese je 32 Bit großen Felder enthalten die IP-Adressen der Kommunikationsendpunkte. IP-Adressen sind logische Adressen.

Options & Padding Optionen enthalten Zusatz- oder Kontrollinformationen und müssen von jedem IP-fähigen Netzwerkgerät syntaktisch verstanden werden können. Optionen nutzen i. Allg. ein Optionsformat, welches sich aus Typ (1 Byte), Länge (1 Byte) und Optionswert (Value) zusammensetzt und als TLV-Struktur (Type-Length-Value) bezeichnet wird.

Data Dem IP-Header und möglichen Optionen folgen die eigentlichen Nutzdaten.

Internet Protokoll Version 6 Aufgrund aktueller Entwicklungen, welche in Kapitel 3 im Detail dargestellt werden, weist IPv4 mittlerweise diverse Schwachpunkte auf. Dazu gehört vor allem

die Verknappung des Adressraumes[2]. Dies ist einerseits durch die Größe von „nur“ 2^{32} Adressen und andererseits durch die Einteilung in Klasse-A, -B, -C und -D Subnetze und deren ineffiziente Ausnutzung begründet. Zudem bietet natives IPv4 keinerlei Sicherheitsmechanismen und eignet sich nur bedingt für die verbreitet zum Einsatz kommenden Ausprägungen von Echtzeit- und Multimediaanwendungen.

Aufgrund dieser Nachteile wurden bereits frühzeitig Randbedingungen für eine Weiterentwicklung definiert [RFC1752]. Diese resultierten letztendlich im Internet Protokoll Version 6 (IPv6) [RFC2460], welches nun nicht mehr die Nachteile seines Vorgängers aufweist. Abbildung 2.7b zeigt den Aufbau eines IPv6-Paketes. Die augenscheinlichsten Veränderungen sind der vereinfachte IPv6-Header, welcher nun weniger Felder umfasst, sowie die Vergrößerung des Adressraumes von 2^{32} (IPv4) auf 2^{128} (IPv6). Die Handhabung von Optionen wurde durch die Nutzung von Erweiterungsheadern (Extension Header) vereinfacht.

Für weitere Details zum Gebrauch von IPv6 und Neuerungen wie Autokonfiguration von Adressen und Integration von Sicherheitsfeatures sei auf die Literatur verwiesen [RFC2460, RFC2464, Hag06, LJS06].

2.6. Zusammenfassung des Kapitels

Dieses Grundlagenkapitel diente dem Einstieg in das Gebiet der Kommunikationsnetzwerke und der Erläuterung grundlegender Bezeichnungen und Fachbegriffe. Abschnitt 2.1 bot einen Abriss über das allgemeine Modell informationsverarbeitender Systeme. Es bildet die Basis beider Themenstränge dieser Arbeit. Typische Ausprägungen von Computernetzen wurden in Abschnitt 2.2 erläutert. Hier wurden bereits Teilnehmerzugangsnetze und Networks-on-Chip als spezielle Formen von Netzwerken erwähnt. Beide werden in den jeweiligen Kapiteln noch genauer behandelt. Abschnitt 2.3 stellte daraufhin die typischen Formen sowie Charakteristika der Paketverarbeitung in Netzwerken dar. Letztendlich wurden in den Abschnitten 2.4 und 2.5 die heutzutage vorherrschenden Schicht-2- und Schicht-3-Protokolle vorgestellt – Ethernet und IP. Beide haben sich aufgrund ihrer Eigenschaften im Internet und der Telekommunikation manifestiert und sind deshalb eng mit dem Inhalt dieser Arbeit verknüpft.

[2] siehe z. B. `http://www.heise.de/newsticker/meldung/99902`

People tend to overestimate what can be done in one year
and to underestimate what can be done in five or ten years.
(Joseph Carl Robnett Licklider, amerik. Psychologe)

Kapitel 3.

Aktuelle Entwicklungen und Trends im Internet und der Telekommunikation

Kapitelstruktur

Dieses Kapitel stellt die Notwendigkeit der in Teil II der Arbeit vorgestellten Mechanismen für Teilnehmerzugangsnetzwerke heraus. Der knappe Abriss über wesentliche Eigenschaften des Internets in Abschnitt 3.1 verweist bereits auf einige Ursachen für die Entwicklungen in der Telekommunikationsbranche. Abschnitt 3.2 analysiert die momentan zu beobachtenden Trends. In Abschnitt 3.3 werden sowohl derzeitige Problemstellungen diskutiert, die sich aus diesen Trends ergeben, als auch resultierende Anforderungen an heutige und zukünftige Netze und Netzkomponenten abgeleitet. Abschnitt 3.4 beinhaltet ein kurzes Resümee.

3.1. Internet

Es gibt Internetzwerke und es gibt *das* Internet. Internetzwerke wurden bereits in Abschnitt 2.2 erläutert. Das Internet ist eine Ausprägung eines Internetzwerks, wie Abbildung 3.1 zeigt. Es ist

ein globales, heterogenes, verteiltes Kommunikationssystem, welches eine Vielzahl verschiedener Teilnetzwerke miteinander verknüpft. Seine Heterogenität besteht dabei in der Vielfalt der Übertragungsmedien, Kommunikationsendpunkte, Datenformate und Protokolle [CDK05].

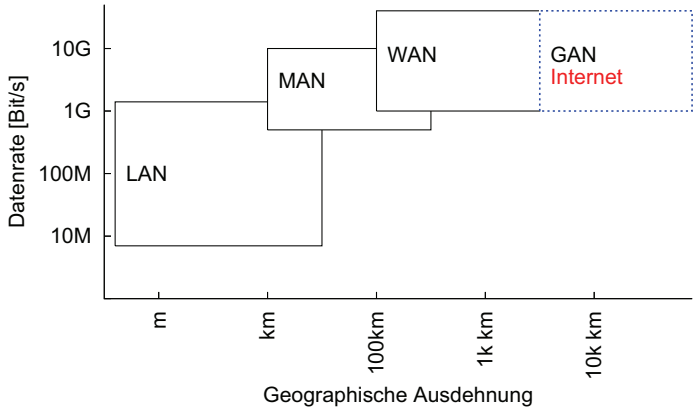


Abbildung 3.1.: Einordnung des Internets in die unterschiedlichen Netzwerkklassen

Abbildung 3.2 zeigt die allgemeine, aus 4 Hauptebenen bestehende Struktur des Internets. Die oberste Hierarchiestufe, das zentrale Backbone, fasst Interkontinentalverbindungen zusammen. Die kontinentalen Backbones stellen die nächste Hierarchiestufe (Ebene 1) dar. Auf Ebene 2 befinden sich nationale und regionale Netzwerke, z. B. ein abgeschlossenes Netz eines Internet Service Provider (ISPs). Der Bereich vom zentralen Backbone bis zu Ebene 2 wird als globales Internetzwerk bezeichnet. *Das* Internet erstreckt sich jedoch bis über Ebene 3, dem Bereich der TZN, welcher in Abbildung 3.2 grau hinterlegt ist. Dort terminiert das Netz letztendlich an den Kommunikationsendpunkten. ISPs pflegen für gewöhnlich sogenannte Peering- oder Transitabkommen, um den Nutzern überregionale Konnektivität gewährleisten zu können. Durch sein sukzessives Wachstum hat das Internet über die Jahre eine skalenfreie Topologie [BB03, BKC08] ausgebildet, die insbesondere robust und flexibel erweiterbar ist[1].

Das Internet ist paketvermittelt und basiert auf IP (siehe Abschnitt 2.5). Es ermöglicht den gemeinsamen Zugriff auf Dienste und Anwendungen, ist geprägt von Parallelität und Simultanität, von Asynchronität sowie von unabhängigen Fehlerpunkten. Die Hauptaufgabe ist die Bereitstellung von Konnektivität und die bestmögliche Übertragung von Daten zwischen den entsprechenden Kommunikationsendpunkten. Verglichen mit klassischen leitungsvermittelten Telekommunikationsnetzen ist das Internet ein „dummes“ Netzwerk. Die Intelligenz befindet sich am Rand in den Terminals [Ise98, RSC98, Gil06]. Zudem gibt es keine übergeordnete Instanz. Die Wartung und Verwaltung des Internets erfolgt indirekt durch die Betreiber (Car-

[1] Unter [Har] ist diese Topologie auf Basis von Städte-Verbindungen anschaulich visualisiert.

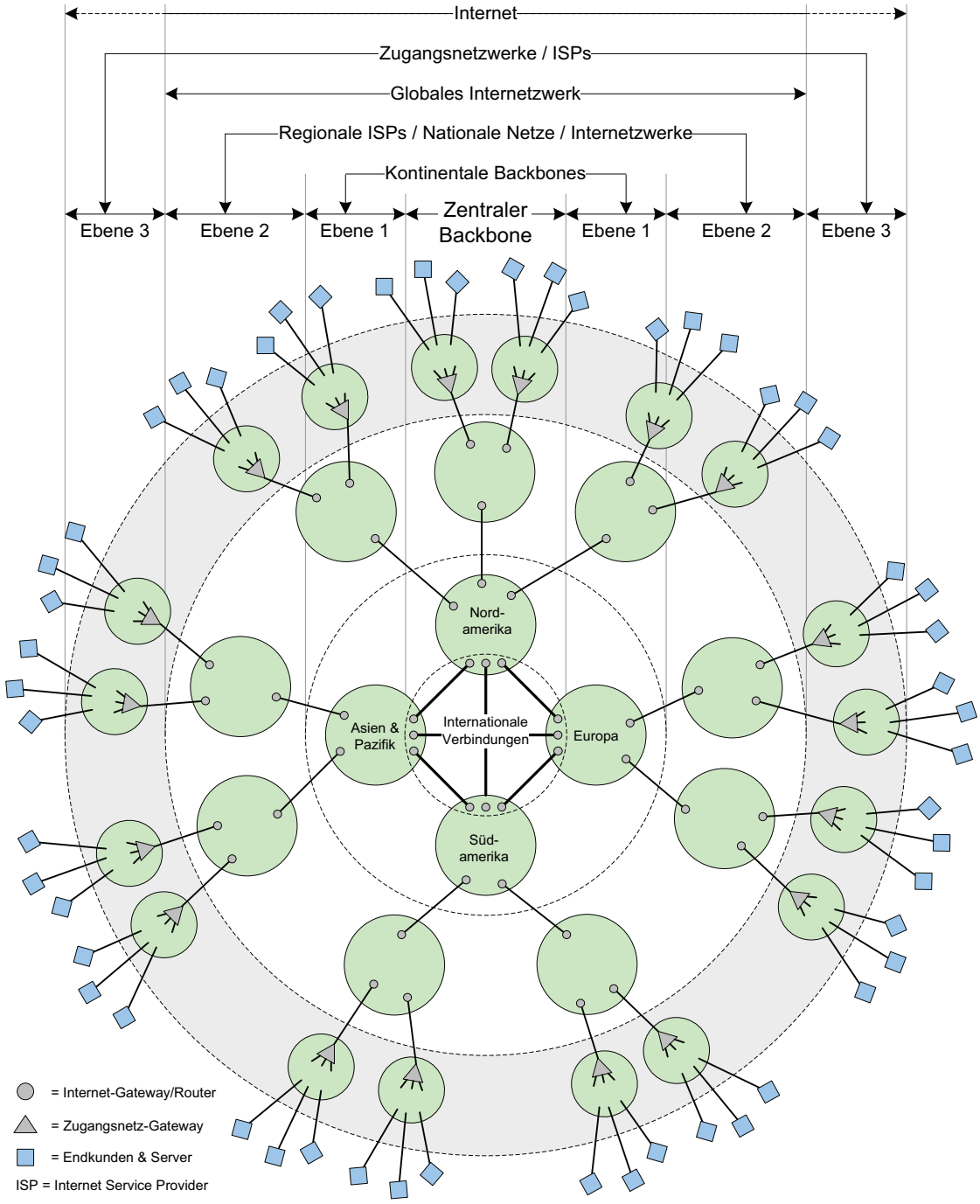


Abbildung 3.2.: Allgemeine, hierarchische Architektur des Internet (nach [Hal05])

rier) regionaler Teilnetze und die ISPs. Non-profit Organisationen wie ICANN, IANA und IETF überwachen die Koordination des Internets. Weitere Charakteristika des Internets sind in [LCC+97, Kle03, Kle04] beschrieben. Zudem bieten [RFC1958, RFC2775, RFC3439] Informationen zu den technischen und architekturbezogenen Eigenschaften des Internets sowie zu den üblichen Verfahren und Mechanismen.

3.2. Aktuelle Trends in der Telekommunikation

Seit einigen Jahren sind nachhaltige Veränderungen auf dem Telekommunikationsmarkt zu beobachten, welche direkten Einfluss auf die verschiedenen Telekommunikationsnetze (Fernsprechnetze, Rundfunknetze, Mobilfunknetze, Datennetze) ausüben. Das Internet und die Kommunikationsindustrie verschmelzen dabei zunehmend [Asc07]. Zu den wichtigsten Ursachen und Gründen dafür zählen u. a.

- die Fortschritte in der akademischen & industriellen Forschung,
- die „Digitale Revolution“ und gesellschaftlichen Entwicklungen [Rus95],
- die Eigenschaften des Internets selbst (siehe Abschnitt 3.1 und angegebene Referenzen),
- die Unterschiede paketvermittelter Netze gegenüber leitungsvermittelten Netzen [SJ06],
- der zunehmende Nutzwert des Internets durch Konnektivität [Nor07a] und das Reedsche Gesetz [SS06].

Die aktuellen Entwicklungen und Trends können in soziale & gesellschaftliche, ökonomische sowie technologische Trends eingeteilt werden. Für die folgenden Darstellungen werden, soweit nicht anders angegeben, Zahlen des Statistischen Bundesamtes [BT07, Sta08], der Bundesnetzagentur [Bun07b, Bun07c], des Bundesverbandes Informationswirtschaft, Telekommunikation und neue Medien (BITKOM) [BI07] sowie aus Analysen von Morgan Stanley [Mor05, Mor07] und Technology Futures, Inc. [Van02] verwendet.

3.2.1. Ökonomische Entwicklungen

Globalisierung & Liberalisierung Die zunehmende internationale Verflechtung wird allgemein als Globalisierung bezeichnet. Während der Globalisierungsprozesse führten bisherige ökologische und politische Entscheidungen mit dem Ziel einer größeren globalen Abdeckung zur Liberalisierung im Welthandel, was sich auch auf die Telekommunikationsbranche auswirkt. So

wurde in Deutschland im Januar 1998 mit dem neuen Telekommunikationsgesetz (TKG) der Telekommunikationsmarkt liberalisiert [Bun04, DH04], wodurch die bisherige Monopolstellung der Deutschen Telekom AG (DTAG) beendet wurde. Andere Staaten zeigen ähnliche Entwicklungen. Schwerpunkte im neuen TKG sind sowohl die Marktöffnung für Mitbewerber, Richtlinien zur Preispolitik als auch die Gewährleistungspflicht angebotener Dienste. Die Deregulierung und Privatisierung hatte bisher vor allem Auswirkungen auf die Mobilfunksparte und breitbandige Internetanbindungen. Insbesondere die Entgelte für die Bereitstellung von Online-Zugängen und Nutzung Online-Diensten unterliegen dabei einem starken Preisverfall.

Migration & Konvergenz Die Verbraucher greifen immer häufiger zu Telefon und Handy und surfen immer öfter im Internet. Netzbetreiber investieren zunehmend in den Ausbau neuer breitbandiger Netze, wodurch es in den kommenden Jahren zu einer nahezu kompletten Umstellung der Telekommunikation auf IP kommen wird.

Unter dem Begriff der *Migration* wird der Übergang der konventionellen Telekommunikationsdienste in das Internet verstanden. Am Ende der Migrationsprozesse steht letztendlich eine nahezu völlige Substitution durch digitale, internetbasierte Dienste. Damit in direktem Zusammenhang steht der Begriff der *Konvergenz*, welcher in der Telekommunikation die Vereinheitlichung verschiedener Einzelnetze und Dienste auf eine gemeinsame Basis bezeichnet. Es wird auch von Netz-Konvergenz oder Konsolidierung mit dem Ziel der Vereinfachung und Flexibilisierung der Infrastruktur gesprochen. Bisher wurden vier klassische Netzarten unterschieden: Sprachnetze, mobile Netze, Datennetze, und Rundfunknetze. Zudem basierten diese Einzelnetze regional auf verschiedenen technologischen Standards und regulatorischen Bestimmungen. Die Offenheit des Internets und die rasante technologische Entwicklung der letzten Jahre boten jedoch die geeignete Umgebung für die Vereinheitlichung auf globaler Ebene. Somit ist das an

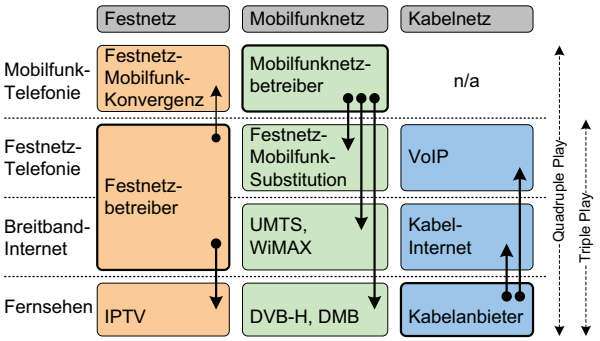


Abbildung 3.3.: Konvergenzrichtungen der Telekommunikationsmedien (nach [BI07])

sich „dumme Internet“ [RFC1958, SJ06] zusammen mit IP die gemeinsame zukünftige Basis, da es die Übertragung verschiedenster Inhalte ermöglicht. Die Migrationsprozesse finden dabei auf Ebene aller Telekommunikationsdienste statt, wobei vor allem Sprachdienste hervorstechen [Int07b]. Abbildung 3.3 zeigt die Migrationsrichtungen für Festnetz-, Mobilfunknetz- und Kabelnetzbetreiber. Die jeweiligen Kernmärkte sind nach wie vor dominant in ihren historisch bedingten Sparten vertreten, jedoch werden zunehmend „netzfremde“ Dienste innerhalb von Bündelpaketen angeboten.

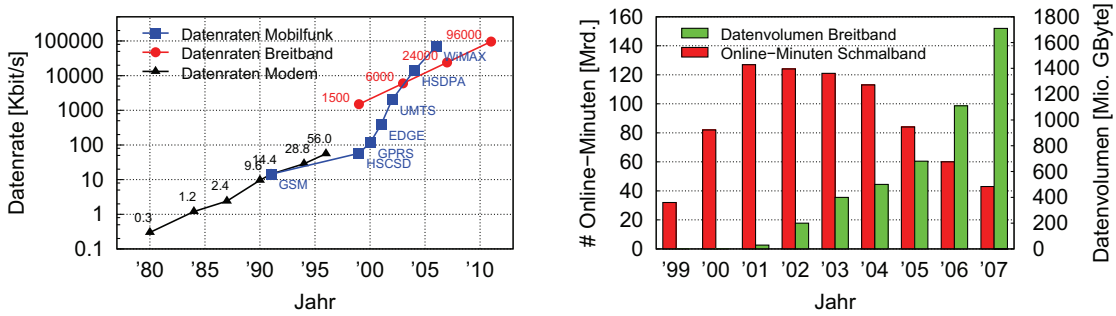
Steigende Nutzerzahlen & Kommerzialisierung Seit der Verfügbarkeit analoger Einwahlverbindungen ist die Zahl der Internetnutzer weltweit stetig gestiegen. Mit Beginn der 90-er Jahre bis heute hat sich die Internet-“Population“ auf mittlerweile circa 1,3 Milliarden Nutzer vergrößert. Vor allem in den letzten Jahren ist dies auf die Migrationsvorgänge und damit den Zuwachs an Online-Haushalten zurückzuführen. Die hohe Akzeptanz des Mediums Internet und die steigenden Nutzerzahlen sind die bis heute wichtigsten Einflussgrößen auf die Entwicklung des Internets [Odl03]. Die Adaptionskurven bezüglich neuer Technologien und Internetnutzung in [Van02, BM06c] belegen dies.

Steigende Nutzerzahlen bedeuten aus ökonomischer Sicht mehr Endverbraucher und damit mehr Umsatz und Einnahmen. Ein Vielzahl von Endkunden kann z. B. mit Werbung im Internet erreicht werden. Außerdem verursacht Online-Werbung deutlich weniger Kosten als Werbung mittels klassischen Rundfunks und kann in personalisierter Form angeboten sowie durch den vorhandenen Datenrückkanal interaktiv gestaltet werden. Ein aktuelles Beispiel ist der Marktkampf zwischen Google, Yahoo und Microsoft[2]. Diese voranschreitende Kommerzialisierung des Internets steht im Gegensatz zu dessen ursprünglicher Nutzung als reines Wissenschaftsnetzwerk. Heute existieren neben den typischen Sprach-, Video- und Informationsdiensten verschiedenste Märkte im Internet und dem World Wide Web (WWW), z. B. der Internethandel (Ebay, Amazon), der Konsum von Musik und Videos (iTunes), Online-Spiele, Social Networking Services (Facebook, MySpace) oder Online-Bezahlsysteme (PayPal).

3.2.2. Technologische Trends

Neue Technologien & Breitbandzuwachs Um anspruchsvolle Dienste in entsprechender Qualität bis zum Endkunden liefern zu können – so wird geschätzt – werden sich die Bandbreitenanforderungen in den nächsten Jahren mehr als verzehnfachen. Im Bereich der TZN stellt herkömmliches DSL in diesem Zusammenhang nur ungenügende Ressourcen bereit. Die in Deutschland vorherrschende kupferbasierte Netzinfrastruktur, insbesondere auf den ersten

[2] siehe dazu `http://www.heise.de/newsticker/meldung/102910`



(a) Entwicklung der Datenraten verschiedener Technologien

(b) Zunahme des Breitbandvolumens und Abnahme der Schmalbandverbindungsminuten in Deutschland

Abbildung 3.4.: Entwicklung der Datenraten und Technologien

Metern von/zum Kunden, ist steuerlich überwiegend abgeschrieben, wodurch jede Chance der Weiternutzung Gewinn bedeuten würde. Die Nutzung von ADSL bzw. VDSL ist an dieser Stelle nur ein Anfang [BMB01]. Aktuelle Arbeiten mit dem Ziel, Datenraten von 1–100 Gbit/s über konventionelle Kupferleitungen zu erreichen, können der vorhandenen Infrastruktur auf der sogenannten First Mile vom/zum Endkunden neuen Wert verleihen. Arbeiten in diesem Bereich sind z. B. Gigabit-DSL (GDSL) [CLM+04, CBP+06] oder CuPON [CJMG07]. Aus diesen Gründen ist die Revitalisierung der existierenden Netzwerkinfrastruktur nicht zuletzt aus Kostengründen ein Muss. Es wird sich diejenige Technologie behaupten, die den besten Kompromiss aus Kosten, Funktion und Flexibilität bietet, wobei vor allem im Bereich der TZN die jeweils vorherrschenden regionalen Gegebenheiten eine große Rolle spielen. Dies ist an den Unterschieden zwischen dem asiatischen Raum (xDSL/Glasfaser/Ethernet), Europa (xDSL) und dem nordamerikanischen Markt (Kabel/xDSL) [BM06c] ersichtlich. Gerade neue Technologien wie Fiber-to-the-x (FTTx) [Lin06], Gigabit Passive Optical Network (GPON) [SS05], oder Ethernet-in-the-first-Mile (EFM) [Bec05, IEEc] erlauben höchste Datenraten direkt zum/vom Endkunden.

Abbildung 3.4a zeigt die Entwicklung der Datenraten verschiedener Klassen von Zugangstechnologien zum Internet. In allen Bereichen ist ein deutlicher Anstieg zu erkennen. Mit der Verfügbarkeit von Glasfaser und zukünftigem GDSL und CuPON sind in den kommenden Jahren Datenraten bis über 1 Gbit/s zu erwarten. Gleichzeitig entwickeln sich auch die Funktechnologien weiter [VLLX02]. Mit UMTS, HDSPA oder WiMAX existieren drahtlose Kommunikationsstandards, die mobile Geräte mit weit über 1 Mbit/s an das Internet anbinden. In Abbildung 3.4b ist die Substitution analoger Einwahlmodems durch Breitbandverbindungen deutlich zu erkennen. Die Summe aller durch Schmalbandzugänge erzeugten Online-Minuten nimmt seit

2001 stetig ab, während das Breitbandverkehrsvolumen steigt. Dies ist auch an der Zunahme des durchschnittlichen Gesamtverkehrs am Deutschen Zentralknoten DE-CIX [DCX] ersichtlich und entspricht den Prognosen aus [CO01]. Dem Verbraucher wird zunehmend mehr Bandbreite für immer geringere Kosten zur Verfügung gestellt. Durch diese Bandbreitensättigung am Rand des Netzes sind hohe Datenraten kein Alleinstellungsmerkmal mehr aus Sicht der ISPs und Netzbetreiber. Viel mehr Bandbreite ändert nicht gleichermaßen viel am Qualitätsempfinden der Kunden. Deshalb müssen ISPs andere Mehrwerte (Mehrwertdienste, Value-Added Services) suchen, um die eigenen Angebote gegenüber denen der Mitbewerber zu differenzieren.

Neue Dienste & Anwendungen Die Konsolidierung der verschiedenen Netze bringt eine zunehmende Dienstevielfalt mit sich. Diese beschränkt sich nicht nur auf die klassischen Basisdienste, welche nun in digitaler Form per Bündelangebote im Internet verfügbar sind, z. B. Triple Play, Quadruple Play (siehe auch Abbildung 3.3) oder gar Quintuple Play [Alc05, KNP07]. Durch seine offenen Standards und Schnittstellen werden im Internet zunehmend neue Dienste, Medien und Nutzungsmöglichkeiten auf verschiedenen Ebenen geschaffen und vermarktet. Dazu zählen u. a. verschiedene Nutzungsformen von Peer-to-Peer (P2P), das sogenannte Web 2.0 bzw. Social Web, neue Medien und die Digitalisierung klassischer Druckmedien, Interaktivität und personalisierte Dienste. Weitere Nutzungsformen sind in [CGS05, DSL07] beschrieben. Die Kommunikation per E-Mail und die bloße Informationssuche im WWW sind dabei nur noch zwei untergeordnete Aspekte.

Tabelle 3.1.: Ungefährer Bandbreitenbedarf einiger moderner videobasierter Dienste

Dienst	Bandbreitenbedarf	Qualität
Standard Definition TV (SDTV)	3–6 Mbit/s	einstellbar
High Definition TV (HDTV)	ca. 10 Mbit/s	einstellbar
Picture-in-Picture, Channel Preview	bis 6 Mbit/s	einstellbar
Pay-per-View, Personal Video Recorder	3–6 Mbit/s	priorisiert
Interaktives TV	bis 3 Mbit/s im Upload	Best-Effort

Am Beispiel von videobasierten Diensten auf Basis von IPTV soll die Verbreiterung der Palette von Mehrwertdiensten verdeutlicht werden, da IPTV als einer der wichtigsten Treiber für das Wachstum der verfügbaren Bandbreite und Anzahl der Breitbandanschlüsse angesehen wird [GSS07]. Einige Vorteile von IPTV gegenüber klassischem Fernsehen sind die digitale Bildqualität, weltweite Verfügbarkeit der persönlichen Lieblingssendung, Interaktivität durch Nutzung des Rückkanals, sowie Zeitunabhängigkeit. Tabelle 3.1 zeigt in diesem Zusammenhang verschiedene videobasierte Dienste und ihren ungefähren Bandbreitenbedarf (u. a. nach [Sie06]).

Unter der Annahme, dass in Zukunft in einem durchschnittlichen Haushalt mehrere dieser Dienste gleichzeitig und nebenher weitere o. g. Diensten genutzt werden, stoßen die heute verbreitet zum Einsatz kommenden DSL-Derivate bereits an ihre Grenzen (vgl. Abbildung 3.4a).

3.2.3. Soziale und gesellschaftliche Trends

Flatrates – Always-on – 24 h×7 d Aus der enormen Adaption moderner Dienste und Nutzungsmöglichkeiten des Internets resultiert eine immer umfassendere Integration des Internets in das alltägliche Leben. Es wird von der Digitalisierung des Alltags gesprochen. Der Einkaufsbummel wird weitestgehend in Online-Shops getätigt, das klassische „Mensch ärgere dich nicht!“ wird durch Online-Gaming ersetzt und der komplette Freundeskreis ist in Buddy-Listen von Chat-Clients und sozialen Plattformen enthalten. Mit dem Erscheinen von Flatrates, einer flacheren Preispolitik und Kombipaketen hoben sich auch die zeitlichen Beschränkungen auf, die die bisherige Nutzung des Internets verglichen mit heute aus finanzieller Sicht eher behinderten [Odl01b, LO08]. Ein Zitat aus [MB00] lautet: „Internet users who pay a fixed fee have no incentive to limit their use of the network.“ Aus diesen Gründen hat sich auf der einen Seite eine sogenannte „Always-on“-Mentalität in der Gesellschaft ausgebreitet und manifestiert, welche z. B. in [CTD08] unter dem Namen *Participation Culture* dokumentiert ist. Auf der anderen Seite hat dies aber auch für ISPs und Content-Provider Auswirkungen. Das Internet ist eine globale Instanz wachsender Beliebtheit. Irgendwo auf der Welt ist immer jemand online. Es gibt keine Schließzeiten im Internet, sondern ausschließlich 24 Stunden Verfügbarkeit an 7 Tagen der Woche. D. h., Erreichbarkeit und Verfügbarkeit müssen insbesondere dann garantiert werden, wenn es sich um abonnierte und gebührenpflichtige Dienste handelt. Ein Großteil der mobilen Nutzer hält z. B. das Handy permanent in direkter Nähe bereit. Deshalb steigen sowohl die Nutzungsdauer des Internets pro Kopf und Tag als auch das Gesamtverkehrsaufkommen.

Generierung eigener Inhalte Mit dem Boom der sozialen Plattformen zeigte sich ein neuer Trend im Internet, der vom bisherigen Modell der alleinigen Bereitstellung der Inhalte durch ISPs und Content Provider abweicht. In [KNP07] wird dafür als Kontrast zum Konsumenten der Begriff des *Prosumenten* gebraucht. Dienste und Informationen werden nicht mehr nur konsumiert, sondern auch in großem Maße von den Endkunden selbst generiert, bereitgestellt und verbreitet. Dies ist allgemein unter den Begriffen User generated Content bzw. Do-it-Yourself Content zu verstehen. Der Slogan von YouTube lautet z. B.: *„Broadcast Yourself“*. Der neueste Trend stellt sich durch die Produktion eigener, qualitativ hochwertiger TV-Sendungen dar, die von anderen Nutzern online konsumiert werden können. Beispiele dafür sind das Projekt Miro TV [MIRO] sowie das Mogulus Projekt [Mog]. Diese Möglichkeiten gehen damit weit darüber

hinaus, was interaktives Fernsehen bieten kann. Die Generierung eigener Inhalte und deren Bereitstellung geht dabei einher mit dem Wachstum von Rechnerleistung und Speicherkapazitäten, der Marktdurchdringung von Breitbandtechnologien und dem Reedschen Gesetz bezogen auf die Ausbildung sozialer Netze und Gemeinschaften.

Mobilität Die Anzahl an frei zugänglichen Internet-Hotspots steigt weltweit an, ebenso die Zahl der verkauften internetfähigen mobilen Geräte (für genaue Zahlen sei z. B. auf die Referenzen unter Abschnitt 3.2 verwiesen). Mobiler Zugriff auf das Internet wird durch zwei Dinge begünstigt. Einerseits durch die globale Ausdehnung des Internets und dessen einheitliche Schnittstellen und Protokolle. Andererseits durch die Verfügbarkeit entsprechend handhabbarer und tragbarer Geräte sowie drahtloser Übertragungstechniken [Kle08].

Von unterwegs und nahezu überall Zugriff auf Informationen zu haben ist bequem und spart Zeit. Zudem sinken die Kosten für mobile Telekommunikationsdienste, wie Abschnitt 3.2.1 bereits erläuterte. Aktuelle drahtlose Übertragungstechnologien weisen mittlerweile akzeptable Datenraten und Bandbreiten auf, um anspruchsvolle multimediale Inhalte konsumieren zu können (siehe Abbildung 3.4a). *Wireless* bedeutet schon längst nicht mehr automatisch, keine gute Verbindung im Sinne von *schnell* zu haben.

Internetkriminalität Während die bisher genannten Trends positive Entwicklungen darstellten, ist der Anstieg der Internetkriminalität das meistdiskutierte Negativbeispiel. Es ist noch nicht gelungen, das Internet genügend sicher zu gestalten. Die Anzahl der Bedrohungen, ihre Häufigkeit und Effektivität nehmen sogar zu, was letztendlich das Vertrauen der Nutzer in das Medium Internet senkt [Fal03, Iro08]. Die Entstehung dieser „Dark Side“ [LCC+97, Kle04] liegt zum einen an der offenen Natur des Internets und den teilweise veralteten Protokollen, da zu deren Entstehungszeitpunkt das Heranwachsen des Internets zu einem Massenmedium nicht absehbar war. Zum anderen befinden wir uns momentan in einer Phase der Veränderungen, in der konventionelle Dienste aus unabhängigen Einzelnetzen in ein gemeinsames, paketvermitteltes Netzwerk migrieren, welches auf einem gänzlich anderen Konzept aufbaut. Dabei kann es sowohl zu rechtlichen Schlupflöchern als auch zum Auftreten technischer Fehler und Hintertüren kommen, welche unbehelligt ausgenutzt werden [Int07a]. Dies geht zumeist weit über den simplen Virusbefall eines durchschnittlichen Heim-PCs hinaus. Die Palette an Bedrohungen reicht von lästigen Spam-E-Mails über Identitätsklau und Spionage bis zu schädigenden Viren, Würmern und Trojanern [Erb05]. Sogenannte Distributed Denial-of-Service Angriffe (DDoS) kompromittieren zudem die Verfügbarkeit von Diensten, Internetseiten und Geräten. Dabei können alle Teilnehmer und Nutzer, Dienste und auch das Netzwerk selbst zum Ziel derartiger Attacken werden. Aktuelle Bedrohungen sind z. B. in [Sym07b, Arb07, Iro08] dokumentiert.

3.2.4. Zwischenfazit – Chancen und Ziele

Das Internet hat sich seit Beginn der 90-er Jahre zu einem Massenmedium entwickelt. Die Zielgruppe hat sich drastisch verändert und diversifiziert. Die o. g. genannten Trends, ausgenommen die steigende Internetkriminalität, verfolgen dabei gemeinsame „größere“ Ziele und sind als Chancen zur Erreichung dieser Ziele zu verstehen:

- Übergang zu einem global vereinheitlichten Kommunikationsmedium und Abschaffung regional unterschiedlicher und inkompatibler Netze und Standards
- Schaffung einer ubiquitären Informationsquelle sowie Ermöglichung kostengünstiger Zugänge zu Informationen für Alle (inkl. Entwicklungsländer)
- Umfassender Schutz vor Sicherheitsrisiken und Internetkriminalität
- Transparente Integration in den Alltag und Schutz der Privatsphäre

Ob diese Ziele erreicht werden, hängt davon ab, wie gut die Probleme und Herausforderungen bewältigt werden, die sich aus den gegenwärtigen Veränderungen ergeben.

3.3. Problemstellungen und abgeleitete Anforderungen

Wie Abschnitt 3.2 zeigte, sind im Telekommunikationsmarkt auf allen Ebenen steigende Wachstumstrends zu erkennen: zunehmende Nutzerzahlen und Heterogenität, steigende Verkehrsvolumina, die Zunahme der Nutzungsdauer, neue Anwendungen und Dienste mit individuellen Güteanforderungen sowie breitbandigere Übertragungstechniken. Hinzu kommt eine verwirrende Preispolitik. Die größte Problematik, die es in diesem Zusammenhang zu bewältigen gilt, ist eben diese zunehmende *Gesamtkomplexität*, welche einen vielschichtigen Entscheidungsraum aufspannt. Um diese Komplexität bewältigen zu können, sind auf allen Ebenen neue *skalierbare und flexible* Konzepte erforderlich. Nachfolgend sind zentrale Problematiken und daraus abgeleitete Anforderungen an den zukünftigen Telekommunikationssektor erläutert.

Sicherheit und Vertrauenswürdigkeit Weitestgehend bedingt durch die zunehmende Internetkriminalität (siehe Abschnitt 3.2.3) ist die Sicherheit in aktuellen und zukünftigen Kommunikationsnetzen ein zentrales Problem. Durch die Kommerzialisierung wird das Internet mehr und mehr zum „Bezahlmedium“ für den Konsumenten. Aus Sicht bösartiger Nutzer wird das Internet jedoch zu einer „Geldmaschine“ mit Selbstbedienung, die den unwissenden Normalverbraucher ausnutzt. Sicherheit bezieht sich deshalb auf verschiedene Ebenen und dient dem Schutz der

Nutzer vor anderen Nutzern und vor sich selbst und dem Schutz der Netzwerkinfrastruktur vor böswilligen oder im besten Fall unwissenden Nutzern. Vertrauenswürdigkeit steht dabei in direktem Zusammenhang mit Netzwerksicherheit. Zum einen betrifft dies das Vertrauen der Nutzer in das Medium Internet, welches durch die genannten Bedrohungen stark beeinträchtigt wird. Zum anderen bezieht sich dies vor allem auf die im Internet herrschende Anonymität. Im Gegensatz zu leitungsvermittelten Netzen bietet das paketvermittelte Internet keine eindeutige Möglichkeit der Rückverfolgung von Datenverkehr und damit der glaubhaften Identifizierung des Senders. Wer im Internet unerkannt bleiben will, kann dies heutzutage prinzipiell durch geeignete Maßnahmen erreichen (einschlägiges Fachwissen vorausgesetzt). Diese Vertrauensbasis mag im Web 1.0 funktioniert haben, ist aber schon für das heutige Web 2.0 ungeeignet und wird deshalb auch für das zukünftige Web 3.0[3] nicht ausreichen.

Neue Sicherheitskonzepte sind deshalb nötig, die an moderne Telekommunikationsdienste und Anwendungen angepasst und kompatibel zu den bisherigen angewandten Mechanismen sind. Die Ziele sind dabei der Schutz privater und sensibler Informationen, die Absicherung kommerzieller Anwendungen, die Garantie der Verfügbarkeit der Netzwerkkomponenten und -dienste sowie die Gewährleistung abonnierter Dienstgüten. Einerseits sind Sicherheitskonzepte somit auf Protokoll- und Anwendungsebene notwendig, vor allem zur Authentifizierung von Teilnehmern und Sicherung der Vertrauenswürdigkeit. Andererseits werden Sicherheitsmechanismen auch innerhalb der Netzwerkinfrastruktur und in Netzwerkgeräten wie Routern benötigt, um z. B. Bedrohungen in sowohl ein- als auch ausgehendem Datenverkehr frühzeitig erkennen zu können. Die Diplomarbeit von Danielis [Dan06, DKT07] zeigt beispielsweise eine derartige Lösung in Form eines Intrusion Detection Systems (IDS).

Veraltete Protokolle Nicht mehr zeitgemäße Protokolle können zu Sicherheitsrisiken und technischen Unzulänglichkeiten führen. Das Protokoll für den Versand von E-Mails, das Simple Mail Transfer Protocol (SMTP) [RFC2821], wurde z. B. ohne jegliche Authentifizierungsmechanismen implementiert, was heutzutage mit einer der Hauptgründe für Spam ist. Darüber hinaus behindern in die Jahre gekommene Protokolle das weitere Fortschreiten der Migration und Konsolidierung, wie das Beispiel der Ausschöpfung des Adressraumes von IPv4 zeigt. Beide Beispiele – IP und SMTP – sind Kernprotokolle des Internets!

Die *Integration neuer Protokolle bzw. Protokollversionen* ist daher notwendig, um auf die veränderten Bedingungen in Internet und TZN zu reagieren. Ein Beispiel ist die schrittweise Umstellung von IPv4 auf IPv6 mit dem Hintergrund effizienterer Sicherheitsfeatures und eines deutlich größeren Adressbereichs. Um ein Höchstmaß an Sicherheit zu gewährleisten, ist *strikte Standard-*

[3] Die Begriffe Web 1.0…3.0 sind hier symbolisch als Evolutionsstufen des gesamten Internets zu verstehen.

konformität Voraussetzung, denn jedwede Aufweichung eines Standards führt zu Schlupflöchern.

Verfügbarkeit Verfügbarkeit ist in der Telekommunikation aus verschiedenen Blickwinkeln zu sehen. Zum einen betrifft Verfügbarkeit die technische Umsetzbarkeit eines Dienstes. Dabei erwarten die Konsumenten, dass die neuen Dienste mindestens die gleichen Qualitäten und Funktionen realisieren wie ihre klassischen Pendants. Gegenbeispiele sind IPTV, welches zum Teil noch unerwünscht lange Umschaltzeiten beim Kanalwechsel aufweist, oder die mobile VoIP-Telefonie, für welche bezüglich der Handhabung von Notrufen noch kein einheitlicher Standard existiert. Verfügbarkeit bezieht sich aber auch auf die Nutzbarkeit von Diensten zu jedem Zeitpunkt und überall. *Zeitliche und räumliche Ubiquität* müssen für den Bedarfsfall gewährleistet sein. Drittens bedeutet Verfügbarkeit Ausfallsicherheit. Zentrale, kritische Netzwerkkomponenten müssen besonders gegen Ausfälle abgesichert sein. Dies betrifft insbesondere den Schutz vor koordinierten Angriffen durch zusätzliche Sicherheitsmaßnahmen.

Wartung und Management Steigende Nutzerzahlen, der Übergang zu neuen Technologien und die Vielfalt gebührenpflichtiger Dienste und abgestufter Preiskategorien verkomplizieren die Verwaltung, Wartung und Vergebührung im Internet. Angefangen von der Zuweisung einer IP-Adresse und Konfiguration der Einwahldaten müssen Endgeräte der Nutzer (Customer Premises Equipment (CPE)) und TALs konfiguriert werden. Aber gerade nicht versierte Normalverbraucher scheuen die technischen Seiten, die mit der Internetnutzung einhergehen. Abonnements müssen zudem verwaltet und in Anspruch genommene Dienstleistungen abgerechnet werden. Hinzu kommt die Wartung der Netzwerkinfrastruktur durch die Netzbetreiber. Diese komplexen Aufgaben fallen in die Bereiche AAA (Authentication, Authorization & Accounting) [NN05] und OAM (Operation, Administration & Management).

Einfachheit und *Transparenz* sind deshalb Anforderungen an zukünftige Lösungen und Mechanismen in der Telekommunikation. ISPs sind an vereinfachtem Netzwerkmanagement und der Flexibilität, schnell auf Kundenwünsche eingehen zu können, interessiert. Das DSL Forum stellt z. B. mit [DSL04b] ein Framework vor, welches sowohl den technisch unbedarften Normalverbraucher als auch den ISP bei der automatischen Konfiguration des CPE unterstützt. Mit ACIP (Access Control and Information Protocol) präsentieren Duchow et al. einen flexiblen und universellen Ansatz für AAA [DBT+06a, DBT+06b]. D. h., die zunehmende Komplexität muss durch Abstraktion, Transparenz und Simplizität verborgen werden.

(A)symmetrische Internetanbindungen Der Wandel der Endkunden von reinen Konsumenten zu Prosumenten (siehe Abschnitt 3.2.3) erfordert auch einen Wandel in der technischen Anbindung an das Internet. Bisher lag das Hauptaugenmerk auf einem möglichst breitbandigen

Kanal zum Konsumenten. Der Rückkanal ist hingegen meist mit nur einer geringen Bandbreite ausgelegt (natives ADSL, VDSL), was in diesem Zusammenhang unter Asymmetrie zu verstehen ist. Jedoch reicht dies heute und in Zukunft nicht mehr aus, um eigene Inhalte vergleichbarer Qualität anderen Nutzern zugänglich machen zu können. Obwohl symmetrische Anschlusstechniken durchaus existieren (SDSL, VDSL-Varianten), sind sie jedoch gerade in Deutschland noch nicht weit verbreitet. Asymmetrische Verbindungen dominieren noch den Markt [Agi06]. Ein Ziel muss deshalb die *Verbreitung breitbandiger, symmetrischer Anschlusstechniken* sein, insbesondere im Bereich der privaten Nutzung.

Best-Effort oder garantierte Dienstgüten? Die Einhaltung von Parametern wie verfügbare Bandbreite oder maximal zulässige Verzögerungszeiten beruht auf einer Kompromissfrage, für die es prinzipiell zwei Ansätze gibt. Einerseits kann der Verletzung von Dienstgütekriterien durch ein Überangebot an Bandbreite entgegengewirkt werden. Dieses Vorgehen basiert letztendlich auf dem Best-Effort-Prinzip und kann deshalb die Einhaltung dieser Randbedingungen ggf. nicht immer garantieren. Es ist jedoch einfach realisierbar. Andererseits müssen sich ISPs infolge der Kommerzialisierung und Liberalisierung differenzieren können und benötigen Alleinstellungsmerkmale für ihre Produkte. Da niemand für Best-Effort bezahlt, ist die Garantie sogenannter Dienstgütevereinbarungen bzw. SLAs vor allem durch ökonomische Aspekte motiviert. Die Zusicherung und Realisierung von QoS hat allerdings negativen Einfluss auf Kosten und Komplexität der Paketverarbeitung. Welcher Grad von QoS sinnvoll ist und wie die technische Umsetzung aussieht, wird vor allem durch die Erlebnisgüte bzw. Quality-of-Experience (QoE) der Konsumenten bestimmt. Zudem erfordern kritische Dienste, z. B. Überwachungs- und Sicherheitsfunktionen oder das Tätigen von Notrufen, in jedem Fall die Einhaltung bestimmter Parameter. Zur Problemminderung sind ausreichend Bandbreite sowie Leistungsreserven bzgl. der Paketverarbeitung erforderlich, was durch sowohl *leistungsfähige Systemarchitekturen* als auch *flexible Netzwerktopologien und Transportmechanismen* erreicht werden kann, welche die vorhandenen Ressourcen effizient ausnutzen.

Leistungsfähigkeit Die Leistungsfähigkeit einzelner paketverarbeitender Geräte wie Switche oder Router ist ein Flaschenhals in der Infrastruktur des Internets [Var05]. Zum einen wachsen Bandbreite und zu bewältigende Datenraten an. Zudem wird Bandbreite für gewöhnlich überbucht (Overprovisioning), da ISPs davon ausgehen, dass nie alle Nutzer gleichzeitig online sind. Während der Hauptbelastungszeiten kommt es dann zu Engpässen. Zum anderen steigt die Komplexität und sinkt die Regularität in der Paketverarbeitung (siehe dazu Abschnitt 2.3). Datenübertragung auf Basis von Best-Effort ist nur noch für wenige Dienste im Internet adäquat. Gerade Value-Added Services und multimediale Dienste fordern die Einhaltung strikter

QoS-Kriterien, welche wiederum eine tiefgründigere Paketklassifizierung sowie intelligentere Filterung nach sich ziehen, die bis in die komplexen Anwendungsschichten hineinreicht. Weiterhin resultiert die steigende Anzahl an Nutzern und Endgeräten in einer gleichsam steigenden Menge von Adressen und unabhängigen Einzelverbindungen, die es zu verwalten gilt. Die Administration dementsprechend tiefer Tabellen und Datenbanken und die Suche nach Informationen benötigen zusätzlich Aufwand und Zeit. Dies bedeutet, dass pro Zeiteinheit mehr Informationen mit komplexeren Routinen verarbeitet werden müssen. Netzwerkgeräte müssen für eine blockierungsfreie Abarbeitung der Daten (Wire-speed Packet Processing) intern ein Vielfaches der eigentlichen Datenrate des Übertragungskanals bewältigen können sowie ausreichend Flexibilität und Raum für zukünftige Erweiterungen bieten. Konventionelle Architekturen und reine Softwarelösungen können dies nicht. Deshalb sind *leistungsstarke* bzw. *flexible Systemarchitekturen* notwendig[4]. Widiger et al. diskutieren aus diesem Grund z. B. die Verwendung eines adaptiven Paketklassifizierers auf Basis einer evolvierbaren Hash-Funktion, welche die durchschnittliche Suchkomplexität stark reduziert [WST06, SWT06, WTST07].

Neben der Performanz paketverarbeitender Geräte spielt auch die verfügbare Bandbreite und Flexibilität der Netzwerkinfrastruktur eine Rolle. Beim Endkunden am Rand des Netzes – sowohl private Haushalte als auch kommerzielle Firmen-Netze – ist genügend Bandbreite vorhanden sein. Im Kernnetzbereich bieten optische Medien ebenfalls ausreichend Bandbreite, da in diesem zentralen Segment der Kostenfaktor weniger kritisch ist und anfallende Kosten auf sehr viele Nutzer verteilt werden können. Durch die Bündelung mehrerer tausend Teilnehmeranschlussleitungen bilden die TZN jedoch den entscheidenden Engpass in Bezug auf die Bandbreite, auch wenn absolut mehr Bandbreite zur Verfügung steht als ein einzelner Kunde benötigt. D. h., die *Stärkung der Aggregationspunkte* innerhalb der TZN ist von großer Bedeutung für die weitere Entwicklung in der Telekommunikation. Dabei wirken sich Overprovisioning, QoS-Forderungen und ein enormer Kostendruck erschwerend auf dieses Netzwerksegment aus. Einerseits sind darum breitbandige und gleichzeitig kostengünstige Übertragungstechnologien für TZN unerlässlich. Gerade die neuen Ethernet-Generationen bieten hier aufgrund ihrer Eigenschaften (siehe Abschnitt 2.4) einfache, günstige Alternativen verglichen mit z. B. ATM oder SONET [Beu07]. Andererseits sind Mechanismen zur effizienten Ressourcenauslastung nötig, z. B. zur dynamischen Bandbreitenanpassung, um vor allem während Hauptverkehrszeiten die Verteilung begrenzter Netzwerkressourcen zu koordinieren. Die Masterarbeit von Strzeletz [Str08] bietet in diesem Zusammenhang Informationen zu derartigen Mechanismen.

[4] Hier sei u. a. auf den zweiten Teil dieser Arbeit verwiesen.

3.4. Zusammenfassung des Kapitels

Kapitel 3 gab einen Einblick in die aktuellen Geschehnisse und Entwicklungen im Telekommunikationsmarkt, welcher momentan durch starke Globalisierungs- und Konsolidierungsprozesse gekennzeichnet ist. Aus den daraus entstehenden Problemstellungen wurden verschiedene Anforderungen abgeleitet, die für die weitere Entwicklung des Internets und des Telekommunikationsmarktes bedeutsam sind. Auf verschiedenen Ebenen spielen zukünftig insbesondere Skalierbarkeit und Flexibilität eine wichtige Rolle:

- Neue Sicherheitskonzepte
- Leistungsstarke Systemarchitekturen für paketverarbeitende Geräte und effiziente Ressourcenausnutzung
- Neue Protokolle unter Einhaltung strikter Standardkonformität
- Flexible physikalische und logische Netzwerkinfrastrukturen und Transportmechanismen
- Einfachheit, Transparenz und Ubiquität für Endnutzer und ISPs
- Verbreitung symmetrischer Breitbandtechnologien zur Anbindung ans Internet

Abbildung 3.5 gibt einen komprimierten Überblick über die Zusammenhänge zwischen Ursachen, Trends, aktuellen Problematiken sowie den abgeleiteten Anforderungen[5], welche in diesem Kapitel diskutiert wurden. Als Beispiel sind nur diejenigen Beziehungen durch Verbinder markiert, die letztlich zur Notwendigkeit neuer Sicherheitskonzepte führen (alle Beziehungen darzustellen wäre zu komplex). Diese Darstellung ist gleichzeitig als ein sich wiederholender Entwicklungszyklus zu verstehen, dessen Ist-Zustand festgehalten wurde. In ihrem weiteren Verlauf greift die vorliegende Arbeit mit den vorgestellten Lösungsansätzen einige der genannten Probleme und Anforderungen direkt bzw. indirekt auf. In Abbildung 3.5 sind diese dick umrandet hervorgehoben.

[5] Abbildung 3.5 ist allenfalls nur ein Ausschnitt der Gesamtsituation und legt daher keinen Wert auf Vollständigkeit.

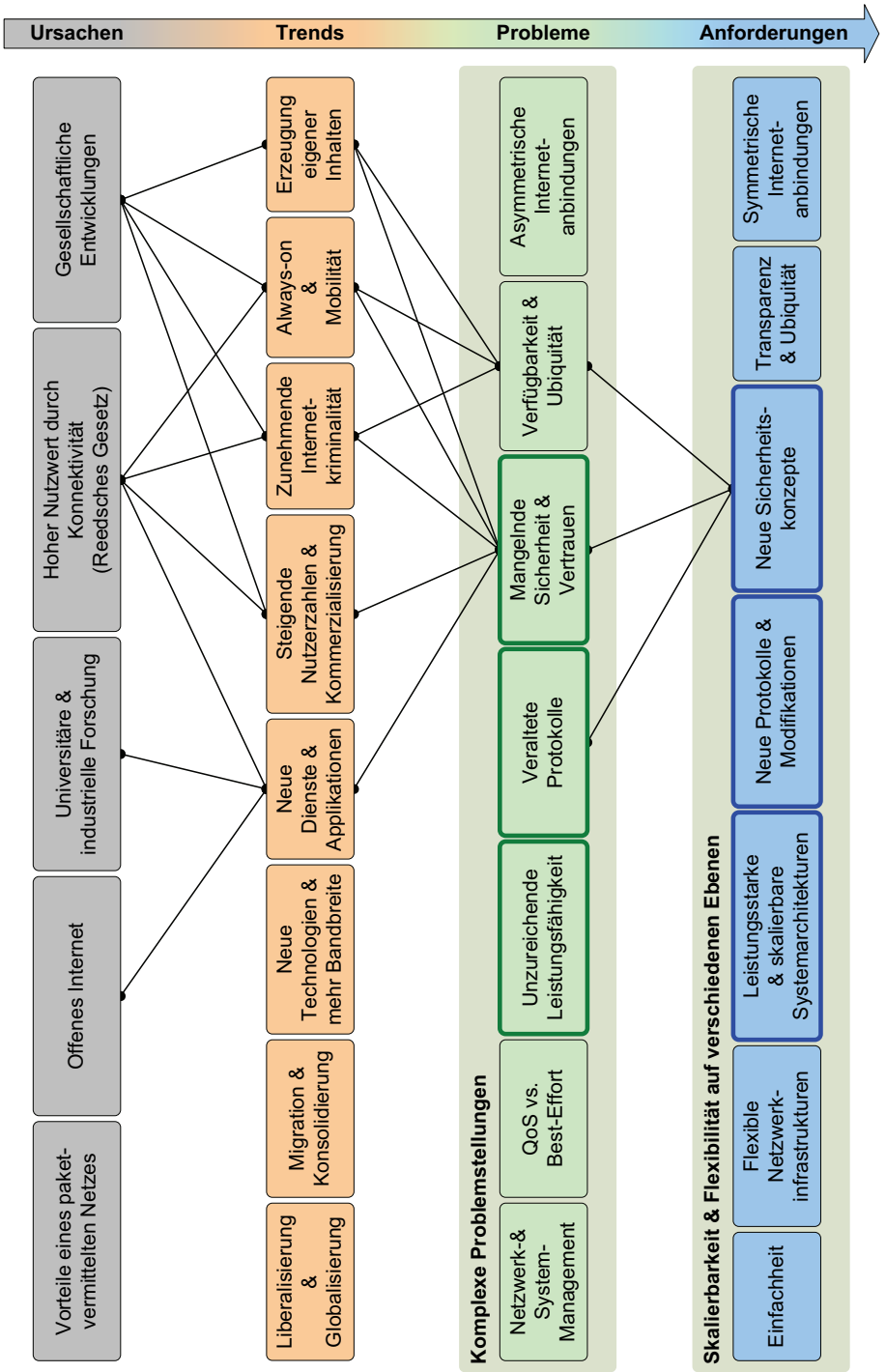


Abbildung 3.5.: Ist-Zustand der aktuellen Entwicklungen in der Telekommunikation und im Internet: Ursachen → Trends → Probleme → abgeleitete Anforderungen

Teil II.

Mechanismen und Architekturen für Teilnehmerzugangsnetzwerke

I don't have any solution,
but I certainly admire the problem.
(Ashleigh Brilliant, engl. Author & Karikaturist)

Kapitel 4.

Schicht-2-Adressumsetzung in Ethernet-basierten Teilnehmerzugangsnetzen

Kapitelstruktur

Kapitel 4 stellt Mechanismen für den Einsatz in Ethernet-basierten TZN vor. Der Schwerpunkt liegt auf den Verfahren MAT & sMAT in Abschnitt 4.2, welche vor allem die Aspekte der Skalierbarkeit und Sicherheit in Ethernet-basierten TZN adressieren. Weiterhin wird in Abschnitt 4.3 das MATMUNI-System vorgestellt. Es schließt u. a. eine direkte funktionale Umsetzung des MAT-Mechanismus ein und ist zudem Ausgangsbasis für weitere Untersuchungen in dieser Arbeit. Zuvor führt Abschnitt 4.1 in knapper Form in das Gebiet der TZN ein.

4.1. Teilnehmerzugangsnetze für das Internet

> To address the changing broadband marketplace, service providers must deploy value-added [...] technologies in their networks. One way to provide this flexibility is to place intelligence at the edge of the network so the network adapts to the nomadic users appearing at that edge, instead of asking the users to adapt to the network. The edge may be defined in a number of ways, but perhaps the most effective is to recognize that the edge is that place in the network where the unmanaged collection of end user devices [...] first meets the managed infrastructure of the Internet.
>
> *(Leonard Kleinrock, „Vater des Internets“, [Kle03])*

Dieses Zitat von Kleinrock beschreibt auf prägnante Weise das Wesen der TZN. Betrachtet man das globale Internet mit all seinen Facetten (siehe Abbildung 3.2), so sind die TZN dessen sogenannte Kanten (engl.: edges). Innerhalb der Hierarchie stellen die TZN – auch als Local Loop, Last Mile oder einfach nur als Access bezeichnet – das notwendige Brückenglied zwischen der Heterogenität der Teilnehmer und der Uniformität des Kernbereiches dar. TZN sind somit die intelligente Schnittstelle der Nutzer zum vergleichsweise einfachen Internet. Sie ordnen sich dabei in die in Abschnitt 2.2 eingeführte Klassifikation im Bereich der MANs ein, wie Abbildung 4.1 zeigt.

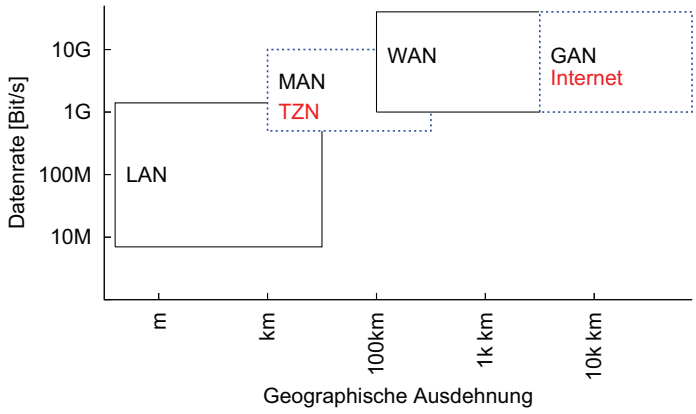


Abbildung 4.1.: Klassifikation von Teilnehmerzugangsnetzen

Ethernet-basierte TZN Eine Vielzahl verschiedener Typen von TZN existiert, z. B. für klassische Rundfunknetze oder das leitungsvermittelte PSTN. Diese Arbeit fokussiert sich jedoch auf breitbandige *Ethernet-basierte TZN* für das Internet. Für zusätzliche Informationen zu anderen Ausprägungen sei auf weiterführende Literatur verwiesen [BMS04, JW02, Hal05, Nok07].

Ethernet-basierte TZN sind die vorherrschende Form von TZN für das Internet [Hea05, TW07, Beu07]. Noch vor wenigen Jahren dominierte ATM auf Basis breitbandiger, optischer Medien. ATM wurde jedoch durch die Einfachheit und geringen Kosten von Ethernet immer mehr verdrängt. Ethernet bietet einen direkten Anschluss an die zumeist Ethernet-basierten LANs der Kunden, wodurch aufwendige Protokollumsetzungen entfallen. Zudem bietet Ethernet mittlerweile gleichwertige Bandbreiten und Mechanismen zur Sicherung von QoS (siehe Abschnitt 2.4). Rahmenbedingungen für die Migration zu Ethernet-basierten TZN sind in [DSL06] und [Met07] definiert. Das Metro Ethernet Forum (MEF) bezeichnet Ethernet im MAN/WAN mit dem Namen *Carrier Grade Ethernet*.

Je nach Umgebung und Übertragungsmedium weisen Ethernet-basierte TZN Unterschiede in Ausdehnung und Datenraten auf. Zu den drahtgebundenen Medien werden die gerade in Deutschland verbreitete Kupferdoppelader (Twisted Pair), das koaxiale TV-Kabel und auch das Stromversorgungsnetz (Powerline) gezählt. Zu den optischen Medien gehören die z. B. FTTx-Varianten auf Basis von (G)PON [SS05, Lin06, BM05], wobei sich 'x' auf den Glasfaseranteil der Anschlussleitung bezieht[1]. Gerade die Bereitstellung von breitbandigem VDSL erfordert einen hohen Glasfaseranteil, da VDSL per Kupferdoppelader – dem weitaus kostengünstigerem Medium – nur über kurze Distanzen möglich ist. Auch wenn die kupferbasierte Infrastruktur durch technologische Weiterentwicklungen revitalisiert werden kann (siehe Abschnitt 3.2.2), so schreitet die Umstellung des Netzes zu Glasfaser ausgehend vom Kernnetz in Richtung Teilnehmer sukzessiv voran. Zugangspunkte auf Basis von WLAN, WiMAX (WirelessMAN Air Interface) [EMSW02], UMTS und Satellitenverbindungen sind Beispiele für drahtlose Übertragungstechnologien, die gerade in unzugänglichen Gebieten oder Gegenden mit unzureichendem Ausbau des Festnetzes an Ethernet-basierte TZN angeschlossen werden, um sowohl der steigende Nachfrage nach Breitbandanschlüssen zu genügen als auch deren Bereitstellung ermöglichen zu können.

Struktur von TZN und Begriffsdefinitionen TZN weisen für gewöhnlich eine Baumstruktur mit verschiedenen Hierarchieebenen bzw. Aggregationsstufen auf. Dabei bleibt die grobe Segmentierung in hochgradig leistungsfähige, schnelle Kernnetze und aufwendige, intelligente TZN in Teilnehmernähe erhalten. Die Konsolidierung in der Telekommunikation, die Migration von Diensten und der Einsatz neuer Technologien resultieren jedoch in verschiedenen, an die jeweilige Technik angepassten TZN. Abbildung 4.2 zeigt die allgemeine Struktur von TZN am Beispiel eines typischen ADSL-Anschlusses der DTAG. Die Teilnehmergeräte sind per Splitter, ADSL-Modem und Telekommunikationsanschlusseinheit (TAE) an den sogenannten Hauptverteiler (HVt), einem Kabelverzweiger, angeschlossen. Der HVt terminiert die sogenannte

[1] D. h., wie weit die Glasfaser bis zum Kunden heranreicht (H = Home/Wohnung , B = Buildung/Gebäude...)

Primärverkabelung im Zugangsnetz. Der Abschnitt der TAL vom HVt zur TAE wird i. Allg. als First Mile (Upstream) oder Last Mile (Downstream) bezeichnet. Der DSLAM (DSL Access Multiplexer) bildet die erste Hauptaggregationsstufe und bündelt eine Vielzahl von Teilnehmern in einen hochvolumigen Datenstrom. Ein optionales Konzentratornetz (Metro Aggregation) enthält weitere kaskadierende DSLAMs bzw. Metro-Switches, die letztendlich an einem BRAS (Broadband Remote Access Server) terminieren. Der BRAS ist der erste sogenannte Point-of-Presence (PoP) bzw. Router des jeweiligen Kernnetzbetreibers im Kernbereich. Vom BRAS ausgehend kann zu den verschiedenen Netzen der ISPs – die letztlich das globale Internet bilden – geroutet werden. Im TZN wird Datenverkehr auf Ebene der Sicherungsschicht weitergeleitet, während im Kernnetz weitestgehend auf Basis von IP (Vermittlungsschicht) geroutet wird.

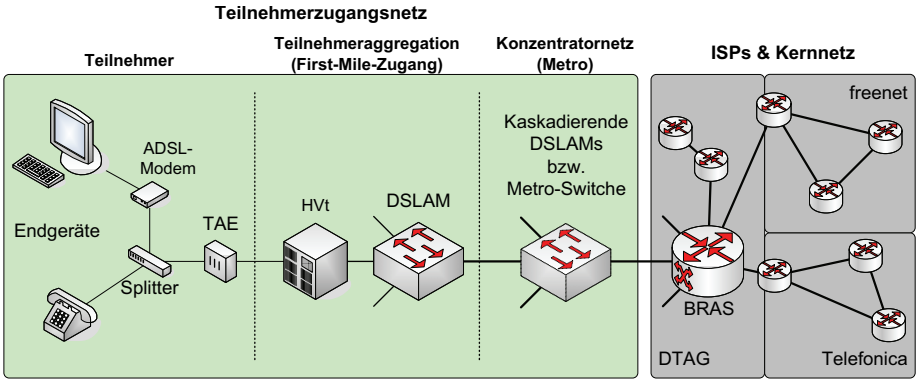


Abbildung 4.2.: Allgemeine Struktur von Teilnehmerzugangsnetze

Aufgrund regionaler Abweichungen, technologiebedingter Unterschiede in der Struktur und zum besseren Verständnis beziehen sich die nachfolgenden Betrachtungen jedoch auf eine vereinfachte, generische Strukturierung der TZN, wie sie in Abbildung 4.3 gezeigt ist. Es werden drei Bereiche unterschieden:

1. Der Teilnehmerbereich beinhaltet alle technischen Geräte direkt beim Kunden, welche zusammengefasst als CPE bezeichnet sind.

2. Der Bereich der TZN besteht aus generischen Zugangsknoten (DSLAMs). Diese Zugangsknoten sind die erste Aggregationsstufe und beinhalten mehrere Linecards zur physikalischen Teilnehmeranbindung, z. B. auf Basis von xDSL, und eine Zentralkarte zur Bündelung der Linecards und breitbandigen Anbindung an den BRAS.

3. Die Breitbandzugangsknoten (BRAS) des Kernbereichs sind die zweite Aggregationsstufe und stellen für gewöhnlich den ersten Hop auf IP-Ebene dar.

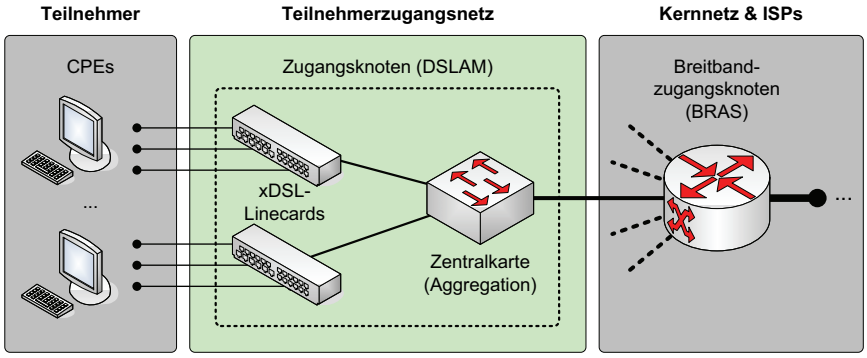


Abbildung 4.3.: Vereinfachte Struktur Ethernet-basierter Teilnehmerzugangsnetzen

Aufgaben von TZN Aufgrund ihrer Brückenfunktion zwischen Teilnehmern und Kernnetz müssen TZN komplexe Aufgaben auf technologischer Ebene sowie im Bereich der Sicherheit und QoS erfüllen. Auf der technologischen Ebene sind vor allem die Bereitstellung von Konnektivität und die Unterstützung verschiedener DSL-Varianten notwendig. Die TAL muss zur Anbindung des Teilnehmers an das Internet mit Sitzungsparametern wie IP-Adressen konfiguriert werden. Im Upstream müssen DSLAMs eine Vielzahl von TALs zu einem gemeinsamen, hochvolumigen Datenstrom bündeln (Multiplexing). Ein BRAS aggregiert mehrere DSLAM-Anbindungen. Im Downstream sind die individuellen Daten wieder auf die einzelnen Teilnehmer aufzusplitten (Demultiplexing).

Die komplexesten Aufgaben stehen jedoch in Zusammenhang mit QoS . Individuelle Dienstgütevereinbarungen und Richtlinien müssen ausgewertet und mit dem tatsächlich vom Nutzer erzeugten Datenverkehr verglichen werden, um mittels Mechanismen zum Verkehrsmanagement wie Policing und Shaping diese Dienstgütevereinbarungen zu erzwingen. Typische Aufgaben der Hauptknotenpunkte DSLAM und BRAS vermischen sich dabei immer mehr, was zum einen Teil auf Kostenfragen und zum anderen Teil auf technologischen Gegebenheiten beruht. In [DSL03, DSL04a, SGS05, Agi06] sind Detailinformationen zu Aufgaben und Anforderungen dieser Breitbandzugangsknoten zu finden.

Auf Ebene der Sicherheit werden zunehmend mehr Funktionen auf DSLAM und BRAS verteilt, um Bedrohungen bereits im TZN erkennen und abwehren zu können, wie z. B. mittels Firewalls und IDS. Aber auch die sichere Authentifizierung und Autorisierung von Nutzern sind sicherheitskritische Aufgaben der TZN. Der im nächsten Abschnitt vorgestellte Mechanismus bezieht sich u. a. auf ausgewählte Sicherheitsaspekte.

4.2. MAC Address Translation – Schicht-2-Adressumsetzung

Motivation Der Übergang von ATM zu Ethernet in TZN ist vor allem durch die geringen Kosten und Einfachheit von Ethernet getrieben. Natives Ethernet bietet jedoch nicht die notwendigen Level an Funktionalität, Skalierbarkeit und Sicherheit. Zusätzliche Mechanismen sind erforderlich. Es wird deshalb ein Mechanismus zur Umsetzung von Ethernet-MAC-Adressen vorgestellt – MAT (MAC Address Translation) – der durch Anforderungen an Sicherheit und Skalierbarkeit in Ethernet-basierten TZN motiviert ist, wie die nachfolgenden Beispiele zeigen.

IEEE Std 802.3 bietet nur eine flache Adresshierarchie, welche zwar innerhalb eines abgeschlossenen LANs adäquat ist, jedoch in den MAN- und WAN-Bereichen der TZN und Kernnetze aufgrund der hohen Teilnehmerzahlen nicht skaliert. Im Bereich der Kernnetze müssen diese steigenden Teilnehmerzahlen und damit eine zunehmende Adressmenge im Bereich einiger Hunderttausend unterstützt werden können, da sich dort der Datenverkehr letztendlich konzentriert. Netzgeräte mit Switching- und Routing-Funktionalität unterhalten umfangreiche Adresstabellen, um Transportinformationen für bekannte Adressen abzulegen. Dies erfordert zum einen eine effiziente und skalierbare Verwaltung dieser Tabellen. Zum anderen müssen Tabellenüberläufe, sogenannte Address Table Explosions [CGEDC+04], verhindert werden. Diese Überläufe, bedingt durch den limitierten physikalischen Speicher der Netzwerkgeräte, können zudem durch das mutwillige Überfluten (Flooding) eines Gerätes mit gefälschten Adresseinträgen erwirkt werden. Anfällige Geräte, insbesondere Schicht-2-Switche, schalten dann in einen transparenten Modus (Failopen Mode), der das unerlaubte Abhören fremder Informationen ermöglicht.

Ein weiteres, inhärentes Problem ist die selbständige, dynamische Aktualisierung dieser Adresstabellen und das automatische Erlernen neuer Adresseinträge durch die Geräte selbst. Dies betrifft die FDB (Filtering bzw. Forwarding Database) von Switches als auch den lokalen ARP-Cache (Address Resolution Protocol [RFC0826, RFC0903]) jedes angeschlossenen Teilnehmers. Diese Automatismen sind zum einen unerlässlich, da die Tabellen aufgrund ihrer Komplexität und hohen Dynamik nicht mehr per Hand zu administrieren sind. Auf der anderen Seite sind diese Hintertüren zum Ziel böswilliger Angriffe geworden, welche den Zweck haben, Adressinformationen zu manipulieren (Poisoning), fremde Identitäten vorzutäuschen (Spoofing) und unbefugt an sensible Informationen zu gelangen (Sniffing). Ethernet-basierte Netze weisen somit Sicherheitsrisiken auf, die den Richtlinien in [DSL06] zufolge unterbunden werden müssen.

Das Auftreten doppelt oder mehrfach vergebener MAC-Adressen innerhalb einer Ethernet-Domäne ist ebenfalls ein kritischer Punkt. Dies kann durch Fehlkonfiguration der Adresse beim Hersteller geschehen oder durch vom Teilnehmer konfigurierte Adressen, da von gängigen Netzwerkadaptern eine Umkonfiguration i. Allg. unterstützt wird. Das Resultat sind Fehlfunktionen im Netzwerk und Nichtverfügbarkeit von Diensten.

Aufgrund der peripheren Position von MAT in den TZN am Rand des Internets, worauf im Abschnitt 4.2.2 noch genauer eingegangen wird, sind sowohl die Adresskonzentration als auch die Datenraten relativ gering. Dadurch kann der Datenverkehr im TZN unter weniger restriktiven Randbedingungen vorverarbeitet und für die folgenden Aggregationsstufen im Kernbereich aufbereitet werden, um dort die Komplexität der Adresstabellen zu reduzieren als auch die Sicherheit zu erhöhen.

4.2.1. Allgemeine Funktionsweise der MAT

Der MAT-Mechanismus ersetzt MAC-Adressen der Teilnehmer innerhalb von Ethernet-Frames (Customer-MACs, CMACs) durch vom Provider definierte MAC-Adressen (Provider-MACs, PMACs) und führt ebenfalls den Rücktausch in Gegenrichtung durch [KWD+06, WKTB06]. Ein wichtiges Merkmal an dieser Stelle ist das echte *Ersetzen* im Gegensatz zu einer weiteren Kapselung. Im Upstream wird jeweils die Quell-Adresse (Source MAC, SRC MAC) durch eine PMAC ersetzt. Im Downstream wird die Ziel-Adresse (Destination MAC, DST MAC) durch die ursprüngliche CMAC ersetzt. Die MAT-Funktionalität teilt somit das Netz in die logischen Bereiche *Teilnehmerzone* und *Providerzone*, wie Abbildung 4.4 zeigt. MAT ist vom Prinzip her vergleichbar mit NAT (Network Address Translation) [RFC3022], welche vor allem aus Gründen der Skalierung privat verwaltete IP-Adressen mit öffentlichen, globalen IP-Adressen maskiert.

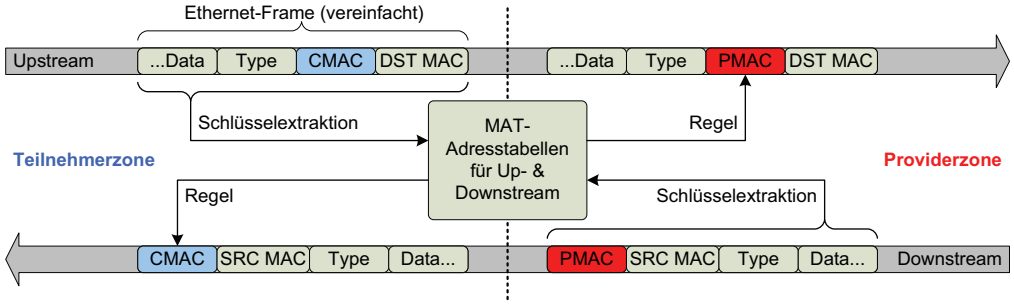


Abbildung 4.4.: Prinzip der MAC Address Translation

Um das Austauschen der MAC-Adressen konsistent für alle angeschlossenen Teilnehmer zu realisieren, müssen diese anhand ausgewählter, eindeutiger Parameter, sogenannter Schlüssel, identifiziert werden. Aus jedem eintreffenden Frame werden deshalb die benötigten Schlüssel extrahiert. Der Gesamtschlüssel wird dann zur Suche in einer MAT-Adresstabelle genutzt, welche den für diesen Frame konfigurierten Rückgabewert, die sogenannte Regel, zurückliefert. Im Upstream wird auf diese Weise die zu einem Schlüssel gehörende, einzigartige PMAC ermittelt. Im Downstream wird die zu einem Schlüssel passende CMAC geliefert. Die Größe der

Adresstabelle ist zum einen durch die Anzahl der angeschlossenen Teilnehmer bestimmt und zum anderen durch die Position des MAT-Moduls innerhalb der Infrastruktur des TZN, worauf im Folgenden noch näher eingegangen wird. In jedem Fall hat sie handhabbare Dimensionen, da der hier vorgestellte MAT-Mechanismus ausschließlich für TZN vorgesehen ist und somit vor den ersten Hauptbündelungspunkten (BRAS) angesiedelt ist. Der Gesamtschlüssel zur Suche in diesen Tabellen kann flexibel aus verschiedenen Teilschlüsseln gebildet werden und hängt von der jeweiligen Netzwerkumgebung und den angedachten Anwendungsfällen ab. Typische Teilschlüssel sind u. a. MAC- und IP-Adressen sowie VLAN-IDs.

Sicherheitsaspekte Wie unter 4.2 beschrieben spielt Sicherheit eine wichtige Rolle in Ethernet-basierten TZN, da diese auf Schicht 2 anfällig für typische Bedrohungsszenarien sind.

Beim *MAC Address Spoofing* sendet ein Angreifer Ethernet-Frames mit gefälschten Absenderadressen, welche durch das automatische Lernen des Switches den zu dieser MAC-Adresse gehörenden Eintrag in der FDB verfälschen. Dadurch sendet der Switch den entsprechenden Datenverkehr über das manipulierte Port zum Angreifer. *MAC Address Flooding* ist eine Form des Spoofings, bei der nicht die Nutzung einer falschen Adresse, sondern das Überfluten mit vielen Adressen im Vordergrund steht. Das massenhafte Versenden von Frames mit unterschiedlichen MAC-Adressen (= „Fluten“) zielt auf den Speicherüberlauf des Netzwerkswitches ab, wodurch das Gerät in einen transparenten Modus schaltet und der komplette Datenverkehr an allen Ports sichtbar ist. *ARP Spoofing* nutzt hingegen eine Schwäche des ARP-Protokolls aus, welche in der automatischen Aktualisierung der ARP-Caches netzwerkfähiger Geräte besteht. Jedes Gerät, das Daten versenden will, benötigt die MAC-Adresse des Zielgeräts, welche per ARP-Anfragen bezogen werden kann. Dabei generiert ein Angreifer manipulierte ARP-Antworten, wodurch in den ARP-Cache des anfragenden Opfers gefälschte Einträge geschrieben werden. Das auf diese Weise manipulierte Gerät sendet seine Frames nun an den Angreifer. Weitere Informationen zu typischen Bedrohungsszenarien geben u. a. [Jos05] und [VP07].

Die zunehmende Beliebtheit von xDSL-Flatrate-Angeboten (siehe Abschnitt 3.2.3) und die Nutzung von DSL-Routern im Heimbereich resultiert in einer geringen Fluktuation der an den Zugangsports von Linecards auftretenden MAC-Adressen. MAC-Adressen sind zumeist immer am selben Zugangsport sichtbar und neue MACs treten normalerweise nur durch Hardwaretausch beim Teilnehmer auf. Deshalb sind die MAT-Adresstabellen von *statischer* Natur. Statisch bedeutet, dass durch manipulierte Frames bzw. ARP-Pakete Einträge in der MAT-Adresstabelle *nicht automatisch generiert oder aktualisiert* werden. Die Einträge sind vorkonfiguriert bzw. werden durch eine externe Funktionseinheit verwaltet und aktualisiert. Sollte ein Schlüssel bei der Suche in den Adresstabellen der MAT-Funktionalität keine Regel zurückliefern, so werden dieser Schlüssel und der Frame an die verwaltende Instanz geschickt, welche z. B. die Generierung

eines neuen Eintrages überprüft oder eine SNMP-Fehlermeldung (Simple Network Management Protocol) [RFC1157, RFC3410] erzeugt. Somit wird sichergestellt, dass jede MAC-Adresse einen zuverlässigen Eintrag in der Adresstabelle hat. Jede potentiell unzuverlässige CMAC wird entweder durch eine eindeutige, zuverlässige PMAC ersetzt oder geblockt, wodurch o. g. Bedrohungen eliminiert werden können. Zudem erlaubt die unter Abschnitt 4.2.2 vorgestellte dezentrale MAT-Version die Konfiguration der maximal zulässigen Anzahl an MAC-Adressen pro Zugangsport einer Linecard, wodurch Flooding-Angriffe direkt unterbunden werden. Weitere Informationen zur Schutzfunktion von MAT bietet Abschnitt A.2 im Anhang.

Ersetzungsstrategien Bei MAT werden MAC-Adressen ersetzt und nicht gekapselt. Zwei wesentliche Gründe dafür sind, dass Ethernet-Frames auf diese Weise ihre standardkonforme Struktur beibehalten (siehe Abbildung 2.5) und nicht unnötig vergrößert werden (Protokolloverhead). Drei verschiedene Ersetzungsstrategien werden unterstützt, die durch die Konfiguration der MAT-Adresstabellen implizit realisiert werden. In jedem Fall wird das Auftreten doppelter und manipulierter MAC-Adressen durch die Maskierung mit einer PMAC vermieden. Die Strategien können unter Beachtung der Konsistenz innerhalb der Adresstabellen auch parallel genutzt werden. Der eigentliche Aufwand bei MAT ist nicht die Ersetzung der MAC-Adressen an sich, sondern, ähnlich wie beim Routing (siehe Abschnitt 2.3), die Konfiguration der Einträge der statischen Adresstabellen. Abbildung 4.5 zeigt zum besseren Verständnis eine einfache MAT-Adresstabelle für den Upstream.

```
1   Gesamtschlüssel                  | Regel
2   Schlüssel 1        Schlüssel 2   | Flags  PMAC
3   CMAC (SRC MAC)     SRC IP        | WBDx
4  ------------------------------------------------------------
5   00:FF:D8:D9:10:6B  86.56.35.5    | 0010   00:DE:AD:BE:EF:01
6   00:0F:32:1A:B6:07  217.69.224.73 | 0000   00:DE:AD:BE:EF:02
7   00:BF:42:FD:08:E2  217.69.224.74 | 0000   00:DE:AD:BE:EF:02
8   00:11:D8:AB:C0:BD  145.8.201.57  | 1000   n/a
9   00:12:25:4F:24:D7  213.187.64.18 | 0100   n/a
10  ...
```

Abbildung 4.5.: Beispiel einer einfachen MAT-Adresstabelle für den Upstream

- Bei einer *1:1 Übersetzung* wird jeder CMAC eine individuelle PMAC zugewiesen, wie die linke Grafik in Abbildung 4.6a darstellt. Dieses Szenario bezieht sich hauptsächlich auf o. g. Sicherheitsaspekte, da es die absolute Anzahl der MAC-Adressen im Netz nicht verringert. Zeile 5 in Abbildung 4.5 ist ein Beispiel dafür. Der Gesamtschlüssel bestehend aus CMAC und SRC IP liefert die PMAC `00:DE:AD:BE:EF:01`.

- Bei einer *n:1 Übersetzung* wird einer Gruppe von CMACs eine gemeinsame eindeutige PMAC zugewiesen, wie die rechte Darstellung in Abbildung 4.6b zeigt. Diese Vorgehensweise bezieht sich sowohl auf Sicherheitsaspekte als auch auf Skalierbarkeit, da mehrere CMACs auf eine PMAC abgebildet werden. Der Skalierungsfaktor n ist dabei konfigurierbar. In Abbildung 4.5 ist dies anhand der Zeilen 6-7 verdeutlicht. Zwei unterschiedliche CMACs werden auf dieselbe PMAC `00:DE:AD:BE:EF:02` abgebildet. Für den Upstream würde in diesem Fall die CMAC als Schlüssel ausreichen. Jedoch wird im Downstream für einen korrekten Rücktausch ein Tupel aus der PMAC und mindestens einem weiteren eindeutigen Schlüssel, z. B. der IP-Adresse, benötigt.
- MAT bietet außerdem mit einer *partiellen Übersetzung* die Möglichkeit, nur bestimmte Bereiche bzw. Bits einer CMAC zu ersetzen. Dies kann hilfreich sein, wenn unterschiedliche Bereiche einer CMAC an verschiedenen Punkten im TZN oder anhand unabhängiger Schlüssel ersetzt werden soll. Dies ermöglicht, z. B. durch serielle Anordnung mehrerer MAT-Module hintereinander, eine hierarchische Strukturierung der PMAC, die dann aufgrund dieser Struktur indirekte Zusatzinformationen mit sich führt. In [WCL07] wurde kürzlich ein ähnlicher Ansatz einer Schicht-2-Adressumsetzung zur Realisierung von VPLS (Virtual Private LAN Services) vorgestellt. Die partielle Ersetzungsstrategie von MAT bietet dafür z. B. die nötige Hardwareunterstützung.

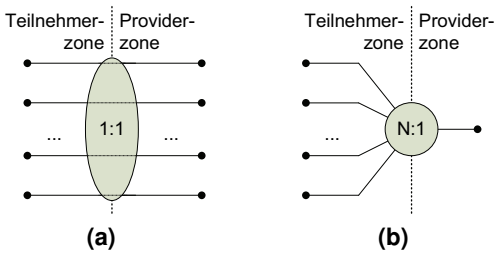


Abbildung 4.6.: (a) 1:1- und (b) n:1-Ersetzungsstrategien bei MAT

Ausnahmebehandlungen Neben der regulären Ersetzung der SRC MAC im Upstream bzw. der DST MAC im Downstream innerhalb des Ethernet-Headers kann in bestimmten Fällen eine andere Vorgehensweise zweckmäßig oder sogar notwendig sein. Dies wird mithilfe zusätzlicher Status-Bits pro Eintrag in den Adresstabellen kenntlich gemacht. Diese Status-Bits (Flags) sind in Abbildung 4.5 als erster Teil der Regel dargestellt. Momentan werden drei Bits genutzt (`W`, `B`, `D`) während das vierte Bit (`x`) für zukünftige Belange reserviert ist.

Einerseits können Ausnahmebehandlungen für bestimmte MAC-Adressen vorgesehen sein. Durch die ersten beiden Flags (`W`, `B`) werden sogenannte White- und Black-Lists realisiert, die z. B. für erste einfache Filtermaßnahmen geeignet sind:

- Weist die Regel ein gesetztes `W`-Flag (White-List) auf, wird der Frame unverändert weitergeleitet, ohne dass eine MAC-Adresse ersetzt wird (Zeile 8 in Abbildung 4.5).
- Weist die Regel ein gesetztes `B`-Flag (Black-List) auf, wird der entsprechende Frame an Ort und Stelle blockiert und verworfen (Zeile 9 in Abbildung 4.5).

Andererseits erfordern auch spezielle Verwaltungs- und Konfigurationsprotokolle eine Sonderbehandlung, da MAC-Adressen nicht nur im eigentlichen Ethernet-Header enthalten sind, sondern auch in den Nutzdaten mitgeführt werden. Um netzwerkweite Konsistenz zu garantieren, müssen diese Adressinformationen in bestimmten Fällen ebenfalls getauscht werden:

- Bei ARP und RARP [RFC0826, RFC0903], Protokollen zur Adressauflösung zwischen Vermittlungs- und Sicherungsschicht in IP4-Umgebungen, werden MAC-Adressen in den Nutzdaten der Vermittlungsschicht mitgeführt. In Abhängigkeit der Datenflussrichtung und des Befehlskodes im (R)ARP-Header sind an dieser Stelle ebenfalls Substitutionen durch MAT durchzuführen.
- Ähnlich verhält es sich bei DHCP (Dynamic Host Configuration Protocol) [RFC2131], welches die automatische Vergabe von IPv4-Adressen regelt. DHCP führt in seinem Header als sogenannte *chaddr* (Client Hardware Address) die Ethernet-MAC-Adresse mit, an welche die zu vergebende IP-Adresse gebunden wird. Diese muss in bestimmten Szenarien ebenfalls getauscht werden, was mit einem weiteren MAT-Status-Bit, dem `D`-Flag (DHCP-Flag) indiziert wird (z. B. Zeile 5 in Abbildung 4.5).
- Im Gegensatz zu IPv4 werden in IPv6-Umgebungen nicht (R)ARP und DHCP sondern ICMPv6 [RFC2463] und DHCPv6 [RFC3315] zur Adressauflösung und -vergabe genutzt. Eine entsprechende Behandlung dieser Protokolle wird deshalb in zukünftigen Netzen eine Rolle spielen. Die Adressvergabe kann bei IPv6 unterschiedlich erfolgen. Die statuslose Autokonfiguration erfolgt mittels ICMPv6 als sogenannter Neighbor Discovery-Prozess [RFC2461, RFC4862]. Neighbor Discovery wird darüber hinaus zur Adressauflösung zwischen Vermittlungs- und Sicherungsschicht genutzt. MAC-Adressen sind bei ICMPv6 als Option in Nachrichten vom Typ Router-Advertisement und -Solicitation sowie Neighbor-Advertisement und -Solicitation enthalten. Zudem verwenden autokonfigurierte IPv6-Adressen die MAC-Adresse als Teil der IPv6-Adresse, dem sogenannten *Interface Identifier* [Hag06, LJS06]. Die statusbehaftete Autokonfiguration erfolgt per DHCPv6, welches neben

der bloßen Adressvergabe zusätzliche Features bietet. Eine IPv6-Adresse ist dabei nicht direkt an eine MAC-Adresse sondern an einen *DHCP Unique Identifier* (DUID) gebunden. Die MAC-Adresse kann Teil des DUID sein. DUIDs und MAC-Adressen sind in verschiedenen DHCPv6-Optionen enthalten und in bestimmten Szenarien durch MAT zu tauschen.

Abbildung 4.7a zeigt zusammenfassend die Hierarchie der Protokolle, welche in IPv4- und IPv6-Umgebungen durch MAT gesondert behandelt werden müssen. Weitere Informationen zur Sonderbehandlung der genannten Protokolle sind in Abschnitt A.1 im Anhang enthalten.

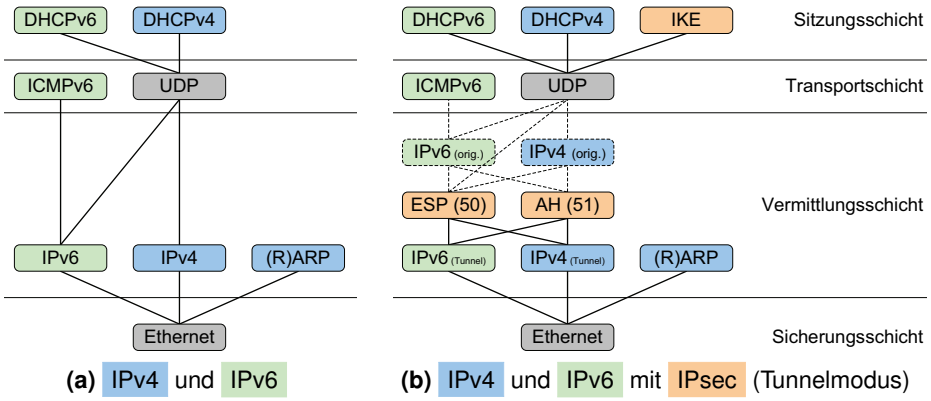


Abbildung 4.7.: Hierarchie verschiedener Protokolle mit Sonderbehandlung

Wechselwirkungen mit anderen Sicherheitsarchitekturen Für eine sichere und verschlüsselte Übertragung von Informationen existieren verschiedene Mechanismen. Transport Layer Security (TLS) [RFC4346, FKK96], auch bekannt unter Secure Sockets Layer (SSL), ist ein zwischen Transport- und Sitzungsschicht angesiedeltes Verschlüsselungsprotokoll. Es bestehen deshalb keine Abhängigkeiten zwischen MAT und TLS/SSL. Das komplexe Gerüst der Internet Protocol Security (IPsec) [RFC4301] ist jedoch eine Sicherheitsarchitektur auf Ebene der Vermittlungsschicht. MAT und IPsec können sich somit gegenseitig beeinflussen. IPsec wird hauptsächlich für Datensicherheit und Verschlüsselung genutzt und basiert auf den Protokollen Encapsulating Security Payload (ESP) [RFC4303] und Authentication Header (AH) [RFC4302], welche entweder im Tunnel- oder Transportmodus genutzt werden. Für die kryptografischen Funktionen von IPsec sind zudem Mechanismen zum sicheren Austausch von Schlüsseln über ein unsicheres Netzwerk erforderlich. Dazu wird Internet Key Exchange (IKE) [RFC4306] genutzt, welches wiederum von verschiedenen anderen Protokollen abgeleitet ist. Abbildung 4.7b ordnet IPsec in die Hierarchie der Protokolle ein, welche durch MAT gesondert behandelt

werden müssen. Ist IPsec aktiv, werden je nach Modus und Sicherheitsprotokoll zusätzliche Header in das IP-Paket eingefügt. Im Transportmodus werden IPsec-Header zwischen IP-Header und Nutzdaten eingefügt. Der originale IP-Header bleibt unverändert und dient weiterhin der Wegewahl. Im Tunnelmodus wird das ursprüngliche IP-Paket komplett durch IPsec gekapselt und ein neuer IP-Header auf der Vermittlungsschicht eingefügt. Der äußere IP-Header dient der Verbindung der Endpunkte des IPsec-Tunnels. ESP und AH können darüber hinaus auch kombiniert werden, wodurch die Komplexität der IPsec-Architektur deutlich wird.

Ein Ersetzen von MAC-Adressen sowohl im Ethernet-Header als auch bei (R)ARP hat keinen Einfluss auf IPsec. ICMPv6, DHCP und DHCPv6 sind jedoch Protokolle der Transport- und Sitzungsschicht und somit in der Nutzlast eines IP-Pakets enthalten, wodurch sich IPsec und MAT gegenseitig ausschließen können. Liegen die IPsec-Endpunkte jeweils innerhalb des Wirkungsbereichs von MAT, sind die Operationen von MAT völlig transparent für IPsec. Liegt der Wirkungsbereich von MAT jedoch zwischen den IPsec-Endpunkten, sind MAT und IPsec *nicht* kompatibel, da die durch MAT zu modifizierenden Felder durch IPsec geschützt sind. AH authentifiziert bestimmte Teile eines IP-Paketes, indem eine Prüfsumme über diese Bereiche ermittelt wird. Ausgenommen sind veränderliche Felder des IPv4- bzw. IPv6-Headers. Die Nutzdaten, in welchen auch die o. g. Protokolle ICMPv6, DHCP und DHCPv6 gekapselt sind, werden in jedem Fall in die Prüfsummenberechnung einbezogen, jedoch nicht verschlüsselt übertragen. ESP hingegen verschlüsselt die Nutzdaten und sorgt somit für Datenschutz und Authentifizierung. In beiden Fällen resultiert eine nachträgliche Modifikation von MAC-Adressen durch MAT im Verlust der Datenintegrität aus Sicht von IPsec. Aus verschiedenen Gründen spielt diese Inkompatibilität jedoch eine untergeordnete Rolle. Einerseits existiert mit NAT z. B. ein anderes verbreitet genutztes Verfahren in IPv4-Umgebungen, welches nicht mit IPsec kompatibel ist. Andererseits müssen sowohl die Konfiguration von IPv4-Adressen als auch die Autokonfiguration von IPv6-Adressen erfolgen, bevor ein IPsec-Sitzung etabliert werden kann. IPsec nutzt IKE zur Aushandlung von Sitzungsparametern und zum Austausch von Schlüsseln. IKE nutzt UDP, welches wiederum IP-basiert arbeitet. Dies bedeutet, IP-Adressen müssen bereits konfiguriert und vergeben sein. Zudem verändern sich während einer bestehenden IPsec-Verbindung die Adressen der Sicherungs- und Vermittlungsschicht normalerweise nicht.

Darüber hinaus ist der Einsatz von IPsec nach wie vor strittig. Auch wenn IPsec das Internet Protokoll momentan am besten absichert, wird es wegen seiner Komplexität und des umfangreichen Konfigurationsaufwandes, der damit einhergehenden Fehleranfälligkeit sowie der hohen Rechenlast und des Paket-Overheads gemieden [Bei06]. Die einzige praktisch relevante Applikation zurzeit ist VPN, welche jedoch auch mit anderen Ansätzen wie dem o. g. TLS realisiert wird. Für verschiedene andere Anwendungen haben sich einfachere alternative Sicherheitsarchitekturen und Werkzeuge durchgesetzt, welche in den höheren anwendungsbezogenen Schichten

angesiedelt sind und verbreitet Einsatz finden. Beispiele dafür sind S/MIME [RFC2311], PGP, oder Kerberos [RFC4120], auf die an dieser Stelle jedoch nicht weiter eingegangen wird.

4.2.2. Zentraler & dezentraler Ansatz

Die Unterscheidung einer zentralen und einer dezentralen MAT-Variante ist primär durch mit dem Industriepartner Nokia Siemens Networks diskutierte, ökonomische Aspekte motiviert. Die Zentralkarte eines DSLAMs ist weitaus weniger kostensensitiv als die Linecards, da die Herstellungs- und Wartungskosten auf eine große Zahl an Nutzern heruntergebrochen werden können. Bei Linecards, die üblicherweise in Dollar pro Port gehandelt werden, erlaubt der Preisfaktor einen wesentlich eingeschränkteren Spielraum.

Zentrale Position Beim zentralen Ansatz ist die MAT-Funktionalität logisch zwischen DSLAM und BRAS angesiedelt. Physikalisch gesehen ist sie am Uplink-Port des DSLAMs auf der Zentralkarte positioniert. Abbildung 4.8 zeigt diese Anordnung. Die zentrale Variante verarbeitet am Uplink-Port gebündelten Datenverkehr in der Größenordnung mehrerer Gbit/s. Um diese Menge an Daten blockierungsfrei verarbeiten zu können, ist eine Realisierung in Hardware sinnvoll, da der Kostendruck auf der Zentralkarte weniger kritisch ist. Die Gesamtfunktionalität umfasst alle eingangs genannten Aspekte. Die MAT-Adresstabellen sind durch den jeweiligen ISP administriert. Dies kann z. B. mithilfe der für gewöhnlich im Zugangsknoten vorhandenen CPU durchgeführt werden. Dabei können für verschiedene Dienstanbieter separate PMAC-Adressbereiche verwaltet werden.

Ein weiterer Grund für die Implementierung auf der Zentralkarte ist die Integration in das in [WKTB06] und [KWD+07] vorgestellte MATMUNI-System. Dabei stellt MAT neben einem Traffic-Manager [KWD+06] und einem MPLS-Modul [WKD+06] nur einen Teil der Gesamtfunktionalität dieses paketverarbeitenden Systems dar.

Dezentrale Position In der Arbeit von Spies [Spi06] wurde die dezentrale MAT-Version unter der Bezeichnung sMAT (simplified MAT) aus der zentralen Variante abgeleitet und spezifiziert sowie in [KWT+07] vorgestellt. Dezentral bedeutet, dass MAT in vereinfachter Form und unabhängig von dem erwähnten MATMUNI-System auf den Linecards des ersten Zugangsknotens aus Sicht der Teilnehmer angesiedelt ist, wie Abbildung 4.9 darstellt. Die MAT-Funktion befindet sich dabei hinter den Aggregationseinheiten der Linecards. Durch diese Anordnung wird eine frühestmögliche Maskierung und Skalierung unorganisierter CMACs mit administrierten, strukturierten PMACs erreicht.

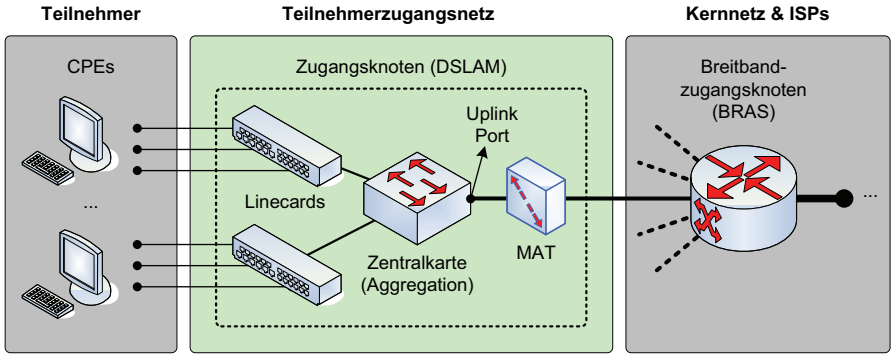


Abbildung 4.8.: Zentrale Position der MAT-Funktionalität im Zugangsknoten im TZN

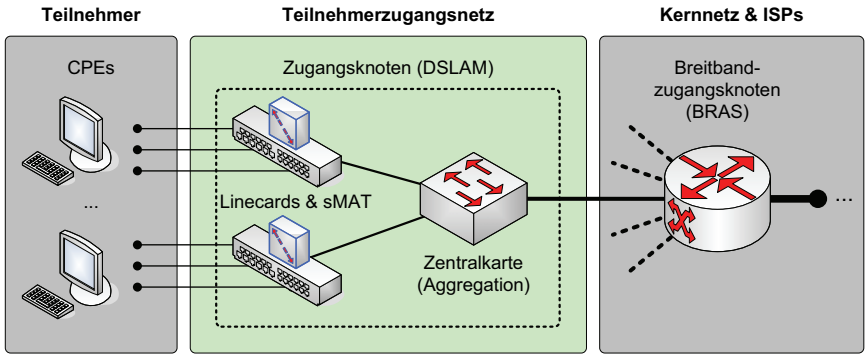


Abbildung 4.9.: Dezentrale Position der sMAT-Funktionalität auf den Linecards eines Zugangsknotens im TZN

Da Linecards kostensensitive Komponenten sind, ist es jedoch erforderlich, den Funktionsumfang von sMAT den Gegebenheiten auf der Linecard anzupassen. Von Vorteil ist, dass Informationen zu den jeweiligen Zugangsportnummern der Teilnehmer als zusätzlicher Teilschlüssel genutzt werden können, da diese *nur* auf den Linecards vorhanden sind. Pro Zugangsport ist damit eine maximal zulässige Anzahl an CMACs konfigurierbar, wodurch MAC Flooding direkt verhindert wird. Da Linecards keine echte Switching-Funktion besitzen, sind sie von MAC-Flooding nicht direkt betroffen. Jedoch ist die Zentralkarte eines DSLAMs praktisch eine vollständige Switching-Einheit, die anfällig für MAC-Flooding ist.

Da jede Linecard ein sMAT-Modul erfordert, ist der Verwaltungsaufwand für alle Adresstabellen zu minimieren. Dazu wird pro Linecard eine Basisadresse festgelegt. In Abhängigkeit der konfigurierten Anzahl von CMACs pro Port werden die PMACs per Offset auf diese Basisadresse vergeben. Zum einen kann dadurch die Struktur der sMAT-Adresstabellen und die Suche in ihnen

vereinfacht werden. Zum anderen haben die Tabellen damit eine feste Größe und erfordern keine Verwaltung während des Betriebes. Die Einträge in der Tabelle werden bis zur konfigurierten Anzahl pro Port automatisch erlernt, wofür ein Alterungsmechanismus genutzt wird [Spi06]. Die Konfiguration der Basisadressen durch den ISP muss konsistent sein, damit jedes sMAT-Modul einen eindeutigen PMAC-Bereich innerhalb der Netzwerkdomäne des ISPs verwaltet.

Da an den Zugangsports der Linecards ausschließlich MAC-Adressen der jeweiligen Teilnehmer auftreten können, wird als weitere Vereinfachung auf die Filterung (White- und Black-List) bestimmter MAC-Adressen verzichtet. Jede CMAC aus Richtung des Teilnehmers wird übersetzt. Dazu wird ein statischer Suchschlüssel aus SRC MAC, SRC IP sowie Zugangsportnummer genutzt (im Downstream DST MAC und DST IP).

4.2.3. Abgrenzung von MAT zu anderen Mechanismen

Verschiedene Mechanismen existieren, um Ethernet auch über LAN-Grenzen hinaus effizient einsetzen zu können, z. B. innerhalb der MAN/WAN-Bereiche der Kernnetze. Relevante Vertreter sind MPLS (Multi Protocol Label Switching) [RFC3031] und IEEE Std 802.1ah [IEE08], auch bekannt unter der Bezeichnung MAC-in-MAC (MiM) [Nor07c, Nor07b]. Zum einen sind beide Ansätze primär aus den Notwendigkeiten heraus entstanden, im Kernsegment OAM-Dienste für natives Ethernet bereitzustellen sowie Dienste und Teilnehmergruppen differenziert behandeln zu können, z. B. auf Basis virtueller Verbindungen (Tunnel) wie VPN und VPLS. Dabei werden i. Allg. separate, verteilte Teilnetze zu einem logischen LAN zusammengeschlossen, z. B. einem weitflächigen Firmen-Intranet. Zum anderen verringern MiM und MPLS auch die Komplexität der FDBs, jedoch nur im Bereich der Kernnetze.

Sogenanntes MAC Address Stacking (MAS) kapselt einen Ethernet-Frame in zwei weitere Felder für SRC PMAC und DST PMAC [CGEDC+04, RHPG04]. Diese PMACs werden ausschließlich im Kernnetz zur Wegewahl genutzt. Mit MiM (IEEE Std 802.1ah) wurde ein Ansatz vorgestellt, der MAS mit QiQ (siehe Abschnitt 2.4) kombiniert. MiM kapselt Ethernet-Frames in einem vollständigen Ethernet-Header, wodurch Frames um 12-20 Byte vergrößert werden. Dieser MiM-Header ist komplett durch den ISP administriert und wird in MiM-fähigen Rand-Routern des Kernnetzes eingefügt bzw. entfernt. Der MiM-Header enthält neben SRC & DST PMAC zwei zusätzliche Q-Tags (Backbone-VLAN-ID & Backbone-Service-ID) zum Verkehrsmanagement im Kernbereich. MiM erfordert eine funktionale Anpassung der Knoten im Kernnetz.

MPLS ist ebenfalls eine Form der Kapselung mit dem Ziel, PDUs verbindungsorientiert und technologieunabhängig (Ethernet, ATM, Frame Relay) über ein paketvermitteltes IP-Kernnetz zu tunneln. Sogenannte MPLS-Labels werden in LERs (Label Edge Router) zwischen Ethernet- und IP-Header eingefügt bzw. entfernt. Anhand dieser Labels kann ein Frame effizient per

LSRs (Label Switched Router) über vordefinierte Pfade, sogenannte LSPs (Label Switched Path), weitergeleitet werden, wodurch die Größe der FDBs im Kernbereich veringert werden *kann*. Nachteile von MPLS sind der Aufwand zur Generierung der Labels, deren Verbreitung und Aktualisierung im gesamten MPLS-Kernnetz per LDP (Label Distribution Protocol) sowie die erforderlichen Anpassungen von Hard- und Software. Dadurch, dass die MPLS-Labels per LDP erst netzwerkweit bekannt gemacht werden müssen, basiert MPLS auf Routing. Im Gegensatz dazu beruhen MiM und MAT ausschließlich auf der (automatischen) Aktualisierung von FDBs.

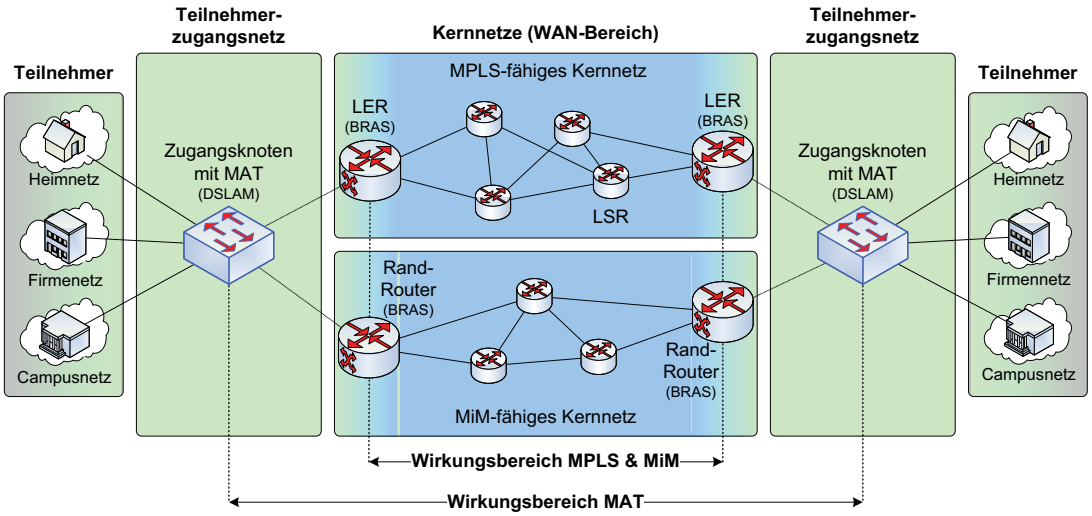


Abbildung 4.10.: Wirkungsbereiche der Mechanismen MiM, MPLS (blau) und MAT (grün). Üblicherweise wird entweder MPLS oder MiM im Kernbereich eingesetzt.

Abbildung 4.10 verdeutlicht die Wirkungsbereiche von MAT, MiM und MPLS. Während MiM und MPLS an den Grenzen der Kernnetze terminieren (blau), siedelt sich die MAT-Funktionalität im Bereich der TZN an und umschließt die Kernnetze. MPLS und MiM verringern die Komplexität der FDBs ausschließlich im Kernbereich. MAT hat jedoch nicht nur Einfluss auf den Kernbereich, sondern auch auf Adresstabellen im TZN und die sichtbaren MAC-Adressen im Teilnehmerbereich. Dies resultiert aus dem Ersetzen von SRC MACs durch PMACs im Upstream und dem Rücktausch von DST MACs durch CMACs im Downstream. Zwei Kommunikationsendpunkte sehen nur die PMAC des Gegenübers und nie die CMAC. In diesem Zusammenhang ist MAT kein „Konkurrent“ zu MPLS bzw. IEEE Std 802.1ah, sondern unterstützt diese vielmehr. Gründe dafür sind:

- MAT ist völlig transparent in die Netzwerkinfrastruktur integriert. Der Wirkungsbereich von MAT umschließt die Kernnetze, in denen MiM und MPLS Anwendung finden.

- Die PMACs für MAT und MiM sind separat verwaltet und voneinander unabhängig.
- Das echte Ersetzen von MAT bewahrt die Standardstruktur von Frames (siehe Abbildung 2.5). Diese können später durch MiM und MPLS problemlos weiterverarbeitet werden.
- MAT bereitet den Datenverkehr für die Folgestufen auf und skaliert durch eine flexible n:1-Strategie den Adressraum bereits im Vorfeld und somit auch für MiM bzw. MPLS.

4.2.4. Zusammenfassung der Eigenschaften von MAT

MAT bereitet an einer peripheren Position im Internet Datenverkehr in Richtung der Kernnetze auf und dient auf der einen Seite der Skalierung der Größe der Adresstabellen in den Teilnehmerzugangsnetzen und Kernnetzen der Netzbetreiber. Dadurch werden sowohl das Risiko von Tabellen-Überläufen als auch die Leistungsanforderungen an Netzwerkknotenpunkte reduziert. Auf der anderen Seite können verschiedene Sicherheitslücken auf Ebene der Sicherungsschicht geschlossen werden, wodurch auch darauf aufsetzende Angriffsmöglichkeiten zumindest erschwert werden, z. B. DNS-Spoofing. Der MAT-Mechanismus ermöglicht eine Klassifizierung und Strukturierung des Datenaufkommens im Netz und erhöht die Kontrolle für den Betreiber. Die folgende Liste ist eine Zusammenfassung der wichtigsten Eigenschaften des MAT-Konzepts.

- Durch konsistente Übersetzung der CMACs beim Eintritt ins Netzwerk und durch Limitierung der MAC-Adressen pro Port werden MAC-Spoofing, MAC-Flooding und ARP-Spoofing verhindert sowie das Auftreten doppelter MAC-Adressen unterbunden. Dadurch werden TZN und Kernnetze robuster gegenüber Attacken bzw. Fehlverhalten.
- Durch die Unterstützung verschiedener Ersetzungsstrategien (1:1, n:1, partiell) kann der Adressraum in der Providerzone skaliert werden und ungeordnete CMACs werden durch (ggf. hierarchisch) strukturierte PMACs maskiert.
- MAT und sMAT sind vollständig transparent bzgl. der Teilnehmer- und Providerzone.
- Das Ersetzen der MAC vermeidet zusätzlichen Protokolloverhead in den Ethernet-Frames.
- Im Gegensatz zu einer Kapselung wird die ursprüngliche Struktur der Ethernetframes beibehalten (siehe Abbildung 2.5). Durch die Konformität zu IEEE Std 802.3 sind keine funktionalen Erweiterungen in existierenden Netzwerkgeräten erforderlich.
- Durch Black- & White-Listen können in der zentralen MAT-Variante erste, einfache Filter realisiert werden. Diese können nicht nur ausschließlich auf Basis von MAC-Adressen filtern, sondern auch andere Teilschlüssel wie IP oder VLAN nutzen.

- Aufgrund seiner Einfachheit und der Regelmäßigkeit des Ersetzungsvorganges sind MAT und sMAT für eine Hardware-Implementierung geeignet, um eine blockierungsfreie Verarbeitung der Daten zu gewährleisten.
- MAT ist flexibel konfigurierbar. sMAT kann darüber hinaus während der Laufzeit konfigurationsfrei betrieben werden.
- MAT unterstützt bzw. ist kompatibel zu Mechanismen wie MiM bzw. MPLS.

Der MAT-Mechanismus ist eine einfache, kostengünstige Erweiterung für TZN. Gerade im hart umkämpften Markt für Linecards und DSLAMs stellt MAT ein interessantes Feature für Hersteller von Netzwerkgeräten dar, um sich gegenüber der Konkurrenz zu differenzieren. Ergebnisse des MAT-Projektes wurden in [KWD+06], [WKTB06] und [KWT+07] veröffentlicht.

4.3. MATMUNI – Paketverarbeitung im Teilnehmerzugangsnetz

Der Name *MATMUNI* setzt sich aus den Teilfunktionen MAC Address Translation (MAT), Traffic Management (TM) und MPLS-User-to-Network-Interface (MPLS-UNI) zusammen, welche in einem Gesamtsystem realisiert wurden. Der Zweck des MATMUNI-Systems ist, Daten bereits im TZN vor dem Eintritt in den Kernbereich des Internets aufzubereiten, um in den Kernnetzen die Komplexität der Paketverarbeitung zu reduzieren (siehe Abschnitt 4.1). Die Umsetzung von MATMUNI ist primär durch konkrete industrielle Anforderungen getrieben. Einerseits müssen ein bis vier bidirektionale GbE-Kanäle mit einer maximalen Gesamtdatenrate von bis zu 8 Gbit/s verarbeitet werden können. Andererseits ist eine transparente Integration in die vorhandene Netzinfrastruktur erforderlich. Aus Kosten- und Flexibilitätsgründen ist zudem die Nutzung eines FPGAs als Zielplattform vorgesehen, denn gerade FPGAs bieten als feldprogrammierbare Hardwarebausteine in der Telekommunikation eine flexible Basis, um den ständig wechselnden Anforderungen, Funktionen und Parametern zu genügen. Der Einsatz von FPGAs insbesondere im Bereich der TZN ist mittlerweile üblich und wird durch die führenden Hersteller unterstützt [ALT, XWC]. Eine weitere entscheidende Anforderung ist die logische Platzierung des MATMUNI-Systems zwischen DSLAM und BRAS. Dies veranschaulicht Abbildung 4.11. Physikalisch ist MATMUNI auf der Zentralkarte des DSLAMs integriert.

4.3.1. Aufbau und Funktionsweise des MATMUNI-Systems

Abbildung 4.12 zeigt ein Blockschaltbild der Architektur des MATMUNI-Systems. Deren Hauptmerkmale sind synchrone gepipelinete Datenpfade auf Basis von Punkt-zu-Punkt-Verbindungen

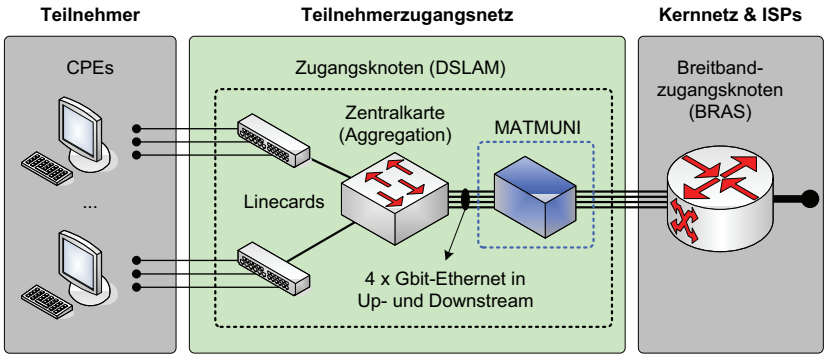


Abbildung 4.11.: Position von MATMUNI im Teilnehmerzugangsnetz

zwischen den einzelnen Teilmodulen. Pro GbE-Kanal existieren zwei Hauptdatenpfade zur Weiterleitung der Ethernet-Frames für den Up- und Downstream (blaue Pfeile). Sie sind 8 Bit breit, da GbE-Schnittstellen mit 8 Bit/Takt bei 125 MHz definiert sind. Dazu orthogonal verlaufen Kontrollsignale zur Steuerung des Arbeitsablaufs (graue Pfeile). Pro GbE-Kanal werden die hinterlegten Komponenten repliziert. Die funktionalen Module (blau) sind seriell verschaltet. Ein Frame durchläuft alle Funktionsmodule (FMs) einer Datenflussrichtung, bevor er das System wieder verlässt. Die korrekte Abarbeitung aller Funktionen wird durch die Systemmodule (hellgrün) realisiert. Die grobe Funktionsweise von MATMUNI entspricht dem typischen Ablauf in der Paketverarbeitung (siehe Abschnitt 2.3): Empfangen und Zwischenspeichern von Ethernet-Frames, Klassifizierung durch Schlüsselextraktion und Suche in einem Speicher, Ausführung von Aktionen in den FMs gemäß der im Speicher abgelegten Regeln sowie Weiterleitung über einen dedizierten Ausgangskanal. Im Folgenden ist die prinzipielle Funktionsweise der einzelnen Komponenten kurz erläutert. Weitere Details bieten die Publikationen [WKTB06] und [WKT07].

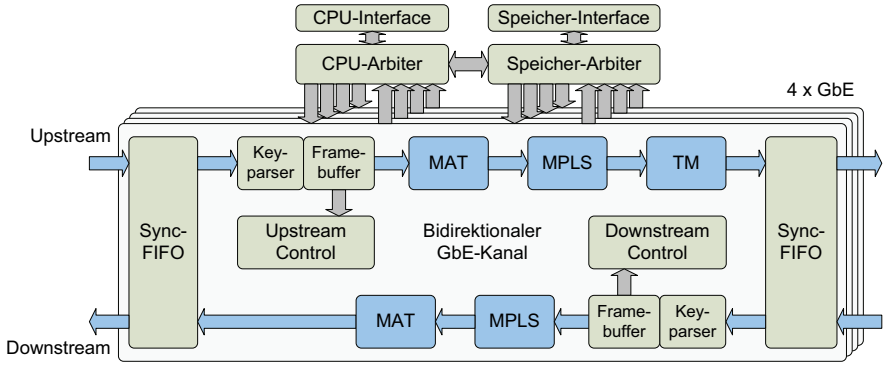


Abbildung 4.12.: Architektur des MATMUNI-Systems am Beispiel eines GbE-Kanals

MAT Die zentrale MAT-Variante (Abschnitt 4.2) wurde in vollem Umfang implementiert. Für Up- und Downstream wurden separate Module entworfen.

MPLS-UNI Über die Funktion von MPLS berichtete bereits Abschnitt 4.2.3. Im Upstream wird in jeden Ethernet-Frame ein MPLS-Label-Stack eingefügt, während im Downstream alle MPLS-Labels entfernt werden [WKD+06]. Dazu wird die Kapselungsvariante nach Martini genutzt [RFC4447, RFC4448]. Die Anordnung von MAT und MPLS-UNI ist an Abbildung 4.10 angelehnt, da MAT den Wirkungsbereich von MPLS umschließt.

TM In Abschnitt 2.3 wurde das Verkehrs- und Bandbreitenmanagement bereits als eine der Hauptaufgaben der Paketverarbeitung bezeichnet. Die in MATMUNI integrierte TM-Funktion basiert auf [RFC2697, RFC2698] und ist in [KWD+06] genauer beschrieben.

Sync-FIFO Im GbE-Betrieb werden Ethernet-Frames byteweise mit einer Taktfrequenz von 125 MHz empfangen bzw. gesendet. Der interne Arbeitstakt von MATMUNI ist jedoch von den externen Schnittstellen unabhängig. Die Sync-FIFOs werden benötigt, um Datensignale konsistent und fehlerfrei zwischen den verschiedenen Taktdomänen zu synchronisieren.

Framebuffer und Keyparser Während Frames in den Framebuffer geschrieben werden, durchlaufen sie den Keyparser, welcher aus jedem Frame für eine Suche im Speicher relevante Informationen wie MAC- und IP-Adressen extrahiert. Diese Schlüssel werden dem Speicher-Arbiter als Suchanfrage übergeben. Frames werden solange im Framebuffer zwischengespeichert, bis der Speicher-Arbiter die jeweilige Anfrage bearbeitet hat. Ein Frame wird dann entweder im Datenpfad zu den funktionalen Modulen weitergeleitet (Normalfall) oder über das entsprechende Control-Modul an die CPU gesendet (Sonder-/Fehlerfall).

Speicher-Arbiter Die Koordination von Speicherzugriffen ist Aufgabe des Speicher-Arbiters, da mehrere Teilmodule gleichzeitig lesend und schreibend auf den Speicher zugreifen (die CPU und maximal 8 unidirektionale GbE-Datenpfade). Grundlage des Arbiters ist der u. a. in [Loc86, Hil04] untersuchte Least-Laxity-First-Algorithmus. Ein Suchergebnis (Regel) wird dem Framebuffer und allen FMs übergeben. Eine neue Regel kann erst übergeben werden, wenn die Bearbeitung des aktuellen Frames im letzten FM abgeschlossen ist.

CPU-Arbiter Die Verwaltung der Zugriffe auf die CPU ist die primäre Aufgabe des CPU-Arbiters. Zudem ist er für die Verarbeitung der von der CPU kommenden Konfigurationsdaten für das Gesamtsystem verantwortlich.

Control-Modul Up- und Downstream verfügen jeweils über ein Control-Modul als separater Zwischenspeicher für fehlerhafte Frames bzw. für Frames mit unbekannter Schlüsselmenge.

Diese werden zur weiteren Analyse über den CPU-Arbiter an die CPU gesendet.

Das MATMUNI-System besteht in seiner größten Ausbaustufe aus den drei Teilfunktionen MAT, TM und MPLS-UNI sowie 4 bidirektionalen GbE-Kanälen. Die Anzahl der GbE-Kanäle und die zu instanziierenden Funktionen sind zur Synthesezeit konfigurierbar. Ein voll funktionsfähiger Prototyp des MATMUNI-Systems auf Basis eines FPGA-Entwicklungsboards [ML405] konnte in [KWD+07] demonstriert werden. Zusätzliche Informationen enthält Anhang A.3.

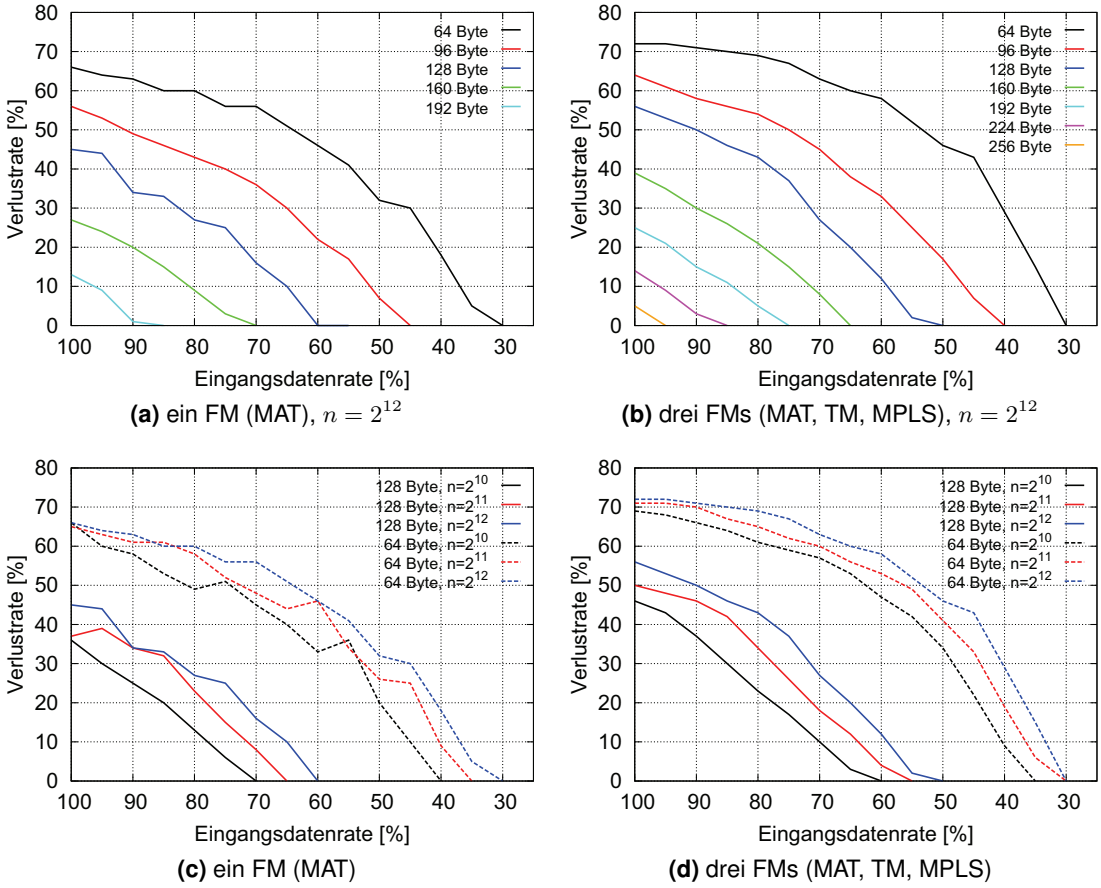
4.3.2. Systemevaluation

Die Charakterisierung des MATMUNI-Systems bzgl. seiner Leistungsdaten ist notwendig, um die Eignung für das angedachte Einsatzgebiet im TZN zu prüfen. Zudem sind einige Ergebnisse dieser Analysen im weiteren Verlauf der Arbeit noch von Bedeutung. Die Bewertung erfolgt nach standardisierten Richtlinien und Vorgehensweisen für den Leistungsvergleich von Netzwerkkomponenten und paketverarbeitenden Systemen [RFC1242, RFC2544, RFC2889]. Die Leistungsparameter Verlustrate, Durchsatz und Latenz sind von besonderem Interesse.

Für die einzelnen Simulationsläufe wurden die Eingangsdatenrate und die Größe der Test-Frames variiert, um verschiedene Verkehrsmuster zu erzeugen. Die Eingangsdatenrate ist in Prozent der maximalen Datenrate des Übertragungsmediums definiert. Im Fall von MATMUNI sind dies 8 unidirektionale Gigabit-Ethernet-Kanäle. Weitere Details zu Ethernet bieten die genannten RFCs sowie Abschnitt 2.4. Da die RFCs nur rein synthetische Verkehrsmuster definieren, werden zusätzliche Simulationen mit einer *realeren* Größenverteilung der Frames durchgeführt, welche an [SPH05] angelehnt ist. Sinha et al. analysieren den Datenverkehr an zentralen Knotenpunkten und leiteten daraus eine typische Verteilung der IP-Paketgrößen im Internet ab. Das Systemverhalten wurde zudem für verschiedene Speichertiefen (n) untersucht, da Anzahl und Dauer von Speicherzugriffen für die Suche nach Schlüsseln und Regeln wesentlichen Einfluss auf den Klassifizierungsprozess besitzen. Die typische Speichertiefe des MATMUNI-Systems liegt bei $n = 2^{12}$ Schlüsseln, um dem Einsatz auf einem DSLAM (siehe Abbildung 4.11) zu genügen. Bei bis zu 16 Linecards mit je 72 Ports können so pro Port mehrere Schlüssel konfiguriert werden. Während der Simulationen dienten die MAC-Adressen eines Frames als Schlüssel. Die Arbeitsfrequenz für das MATMUNI-System wird auf $f = 125$ MHz ($T_{clk} = 8$ ns) festgelegt, da dies die *Mindestanforderung* ist, um bei GbE pro Takt ein Byte verarbeiten zu können.

Verlustrate Die Verlustrate gibt den prozentualen Anteil verworfener Frames bzgl. aller zum Testsystem (Device-under-Test, DUT) gesendeten Frames an.

Drei Parameter haben prinzipiell Einfluss auf die Verlustrate: Eingangsdatenrate, Framegröße und Speichertiefe. Die Abbildungen 4.13a und 4.13c zeigen den Verlauf der Verlustrate des



(a) ein FM (MAT), $n = 2^{12}$

(b) drei FMs (MAT, TM, MPLS), $n = 2^{12}$

(c) ein FM (MAT)

(d) drei FMs (MAT, TM, MPLS)

Abbildung 4.13.: Verlustrate über der Eingangsdatenrate für verschiedene Framegrößen sowie Speichertiefen (siehe jeweils Legenden) und Funktionsmodule (FMs)

MATMUNI-Systems, wenn nur die MAT-Funktionalität eingebunden ist; Abbildungen 4.13b und 4.13d wenn alle drei o. g. FMs konfiguriert sind. Der ungünstigste Fall ist eine konstante Belastung des Systems mit Frames minimaler Größe (64 Byte) bei 100 % Eingangsdatenrate. Deshalb erscheint gerade bei Datenraten nahe 100 % und kleinen Framegrößen die Verlustrate mit bis zu 72 % relativ hoch. In kürzester Zeit ist die Verarbeitung sehr vieler Frames erforderlich. Am Speicher-Arbiter liegen fortwährend konkurrierende Suchanfragen an. In der Zwischenzeit füllen sich die Puffer in den Framebuffern, so dass Frames an den Eingängen des Systems verworfen werden. Erst bei Eingangsdatenraten um 30 % sinkt bei minimal großen Frames die Verlustrate auf null. Ab einer Framegröße von ca. 200 Byte in Abbildung 4.13a und 256 Byte in Abbildung 4.13b werden selbst bei 100 % Eingangsdatenrate keine Frames mehr verworfen.

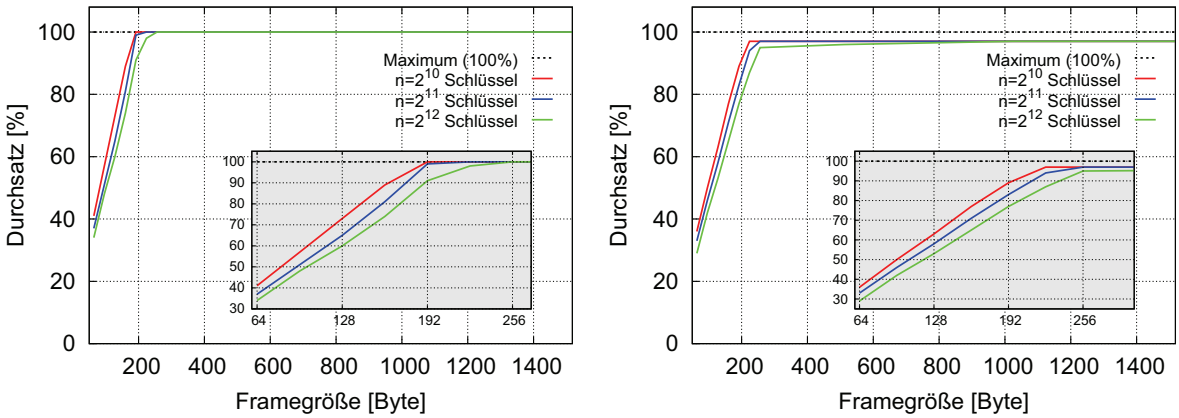
Steigt die Framegröße oder sinkt die Eingangsdatenrate, sinkt auch die Verlustrate, da zum einen pro Frame mehr Zeit für die Suche im Speicher zur Verfügung steht und zum anderen die Pufferspeicher sich nicht so schnell füllen. Die Diagramme in 4.13c und 4.13d zeigen den Einfluss der Speichertiefe auf die Verlustrate. Je geringer n ist, desto geringer ist die Verlustrate. Die Schlüssel werden sortiert verwaltet, wodurch die Suche nach Schlüsseln maximal $\log_2(n)$ Schritte benötigt und eine Komplexität von $\mathcal{O}(\log n)$ besitzt. Löschen und Einfügen haben zwar eine Komplexität von $\mathcal{O}(n)$. Der Normalfall ist allerdings die Suche.

Die Unterschiede zwischen den MATMUNI-Konfigurationen mit einem FM bzw. drei FMs stammen zum einen von den zusätzlichen Verzögerungen, welche die MAT-, TM- und MPLS-Module im Datenpfad erzeugen. Zum anderen sind sie durch die TM-Funktionalität begründet. Während MAT und MPLS pro Frame nur einen gemeinsamen Suchvorgang im Speicher erfordern, benötigt TM zwei. Die erste Suche liefert aktuelle Zählerstände bzgl. des genutzten Verkehrsvolumens für einen bestimmten Schlüssel. Der zweite Zugriff ist nötig, um diese Zählerstände um die Länge des aktuellen Frames zu erhöhen. Zudem erfolgt jeweils alle 100 ms ein periodisches Auffüllen der Zählerstände eines Schlüssels. Letztendlich sind somit zusätzliche Zugriffe auf den Speicher nötig, wodurch sich die Verlustrate verschlechtert.

Die Diagramme in Abb. 4.13 zeigen jedoch nur die Verlustraten für synthetische Verkehrsmuster, welche real nicht vorkommen. Bei Simulationen mit realer Framegrößenverteilung nach [SPH05] wurden in keinem Fall Frames aufgrund einer zu geringen Systemleistung verworfen. Eine Ausnahme stellt die Nutzung der MPLS-Funktionalität dar, welche durch die Kapselung nach Martini [RFC2697, RFC2698] *jeden* Frame um 22 Byte verlängert. Somit kommt es bei symmetrischen Ein- und Ausgangsbandbreiten unabhängig von Framegrößenverteilung und Pufferdimensionierung bei Eingangsdatenraten nahe 100 % *immer* zum Verwurf einiger Frames.

Durchsatz Der Durchsatz beschreibt die maximale Daten- bzw. Framerate, bei dem das DUT keinen Testframe mehr verwirft. Die Abbildungen 4.14a und 4.14b zeigen den Durchsatz des MATMUNI-Systems. Ähnlich wie bei der Verlustrate zeigt sich eine Abhängigkeit von der Speichertiefe und der Größe der Frames. Ist nur die MAT-Funktionalität in MATMUNI instanziiert, wird ab einer Framegröße von ca. 224 Byte der maximale Durchsatz von 100 % erzielt, wie der vergrößerte Ausschnitt in Abbildung 4.14a zeigt. Mit drei FMs erreicht der Durchsatz in keinem Fall 100 %, sondern liegt bei maximal 97 %. Der Grund ist wiederum die MPLS-Funktionalität, welche jeden Frame um 22 Byte erweitert, wodurch es zum Verwurf einzelner Frames kommt.

Latenz Für paketverarbeitende Systeme in der Telekommunikation beschreibt die Latenz das Zeitintervall t zwischen der Übernahme des letzten Bytes eines Frames am Eingang und dem Erscheinen des ersten Bytes am Ausgang [RFC1242, RFC2544]. Abbildung 4.15 zeigt die



(a) Durchsatz bei einem FM (MAT)

(b) Durchsatz bei drei FMs (MAT, TM, MPLS)

Abbildung 4.14.: Durchsatz über Framegröße für $n = 2^{10}, 2^{11}, 2^{12}$ Schlüssel

durchschnittliche Latenz t_{avrg} über der Eingangsdatenrate. Die Regressionsgerade nähert diese Abhängigkeit an. Simuliert wurde das MATMUNI-System mit drei FMs und einer realen Verteilung der Framegrößen. Die Streuung der Simulationswerte ergibt sich aus der pro Simulationsdurchlauf unterschiedlichen zufälligen Verteilung der Framegrößen und Schlüssel. Unter Maximallast beträgt die Latenz ca. 1170 Taktzyklen, was bei $f = 125$ MHz ca. $t = 9{,}36 \cdot 10^{-6}$ s entspricht. Unter Betrachtung der Struktur des Internets bzw. der TZN im Ganzen kann diese zusätzliche Verzögerung vernachlässigt werden. Mit sinkender Eingangsdatenrate sinkt t, da Frames weniger lange auf ihre Verarbeitung warten müssen. Simulationen mit einzelnen Frames, d. h. im lastfreien Betrieb, haben ergeben, dass der minimale Wert t_0 für diese Systemkonfiguration nur zwischen 73 und 79 Taktzyklen liegt. t_0 ist alleine abhängig vom Schlüssel, d. h. von der Dauer der Suche im Speicher. Die Framegröße hat keinen Einfluss auf t_0. Gleichung (4.1) zeigt die einzelnen Komponenten von t. n_{buf} ist Anzahl der Takte, die ein Frame im Puffer wartet, bis er verarbeitet wird, d. h., bis der entsprechenden Speicheranfrage Zugriff auf den Speicher gewährt wird. Im Speicher werden n_{mem} Takte für die Suche benötigt. $n_{signaling}$ ist die Anzahl der Takte der system-internen Signalisierung durch synchrone Steuersignale. n_{func} ist die Summe der Verzögerungen durch die FMs. $n_{singaling}$ und n_{func} sind fix. n_{buf} und n_{mem} sind variabel. Unter Last ist n_{buf} ausschlaggebend für t und damit für die Leistung des Systems.

$$t = T_{clk} \cdot (n_{buf} + n_{mem} + n_{func} + n_{signaling}) \tag{4.1}$$

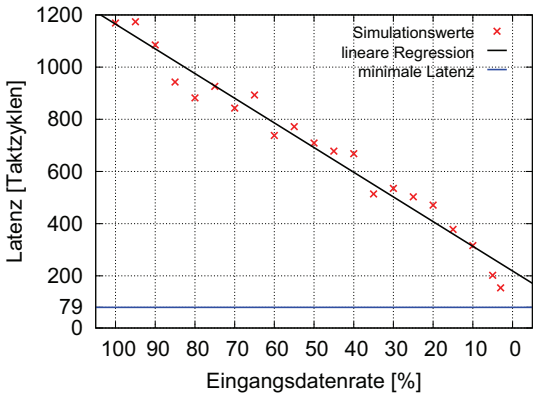


Abbildung 4.15.: Latenz über Eingangsdatenrate bei realer Framegrößenverteilung

Synthese Tabelle 4.1 fasst die Syntheseergebnisse des MATMUNI-Systems zusammen. Zielplattform ist ein Xilinx Virtex-4 FX100 FPGA (XC4VFX100-10). Genutzt wurde das Integrated Software Environment 9.1 von Xilinx. Synthese sowie Platzierung und Verdrahtung (Place & Route) erfolgten unter Standardeinstellungen. Die einzelnen Komponenten und das Gesamtsystem wurden für eine Systemkonfiguration mit 1, 2 und 4 bidirektionalen GbE-Kanälen synthetisiert. Während Größe und Frequenz der Framebuffer, Control-Module und FMs unabhängig von der Anzahl der GbE-Kanäle sind, steigt die Komplexität der Arbitrierungslogik des Speicher- und CPU-Arbiters mit steigender Zahl der Datenpfade, d. h. mit steigender Zahl konkurrierender Speicheranfragen. Gleichzeitig sinkt die maximale Arbeitsfrequenz f_{max} dieser zentralen Module. Die Framebuffer und die zentralen Arbiter erfordern die meisten Logikressourcen. Die eigentlichen FMs hingegen sind vergleichsweise kompakt. Der erzielbare Arbeitstakt nach der Platzierung und Verdrahtung ist um bis zu 14 MHz geringer, da nun die tatsächlichen Leitungsverzögerungen bekannt sind und in die Parametrisierung des Systems einfließen.

Diskussion Die Systemevaluation hat gezeigt, dass das MATMUNI-System unter *synthetischen* Verkehrsmustern den Randbedingungen für einen Einsatz nach Abbildung 4.11 aufgrund hoher Verlustraten nicht genügt. Unter *realen* Lastbedingungen arbeitet das System jedoch verlustfrei. Im Folgenden sind markante Eigenschaften der Systems auf Basis der in Abbildung 4.12 dargestellten synchronen Datenpfadarchitektur zusammengefasst:

Zentrale geteilte Ressourcen Aus Sicht der simulativen Leistungsbewertung kann der zentrale Speicher als der bestimmende limitierende Faktor des Gesamtsystems ermittelt werden. Steigt die Anzahl der Speicheranfragen pro Zeiteinheit auf den gemeinsamen Speicher, ist die Maximalleistung der Suchlogik schnell erreicht, z. B. durch hohe Eingangsdatenraten,

Tabelle 4.1.: Syntheseergebnisse für MATMUNI auf Basis einer synchronen Systemarchitektur mit 1/2/4 GbE-Kanälen

Modul	Slices	f_{max} [MHz]
Sync-FIFO	187	245
Upstream Keyparser & Framebuffer	672	131
Downstream Keyparser & Framebuffer	722	131
Speicher-Arbiter	580/1085/2186	182/134/126
CPU-Arbiter	365/346/414	265/264/163
Control-Module	67	203
MAT Upstream	116	215
MAT Downstream	115	213
TM	226	190
MPLS Upstream	177	232
MPLS Downstream	105	280
Gesamtsystem nach Synthese	2941/5084/9713	129/131/124
Gesamtsystem nach Place & Route	3151/5463/10236	126/129/110

kleine Framegrößen oder die Anzahl konkurrierender Speicheranfragen. Der zentrale Speicher ist ein gemeinsam genutztes Medium, welches nur eine bestimmte Menge an Suchanfragen pro Zeiteinheit bearbeiten kann. Dies wird in übertragenem Sinn auch als „von Neumann-Flaschenhals“ bezeichnet. Zudem steigt mit der Anzahl der Datenpfade die Komplexität des Speicher-Arbiters an. Gleichzeitig sinkt jedoch die maximale Frequenz seines Arbeitstakts.

Angepasstheit & Flexibilität Das MATMUNI-System ist für einen konkreten Anwendungsfall zugeschnitten. Die Konfiguration mit mehr als den vier angedachten bidirektionalen GbE-Datenpfaden und weiteren FMs ist theoretisch möglich. Einerseits erfordert dies jedoch tiefgreifende Modifikationen an den Schnittstellen und dedizierten Kommunikationskanälen sowie den Zustandsmaschinen der beteiligten Systemkomponenten. Andererseits bricht die Leistung aufgrund des zentralen Speichers ein. Die serielle Anordnung der FMs in den Datenpfaden und die synchrone Systemansteuerung führen zudem dazu, dass der Framebuffer eines Datenpfades und der Speicher-Arbiter solange auf das Auslesen eines neuen Frames warten müssen, bis das letzte FM die Verarbeitung des aktuellen Frames signalisiert hat. Diese Kopplung der eigentlich autarken FMs und Systemmodule führt zu unnötigen Wartezyklen in den Datenpfaden und senkt so den maximalen Durchsatz.

Maximaler Arbeitstakt Die Abhängigkeit der maximalen Arbeitsfrequenz f_{max} des Gesamtsystems von den langsamen Teilmodulen ist ein inhärenter Schwachpunkt der synchronen

Systemarchitektur (in Tabelle 4.1 hervorgehoben). Während einzelne FMs alleine z. T. weit über 200 MHz erreichen können, beträgt f_{max} im Fall von 4 parallelen bidirektionalen GbE-Kanälen insgesamt nur 110 MHz[2]. Dies liegt noch unter den Mindestanforderungen von 125 MHz für den GbE-Betrieb. Sowohl der komplexe Speicher-Arbiter als auch die Framebuffer des Up- und Downstreams limitieren somit den maximalen Systemtakt und damit die Leistung des Gesamtsystems.

Mit dem MATMUNI-System als Beispielanwendung werden in Teil III dieser Arbeit Ansätze zur Optimierung der Systemarchitektur diskutiert. Durch Nutzung einer Network-on-Chip-Architektur werden insbesondere die genannten Problemstellungen bzgl. der Flexibilität und ungenügenden Skalierbarkeit der synchronen Architekturvariante adressiert.

4.4. Zusammenfassung des Kapitels

Ethernet bietet mit seinen verschiedenen Ausprägungen auf der Sicherungsschicht eine einheitliche Basis für den konvergierenden Telekommunikationsmarkt. Kapitel 4 hat mit MAT & sMAT Mechanismen zum Einsatz in aktuellen und zukünftigen Ethernet-basierten TZN vorgestellt. MAT befasst sich dabei mit Anforderungen an Sicherheit und Skalierbarkeit in der Telekommunikation, die in Kapitel 3 aus den aktuellen Trends und Problemen in diesem Bereich abgeleitet wurden. Durch die direkte Verhinderung grundlegender Bedrohungsszenarien auf der Sicherungsschicht können darauf aufbauende Bedrohungen auf höheren Schichten ebenfalls unterdrückt werden. Zum anderen bewirken die flexibel konfigurierbaren Ansätze MAT & sMAT eine Reduktion der steigenden Anzahl an MAC-Adressen in TZN und im Kernbereich des Internets. Das unter Abschnitt 4.3 vorgestellte MATMUNI-System stellt eine konkrete Realisierung des MAT-Konzeptes als voll funktionsfähigen Prototyp dar. Zudem ist das MATMUNI-System auf Basis der in Abschnitt 4.3.2 evaluierten synchronen Datenpfadarchitektur Ausgangspunkt für weitere Betrachtungen in Teil III dieser Arbeit.

[2] Andere FPGAs bzw. neuere FPGA-Generationen können bessere Syntheseergebnisse liefern.

Interessante Sache... Aber wozu ist sie gut?
(ein IBM Manager zum Mikroprozessor, 1968)

Kapitel 5.

Vertrauenswürdigkeit im Internet – IP Calling Line Identification Presentation

Kapitelstruktur

Vertrauenswürdigkeit spielt in einem derart offenen und komplexen Netz wie dem Internet eine wesentliche Rolle, die noch erheblich an Bedeutung zunehmen wird. In diesem Kapitel liegt der Fokus auf dem in Abschnitt 5.1 vorgestellten Mechanismus *IP Calling Line Identification Presentation (IPclip)*, welcher primär der Bereitstellung einer glaubwürdigen Ortsreferenz auf IP-Ebene und dadurch der Schaffung einer Vertrauensbasis im Internet dient. Darüber hinaus werden in Abschnitt 5.2 hochaktuelle Szenarien diskutiert, welche das breite Spektrum der Anwendungsmöglichkeiten von IPclip belegen.

5.1. Der IPclip-Mechanismus

Motivation Moderne Nutzungsformen IP-basierter Kommunikation wie VoIP, Homebanking oder E-Mail sind bereits als fester Bestandteil des Alltags anzusehen. Dessen ungeachtet kann gerade beim durchschnittlichen Privatnutzer nicht vorausgesetzt werden, dass ein technologisches Verständnis für alle Nuancen und damit auch Risiken des Mediums Internet vorhanden ist. Der Durchschnittsanwender möchte moderne Internetdienste benutzen bzw. konsumieren, erwartet dabei aber einen mit klassischen Diensten zumindest vergleichbaren Sicherheitsstandard. Mehr noch – die Existenz eines vergleichbaren Grades an Sicherheit wird meist stillschweigend angenommen und vorausgesetzt. Darüber hinaus können die neuen Technologien in vielen Bereichen nicht die gleichen Dienste und Qualitäten liefern, wie sie von traditionellen Architekturen, wie etwa dem leitungsvermittelten PSTN, gewohnt sind. Das rasante Wachstum des Internets macht das Thema Sicherheit zu einer Problematik, welche die breite Masse betrifft. Sicherheitsaspekte spielen deshalb eine bedeutsame Rolle, was u. a. die folgenden Gründe hat:

- Durch die Entwicklung des Internets von einem Wissenschaftsnetz in ein globales Kommunikationsmedium haben sich die Anforderungen an die Netzwerkinfrastruktur und Dienste radikal verändert (siehe Kapitel 3).

- Veraltete Protokolle, zu denen z. B. auch IP(v4) zählt, wurden ursprünglich nicht für eine derart große Zahl an Nutzern und verschiedenen Nutzergruppen entworfen und zeigen Schwachpunkte und Mängel, die in diesem Zusammenhang nicht vorhersehbar waren. Es ist erforderlich, diese Schlupflöcher zu schließen.

- Die Vergabe dynamischer IP-Adressen und die Zunahme der Mobilität im Internet (Laptops, VoIP-Handys) sind Gründe dafür, dass ein Nutzer nicht eindeutig zu einer bestimmten IP-Adresse bzw. einem Internet-Anschluss zuordenbar ist.

- Durch das Wachstum des Internets steigt dessen Komplexität und damit die Anonymität des Einzelnen. Zu Beginn existierte im Internet eine überschaubare Zahl seriöser Teilnehmer. Heutzutage schöpfen „schwarze Schafe“ alle Möglichkeiten aus, um ihre Identität zu verschleiern und das Internet für ihre Zwecke zu missbrauchen.

Die Schaffung einer gesicherten Vertrauensbasis im Internet ist wegen der steigenden Anonymität unabdingbar. Dazu wird zunächst der Begriff *Trust-by-Wire* näher erläutert.

Trust-by-Wire (TBW) beschreibt i. Allg. die implizite Sicherheit in herkömmlichen, leitungsvermittelten Kommunikationsnetzen wie dem PSTN, in welchen das Leitungsnetz und die

Leitungsvermittlung voneinander getrennt sind. Unter TBW ist eine direkte, eindeutige Beziehung zu verstehen zwischen einer Art von Nutzer-Identifikation, z. B. der Rufnummer, und dem physikalischen Kommunikationsmedium, z. B. der TAL. Mit anderen Worten – TBW steht für Eindeutigkeit und Vertrauenswürdigkeit in Telekommunikationsnetzen. Im PSTN z. B. dienen Telefonnummern der Ende-zu-Ende-Kommunikation. Eine Rufnummer gibt direkten Rückschluss auf die TAL und damit auf einen wohldefinierten Ursprung und ist somit eine vertrauenswürdige Information für die Kommunikationsteilnehmer. Darüber hinaus wird in dieser Arbeit unter TBW die zurzeit vorhandene Diskrepanz zwischen leitungsvermittelten und paketvermittelten Netzen verstanden, da Vertrauensverhältnisse in modernen, paketvermittelten IP-Datennetzen *nicht* auf dem TBW-Prinzip basieren. Im Internet wird Ende-zu-Ende-Konnektivität per IP realisiert. Ein direkter Zusammenhang zwischen IP-Adressen und physikalischen Leitungen existiert allerdings nicht. IP-Adressen sind somit kein Äquivalent zu klassischen Rufnummern. Zudem wurden sie ursprünglich nicht für den Zweck entworfen, in irgendeiner Form Ortsinformationen zur Verfügung zu stellen, und sind daher unter diesem Gesichtspunkt nicht sicher. Durch Proxys und Mechanismen wie NAT kann zudem ein Tausch der IP-Adressen auf der Übertragungsstrecke erfolgen, wodurch im Netzwerk und beim Empfänger keine glaubwürdigen Informationen über den tatsächlichen Ursprung der IP-Pakete existieren. Weiterhin sind Absender-IP-Adressen fälschbar (z. B. IP-Spoofing) und können daher nicht in jedem Fall zur Ermittlung der Quelle herangezogen werden. Ein Beispielszenario mit einer gewissen Brisanz, welches in Abschnitt 5.2.1 noch genauer erläutert wird, sind VoIP-Notrufe in mobilen Umgebungen. Gerade hier ist eine vertrauenswürdige und vor allem eindeutige Ortsreferenz entscheidend, um dem Anrufer schnellstmöglich und zielgerichtet Hilfe zukommen zu lassen.

Da im Internet Vertrauensverhältnisse auf Basis von TBW nicht realisierbar sind, werden bisher hauptsächlich Ansätze auf Basis von *Trust-by-Authentication* (TBA) genutzt. TBA beschreibt den Aufbau eines Vertrauensverhältnisses durch die Verifizierung der Identität eines Teilnehmers bzw. Dienstnutzers im Internet (nicht der TAL oder des CPEs) anhand einer sicheren Information, z. B. durch das Wissen eines Passwortes oder den Besitz eines Schlüssels. Eine Vielzahl anwendungsspezifischer und proprietärer Insellösungen existiert, was jeder Internetnutzer an der Größe seiner privaten Account- & Passwort-Liste erkennen kann. Das Ziel sind jedoch globale, allgemeine TBA-Ansätze [Wim07]. In diesem Zusammenhang existiert z. B. das mittlerweile seit einigen Jahren bestehende ENUM-Projekt (Telefon Number Mapping) zur global einheitlichen Assoziation von Ressourcen aus dem Telekommunikations- und dem Internetbereich [RFC3761, DEN]. Es hat den Zweck, jeden Teilnehmer und insbesondere seine Dienste ausschließlich über ein einziges Merkmal, der Telefonnummer, kontaktieren, authentifizieren und identifizieren zu können. Jedoch hat ENUM aufgrund ökonomischer Ursachen bisher nur unzureichende Akzeptanz gefunden.

Zur Garantie eines Vertrauensverhältnisses auf Ebene der Vermittlungsschicht in paketvermittelten IP-Datennetzen wurde ein neuartiger Mechanismus mit der Bezeichnung *IPclip* (Internet Protocol-Calling Line Identification Presentation) entwickelt [KWD+08b, DKW+08c, DKW+08a]. Dieser Mechanismus wird im Folgenden vorgestellt.

5.1.1. Allgemeines Prinzip

Der Name IPclip ist von der CLIP-Funktion (Calling Line Identification Presentation) konventioneller ISDN-Telefonnetze abgeleitet. Ursprünglich stellt CLIP ein optionales Feature für ISDN dar, bei dem die Nummer des Anrufers dem Angerufenen übermittelt und z. B. in Klartext angezeigt wird, wodurch der Angerufene den Anrufer identifizieren kann. Im Fall von IP-Datennetzen kann die IP-Adresse eines Nutzers jedoch nicht als Äquivalent zu einer Festnetztelefonnummer angesehen werden, wie eingangsseitig beschrieben wurde. Deshalb werden der eigentliche Grundgedanke und der Name des CLIP-Features klassischer ISDN-Netze mittels des neuen IPclip-Mechanismus in paketvermittelten IP-Datennetzen adaptiert. Dabei ist IPclip aus technischer Sicht ein vollkommen neuartiger Ansatz und mit ISDN-CLIP nicht vergleichbar.

Das Ziel von IPclip ist die Bereitstellung *eindeutiger* und *glaubwürdiger* Ortsreferenzen auf Ebene der Vermittlungsschicht (IP). Ein Rückschluss auf die Herkunft von IP-Datenverkehr soll ermöglicht werden, um ein ähnliches Level an Vertrauenswürdigkeit (TBW) wie z. B. im PSTN zu erreichen. IPclip-Informationen authentifizieren jedoch keinen Nutzer, sondern geben ausschließlich Aufschluss über die jeweilige geographische Position des CPE. TBA-Mechanismen werden dabei durch IPclip nicht ersetzt, sondern komplementiert und unterstützt. Um diese zusätzlichen Ortsreferenzen in globalem Umfang bereitstellen zu können, wird in jedem IP-Paket ein zusätzliches Datenfeld (IP-Option) eingefügt. Der Grund dafür ist, dass IP das zentrale Protokoll im Internet ist und in der Regel Ende-zu-Ende-Konnektivität zwischen Kommunikationsendpunkten bereitstellt. Mittels IPclip werden ein Teilnehmer und seine aktuelle geographische Position anhand eines Tupels identifiziert, welches aus seiner momentanen IP-Adresse plus Zusatzinformationen besteht. Da IP-Adressen diese Beziehung nicht alleine eindeutig herstellen können, muss die zuverlässige Ortsreferenz in den Zusatzinformationen enthalten sein. Dazu werden vorzugsweise standardisierte Datenformate wie GPS (Global Positioning System) genutzt, um globale Interoperabilität, welche im Internet unerlässlich ist, zu gewährleisten. Die Summe aller Zusatzinformationen – nachfolgend insgesamt mit LI (Location Information) bezeichnet – kann zu Analysezwecken, zur Realisierung neuer Dienste oder zur Stimulierung weiterführender Aktionen genutzt werden. Neben der Funktionalität, IPclip-Optionen im Upstream in IP-Pakete einzufügen, kann der IPclip-Mechanismus diese im Downstream auch wieder entfernen. Dies kann aus Datenschutzgründen notwendig sein oder wenn ein Anwendungsszenario dies nicht

vorsieht. Der Einsatz von IPclip ist in diesem Fall völlig transparent für den Endbenutzer.

Der TBW-Ansatz via IPclip folgt dabei den Empfehlungen aus [PZWF07], ISPs und Netzbetreiber für den Datenverkehr verantwortlich zu machen, der aus ihren jeweiligen Verwaltungsdomänen stammt, um die allgemeine Sicherheit und Vertrauenswürdigkeit im Internet zu erhöhen. ISPs sollten sich nicht ausschließlich nur auf die Untersuchung und Kontrolle von eingehendem Datenverkehr fokussieren (Inbound Traffic Control), um die eigenen Kunden zu schützen. Stattdessen sollte der Fokus auch auf der Vorabkontrolle von ausgehendem Datenverkehr (Outbound Traffic Control) liegen. IPclip *ist* eine Form von Outbound Traffic Control.

Die Addition von Ortsinformationen durch IPclip einschließlich ihrer Verifikation und Analyse wirft dabei die folgenden, wesentlichen Fragen auf, welche in den Abschnitten 5.1.2 bis 5.1.6 diskutiert werden:

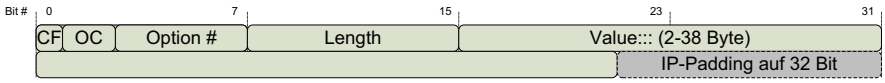
- Welche Informationen sind als Ortsinformationen nutzbar? Wie sind diese strukturiert?
- Welches ist die Position in der Netzinfrastruktur, an der derartige Ortsinformationen verfügbar sind und in IP-Pakete eingefügt werden können?
- Wie kann ein Vertrauensverhältnis bzw. ein gewisser Grad an Zuverlässigkeit bei der Analyse und Validierung der Ortsinformationen gesichert und dargestellt werden?
- Welche Einflüsse hat das Hinzufügen von Informationen auf IP und den Datenfluss?
- Gibt es Randbedingungen, die beachtet werden müssen?

5.1.2. IPclip-Optionen

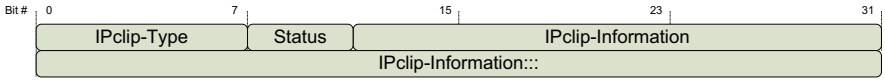
IPclip-Optionen werden als Wert des Value-Feldes einer IPv4-Option (siehe auch Abschnitt 2.5) in den IPv4-Header eingefügt . Struktur und Größe von IPv4-Optionen sind in [RFC0791] spezifiziert und in Abbildung 5.1a dargestellt. Somit ist IPclip ein standard-konformer Weg zur Übertragung zusätzlicher (Orts)Informationen. Jedes IP-fähige Netzwerkgerät kann dadurch IPclip-Optionen entweder verarbeiten oder, wenn es nicht IPclip-fähig ist, zur Wahrung der Interoperabilität ignorieren und überspringen. Die Entscheidung zur Nutzung von IP-Optionen für IPclip ist darüber hinaus durch folgenden Auszug aus RFC 791 motiviert: „The options provide for control functions [...] options include provisions for timestamps, security, and special routing."

Das Typ-Feld einer IPv4-Option unterteilt sich in 1 Bit für das Copied-Flag (CF), 2 Bit für Option Class (OC) und 5 Bit für die IP-Optionsnummer. Obwohl der IPclip-Mechanismus zwar Funktionen zur Vermeidung der Fragmentierung von IP-Paketen bietet (siehe Abschnitt 5.1.5), sollte CF jedoch auf 1 gesetzt sein, um eine IPclip-Option zu allen Fragmenten eines IP-Paketes

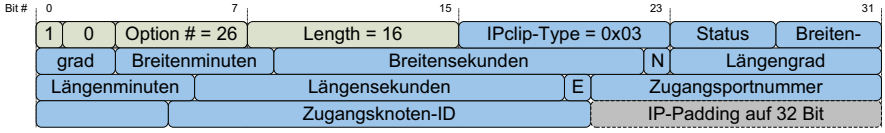
zu kopieren, falls eine Fragmentierung dennoch nötig ist. OC hat den Wert 0 (= control), da für IPclip eine reguläre Implementierung vorgesehen ist. Als IP-Optionsnummer wird vorerst 26 genutzt, da diese laut [Int] verfügbar ist.



(a) Allgemeine TLV-Struktur einer IPv4-Option



(b) Allgemeine Struktur einer IPclip-Option als Wert (Value) einer IPv4-Option



(c) Struktur einer IP-Option mit kompletter IPclip-Option des Typs `0x03` (blau) inkl. Zugangsportnummer und Zugangsknoten-ID

Abbildung 5.1.: Format von IPclip-Optionen

Die IPclip-Option selbst setzt sich aus 8 Bit für den IPclip-Typ, einem 4-Bit-Statusfeld für verschiedene Flags und den eigentlichen Optionsinformationen (LI) variabler Länge zusammen. Diese Struktur veranschaulicht Abbildung 5.1b. Das Statusfeld stellt 4 Flags für verschiedene Zwecke bereit:

- Das Source-Flag (SF) gibt die Quelle der LI an (0 = vom Nutzer eingefügt, 1 = durch IPclip im Zugangsknoten eingefügt bzw. ersetzt).
- Das Trustability-Flag (TF) gibt Rückschluss über die Glaubwürdigkeit der LI (0 = inkorrekt bzw. nicht glaubwürdig, 1 = korrekt bzw. glaubwürdig).
- Das Peering-Flag (PF) wird benötigt, wenn Datenverkehr über die Domain-Grenzen eines Netzbetreibers hinaus weitergeleitet wird (0 = das IP-Paket entstammt der eigenen Verwaltungsdomäne, 1 = das IP-Paket stammt aus einer fremden Verwaltungsdomäne)[1].
- Das Removal-Flag (RF) dient Datenschutzzwecken und der Privatsphäre. Dürfen Ortsinformationen nicht direkt bis zum Empfänger weitergeleitet werden, kann die LI auf dem empfängerseitigen Zugangsknoten durch die dort vorhandene IPclip-Funktionalität wieder entfernt werden (0 = LI kann bis zum Ziel weitergeleitet werden, 1 = LI darf nicht

[1] Weitere Details zum Umgang mit PF bietet der Anwendungsfall in Abschnitt 5.2.2.

bis zum Ziel weitergeleitet werden). RF-Flags sind abhängig von den jeweiligen regionalen Richtlinien und Anwendungsfällen und werden durch die IPclip-Funktionalität auf dem senderseitigen Zugangsknoten gesetzt.

Die IPclip-Option ermöglicht die Übertragung verschiedener Informationstypen. Tabelle 5.1 enthält bereits definierte IPclip-Typen und die Gesamtgröße L_{IP-Opt} der entsprechenden IP-Option. Die Werte `0x00` und `0xFF` sind reserviert. Aufgrund ihrer globalen Verfügbarkeit werden GPS-Daten zur Darstellung geographischer Ortsinformationen bevorzugt. Zur Übertragung von GPS-Informationen wird das NMEA-0183 Datenformat [Nat02] auf Basis von WGS84-Koordinaten (World Geodetic System 1984) [NIM00] genutzt. Reine GPS-Daten sind z. B. durch den IPclip-Typ `0x01` gekennzeichnet. GPS-Daten in Kombination mit der Zugangsknoten-ID und der Zugangsportnummer der Linecard (= TAL-Identifikation) sind durch den IPclip-Typ `0x03` definiert. Die für den Inhalt dieser beiden IPclip-Typen maßgeblichen Ortsinformationen bestehen aus der Angabe von Längen- und Breitengrad. Die Kodierung der GPS-Koordinaten ist Tabelle 5.2 zu entnehmen. IPclip-Typ `0x03` enthält zusätzlich vier Byte für die TAL-Identifikation. Abbildung 5.1c zeigt die komplette IP-Option für diesen IPclip-Typ. Aufgrund des notwendigen Paddings einer IP-Option auf ein Vielfaches von 32 Bit beträgt L_{IP-Opt} in diesem Fall 16 Byte. Details zu weiteren IPclip-Typen sind in Abschnitt B.1 im Anhang enthalten.

Tabelle 5.1.: IPclip-Optionstypen

IPclip-Typ	Beschreibung	L_{IP-Opt}
`0x00` & `0xFF`	reserviert	n/a
`0x01`	GPS-Daten	12 Byte
`0x02`	GLI-Daten	20 Byte
`0x03`	GPS-Daten + TAL-Identifikation	16 Byte
`0x04`	GLI-Daten + TAL-Identifikation	24 Byte
`0x05`	GPS-Zeitinformationen	8 Byte
`0x06...0x030`	derzeit noch ungenutzt	n/a

5.1.3. IPclip's Position im Teilnehmerzugangsnetzwerk

Die IPclip-Funktionalität ist im TZN auf den ersten Linecards aus Sicht der Endkunden angesiedelt, wie Abbildung 5.2 veranschaulicht. Diese Positionierung unmittelbar am Eintrittspunkt der Teilnehmer in die TZN hat dabei verschiedene Gründe:

- Um eine geeignete Genauigkeit der Ortsreferenz zu gewährleisten, welche durch IPclip selbst eingefügt wird, muss die Funktionalität so nahe wie möglich in der Nähe der Endkunden bzw. der Eintrittspunkte in das TZN angesiedelt sein. Die Zugangsportnummer einer

Tabelle 5.2.: Format der GPS-Information in einer IPclip-Option des Typs `0x01` bzw. `0x03`

Parameter		Wertebereich	# der Bits
Breite	Grad	0...90	7
	Minuten (Ganzzahl)	0...90	6
	Minuten (Nachkommateil)	0...(1-2^{-15})	14
	Hemisphäre	Nord/Süd	1
Länge	Grad	0...180	8
	Minuten (Ganzzahl)	0...59	6
	Minuten (Nachkommateil)	0...(1-2^{-15})	14
	Hemisphäre	Ost/West	1

Linecard wird bereits als inhärente Leitungsinformation und somit als eine besondere Form von Ortsreferenz angesehen, welche aber nur auf den Linecards vorhanden ist. Zudem bietet sich die Kombination aus geographischer Position und der TAL-Identifikation (ID des Zugangsknotens & Zugangsportnummer der Linecard) als präzise LI zur Identifizierung und Lokalisierung eines Nutzers an.

- Der IPclip-Mechanismus basiert auf der Annahme, dass LI entweder vom Kunden selbst (durch das CPE) *oder* vom Netzbetreiber (auf den Linecards) eingefügt wird. Der Vergleich und die Validierung der Ortsinformationen sowie daraufhin angewandte Maßnahmen werden jedoch *ausschließlich* auf den Linecards durchgeführt. Diese Regelung rührt daher, dass CPEs von Netzbetreibern als nicht vertrauenswürdige Netzwerkelemente angesehen werden, da sie i. Allg. außerhalb seines Verwaltungsbereiches liegen. Linecards hingegen sind Teil des Betreibernetzes und unterliegen seiner Verwaltung.

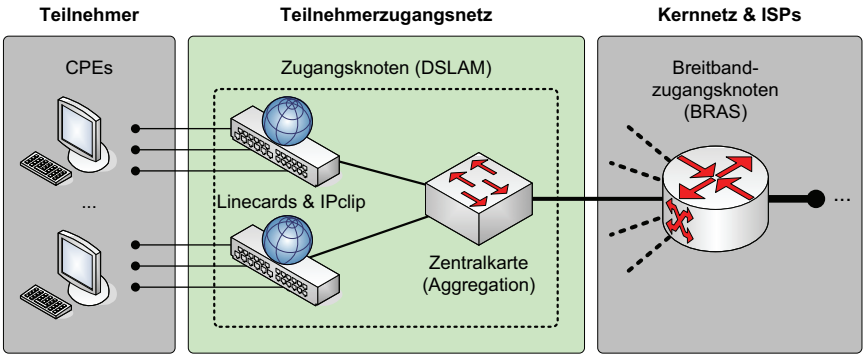


Abbildung 5.2.: Anordnung von IPclip auf den Linecards des DSLAMs im TZN

5.1.4. Validierung von Ortsinformationen

IPclip dient der Bereitstellung einer glaubwürdigen Ortsreferenz. Um die notwendige Vertrauensbasis gewährleisten zu können, muss die in einer IPclip-Option enthaltene LI verifiziert und auf Plausibilität hin überprüft werden. Der Grund dafür ist, dass die LI entweder durch das CPE des Nutzers oder auf dem Zugangsknoten durch IPclip selbst eingefügt werden kann. Einerseits können Netzwerkteilnehmer diese LI unbeabsichtigt bzw. unwissentlich falsch konfigurieren. Andererseits kann durch die Vortäuschung einer inkorrekten LI die wahre Ortsreferenz aber auch gezielt verschleiert werden. Deshalb sind CPEs aus Sicht von IPclip prinzipiell als nicht vertrauenswürdig angesehen und vom Nutzer bereitgestellte LI muss durch die IPclip-Instanz im Zugangsknoten überprüft werden.

IPclip kann inkorrekte LI erkennen. Dazu wird die Tatsache ausgenutzt, dass nur Kunden an einen DSLAM angeschlossen sein können, welche sich in einer geographisch sinnvollen Nähe relativ zu diesem DSLAM befinden. Diese Zone um einen DSLAM wird als *Subscriber Catchment Area* (SCA) definiert – dem Einzugsbereich des DSLAMs. Die SCA ist ein pro DSLAM konfigurierbarer Parameter, welcher bildhaft als Kantenlänge eines Quadrates bzw. Durchmesser eines Kreises verstanden werden kann, in dessen Zentrum sich der DSLAM befindet. Ein anschauliches Beispiel für die Größe der SCA ist die doppelte Länge der TAL von der TAE bis zum DSLAM, welche u. a. abhängig von der genutzten Übertragungstechnologie ist. Während des Validierungsprozesses wird die Plausibilität einer vom Nutzer bereitgestellten LI durch einen Vergleich mit der fixen Position des DSLAMs unter Berücksichtigung der SCA ermittelt. Eine LI ist falsch, wenn sie nicht innerhalb der SCA des jeweiligen DSLAMs liegt. Ergibt der Vergleich, dass eine falsche LI durch den Nutzer eingefügt wurde – z. B. aufgrund von Mobilität, Fehlkonfiguration oder absichtlicher Manipulation – wird diese durch die inhärente LI des DSLAMs ersetzt. Hat der Nutzer keine IPclip-Option zur Verfügung gestellt, wird eine neue, komplette IPclip-Option eingefügt. Durch diese Vorgehensweise ist jedes IP-Paket mit verifizierter LI angereichert, wenn es den Zugangsknoten in Richtung Kernnetz verlässt. Die LI hat dabei mindestens die Genauigkeit der SCA des Zugangsknotens plus der Zugangsportnummer der Linecard.

Das Ergebnis der Verifikation wird mittels der Flags SF und TF in der IPclip-Option gespeichert. Wie unter Abschnitt 5.1.2 bereits beschrieben, geben beide Bits primär Auskunft über die Herkunft und Glaubwürdigkeit der LI. Sie können zudem z. B. zur Verwaltung oder als Auslöser für verschiedene vom Anwendungsfall abhängige Maßnahmen verwendet werden. Durch diese Flags wird das Vertrauensverhältnis zu jeder Zeit gewahrt. Als zentraler Bestandteil des TBW-Konzeptes ist die Bezeichnung der Flags dabei an den international üblichen Fachjargon angepasst. Tabelle 5.3 zeigt die vier möglichen Kombinationen der beiden Flags sowie ihre

Tabelle 5.3.: Allgemeine Interpretation der Status-Flags SF & TF einer IPclip-Option

Bezeichnung	Interpretation	Wert [SF,TF]
`user provided/ untrusted`	Eine vom CPE eingefügte LI hat die Verifikation nicht überstanden.	`00`
`user provided/ trusted`	Eine vom CPE eingefügte LI hat die Verifikation überstanden.	`01`
`network provided/ untrusted`	Eine vom CPE eingefügte LI hat die Verifikation nicht überstanden und ist durch IPclip mit der LI des Zugangsknotens überschrieben worden.	`10`
`network provided/ trusted`	Der Nutzer stellte keine LI bereit. IPclip hat auf dem Zugangsknoten eine neue IPclip-Option eingefügt.	`11`

Bezeichnungen und Interpretationen.

Abbildung 5.3 veranschaulicht an einem Beispiel die Arbeitsweise des Validierungsprozesses. Zum besseren Verständnis sind die Koordinaten und die SCA auf den Bereich 0...1 normalisiert. Zwei Nutzer (Alice und Eve) befinden sich an den Positionen (0.2;0.7) und (0.3;0.2). Der DSLAM befindet an der zentralen Position (0.5;0.5) bezogen auf die SCA. Von Alice gesendete IP-Pakete enthalten IPclip-Optionen mit korrekter LI. Durch die Verifikation auf dem DSLAM werden SF und TF auf `user provided/trusted` gesetzt. Im Gegensatz dazu stellt Eve jedoch IPclip-Optionen mit falscher LI bereit (1.2;1.4), welche nicht innerhalb der SCA liegen. Die falsche LI wird durch die LI des DSLAMs (0.5;0.5) ersetzt. SF und TF werden auf `network provided/untrusted` gesetzt.

5.1.5. Automatische Anpassung der MTU des Datenpfades

Die Maximum Transmission Unit (MTU) beschreibt die maximal zulässige Größe einer PDU. Sie ist entweder durch den Protokollstandard definiert (= MTU) oder ergibt sich durch die technischen Leistungsparameter bzw. der Konfiguration der Router des jeweiligen Datenpfades (= Path MTU, PMTU). Die Ethernet-MTU beträgt z. B. 1500 Byte. Die MTU eines IP-Paketes liegt bei 65536 Byte. Im Internet sind IP-Pakete meist an die MTU der Übertragungstechnologie angepasst, um Fragmentierung zu vermeiden.

Aufgrund des Einfügens einer zusätzlichen IPclip-Option in den IP-Header vergrößern sich ein IP-Paket und damit ein Ethernet-Frame um L_{IP-Opt}, z. B. 16 Byte für den IPclip-Typ `0x03`. Es kann somit zu einer Überschreitung der PMTU auf der Übertragungsstrecke kommen. In diesem Fall werden die entsprechenden IP-Pakete entweder fragmentiert oder verworfen. Fragmen-

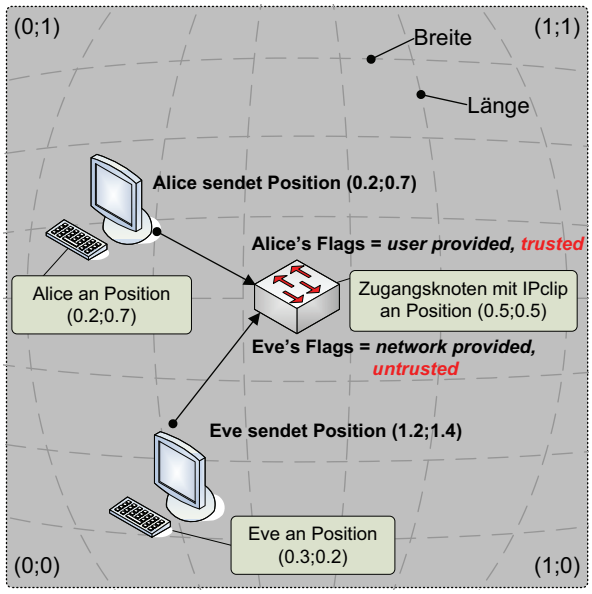


Abbildung 5.3.: Verifikation der LI durch IPclip. Die SCA und die Positionen der Nutzer sind zum besseren Verständnis normalisiert $(0 \leftrightarrow 1)$. In einer realen Umgebung sind sie z. B. durch GPS-Koordinaten gegeben (Länge & Breite).

tierung ist aber aus verschiedenen Gründen zu vermeiden [Cis06]. Fragmentierung erfordert zusätzlichen Rechenaufwand und Speicher im dem jeweiligen Netzwerkgerät. Der Empfänger muss Speicher für bereits empfangene Fragmente reservieren, um diese wieder zusammensetzen zu können. Dazu wird der größte vorhandene Puffer genutzt, da die Gesamtgröße des IP-Paketes nicht bekannt ist. Für das CPE ist dies nicht von Bedeutung, da am Rand des Netzes genügend Zeit- und Speicherressourcen zur Verfügung stehen. In Routern sind Fragmentierung & Reassemblierung allerdings ineffizient, da deren primäre Aufgabe das schnelle Weiterleiten von IP-Paketen bei höchsten Datenraten ist. Zudem muss das komplette IP-Paket erneut versendet werden, sobald ein Fragment fehlerhaft ist. Soll keine Fragmentierung erfolgen, wird das IP-Paket verworfen, welches die PMTU überschreitet. Dem Sender muss dieser Vorgang mitgeteilt werden, damit nachfolgende IP-Pakete mit angepasster Größe erzeugt werden und ein weiteres Verwerfen vermieden wird. Zeitpunkt, Art und Weise der Signalisierung hängen dabei von den Charakteristika des jeweiligen TZN ab. Typische Protokolle zum Verbindungsaufbau und -management in Ethernet-basierten TZN sind Internet-Protocol-over-Ethernet (IPoE) und Point-to-Point-Protocol-over-Ethernet (PPPoE) [Jun08]. Im Folgenden wird die MTU-Anpassung im Detail für IPoE erläutert. Für PPPoE sei auf [KWD+08b] verwiesen.

In IP-basierten Netzwerken wird zur Vereinbarung der gemeinsam genutzten, minimalen PMTU der Mechanismus der Path MTU Discovery (PMTUD) angewandt [RFC1191], welcher

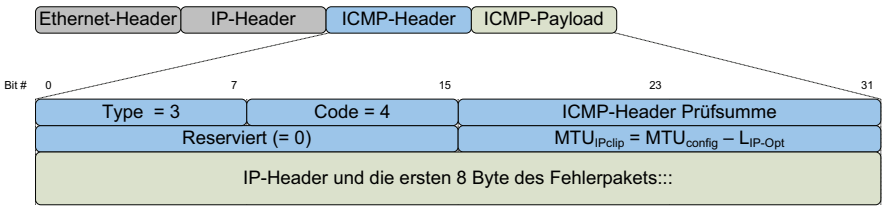


Abbildung 5.4.: Aufbau eines ICMP-Paketes mit durch IPclip modifizierter PMTU

durch jedes zu große IP-Paket stimuliert wird. Dazu müssen IP-Pakete ein gesetztes „Don't Fragment-Bit (DF)" im Flags-Feld des IP-Headers aufweisen (siehe auch Abschnitt 2.5). Dies bedeutet, dass keine Fragmentierung zu großer IP-Pakete erfolgen soll. Stattdessen werden diese IP-Pakete verworfen und es wird eine ICMP-Nachricht (Internet Control Message Protocol) [RFC0792] mit Informationen zur erlaubten Größe der PMTU an den Sender erzeugt, damit der Sender IP-Pakete mit entsprechend angepasster Größe erzeugt. Abbildung 5.4 zeigt die Struktur einer ICMP-Nachricht. Dem 4 Byte großen ICMP-Header folgt ein optionales Datenfeld. Für die Signalisierung während der PMTUD sind Type = 3 und Code = 4 (*Fragmentation required, and DF set*). In den Nutzdaten schließen sich an 2 reservierte Bytes der IP-Header und die ersten 8 Datenbytes des zu großen Pakets an.

Die IPclip-Funktionalität fügt sich transparent in die PMTUD ein. Es wird zwischen Up- und Downstream unterschieden: Im Upstream wird überprüft, ob die Größe jedes eingehenden IP-Paketes nach dem Hinzufügen einer IP-Option mit der Länge L_{IP-Opt} die allgemeine PMTU des Netzes überschreitet. Diese PMTU (MTU_{config}) ist konfigurierbar. Für Ethernet-basierte TZN hat MTU_{config} den Wert 1500 Byte. Ist $L_{Paket} + L_{IP-Opt} > MTU_{config}$, wird das Fehlerpaket verworfen und eine ICMP-Nachricht vom IPclip-Modul generiert und an die Quelle gesendet. Die MAC- und IP-Adressen für Ethernet- und IP-Header werden dem Fehlerframe entnommen. Die durch IPclip korrigierte MTU ($MTU_{IPclip} = MTU_{config} - L_{IP-Opt}$) wird in die ICMP-Nachricht eingetragen. Nachgeschaltete Netzwerkkomponenten können unter Umständen jedoch nur eine kleinere PMTU MTU_x als die allgemeine MTU_{config} verarbeiten. Deswegen wird der Downstream auf vom Netz kommende ICMP-Nachrichten überwacht. Wird eine ICMP-Nachricht erkannt (Type = 3, Code = 4), ist der Wert für MTU_x durch IPclip zu modifizieren ($MTU_{IPclip} = MTU_x - L_{IP-Opt}$). Die modifizierte ICMP-Nachricht wird dann weitergeleitet.

5.1.6. Nebeneffekte und erforderliche Rahmenbedingungen

Sowohl für eine netzwerkweite und konsistente Realisierung des TBW-Konzeptes auf Basis von IPclip als auch zur Ermöglichung der in Abschnitt 5.2 beschriebenen Anwendungsfälle sind verschiedene Randbedingungen zu definieren und Nebeneffekte zu diskutieren.

IPclip-Fähigkeit Eine wesentliche Voraussetzung für das Funktionieren von IPclip ist die Existenz eines IPclip-fähigen IP-Protokollstacks. Dieser ist in den Netzwerkelementen erforderlich, für die eine direkte Verwendung der IPclip-Optionen und LI vorgesehen ist. Andere Netzwerkelemente brauchen nicht zwingend eine Anpassung an IPclip, da jeder standardkonforme IP-Protokollstack mit IP-Optionen umgehen kann. Selbst wenn eine IP-Option einen unbekannten Typ aufweist, muss sie aus Gründen der Interoperabilität zumindest übersprungen bzw. weitergeleitet werden.

Konsistenter Geltungsbereich Für den Einsatz von IPclip ist eine komplett abgeschlossene IPclip-Domäne zwingend erforderlich. D. h., auf *jedem* Zugangsknoten eines TZN muss IPclip implementiert sein. Bereits ein einziger DSLAM ohne IPclip-Funktionalität würde ein Schlupfloch in der Netzwerkstruktur darstellen. IP-Pakete mit manipulierten Status-Bits und verfälschter LI könnten ohne durch eine vertrauenswürdige IPclip-Instanz verifiziert und validiert zu werden in das TZN und Kernnetz gelangen. Eine praktikable IPclip-Domäne kann z. B. auf Basis eines in sich abgeschlossenen Providernetzes realisiert werden.

Privatsphäre & Datenschutz Durch die Existenz von Ortsinformationen über den Ursprung von IP-Paketen und damit über den eigentlich Nutzer ergeben sich zwangsläufig Fragen bzgl. Verfügbarkeit und Speicherung privater bzw. personenbezogener Daten. In Deutschland wie auch weltweit sind u. a. Bedenken bezogen auf die Vorratsdatenspeicherung ein aktuelles Thema. Darüber hinaus werden diese Themen in vielen Bereichen der Telekommunikationsbranche diskutiert, in denen es um ähnlich sensitive und private Informationen geht, z. B. die Speicherung ab- und eingehender Telefonate und Rufnummern. Fakt ist jedoch, dass Sicherheit und Vertrauenswürdigkeit *immer* durch gewisse Einschränkungen und Kompromisse erkauft werden müssen – dies trifft insbesondere für das offene und komplexe Internet zu. Unabhängig davon bietet der vorgestellte IPclip-Ansatz dennoch Möglichkeiten, die Preisgabe von Ortsinformationen zu verhindern oder auf das notwendige Minimum zu reduzieren.

Die IPclip-Funktionalität kann nicht nur IPclip-Optionen im Upstream Datenpfad einfügen, sondern auch im Downstream wieder entfernen. Informationen dazu bietet u. a. der im Anhang unter Abschnitt B.2 vorgestellte Prototyp. Das Removal-Flag der Status-Bits einer IPclip-Option kann genutzt werden, um kenntlich zum machen, ob eine IPclip-Option wieder entfernt werden muss (siehe Abschnitt 5.1.2). Soll eine IPclip-Option wieder entfernt werden, gibt es zwei Möglichkeiten. Einerseits kann die gesamte IPclip-Option wieder entfernt werden. Andererseits ergibt es Sinn, nur die darin enthaltene LI zu löschen. Die Status-Bits, welche nur das Verifikationsergebnis darstellen, werden bis zum Ziel weitergeleitet.

Anwendungsabhängig ist es zudem nicht immer erforderlich, alle IP-Pakete mit IPclip-Optionen anzureichern. Es brauchen nur diejenigen IP-Pakete IPclip-Optionen erhalten, die ein bestimmtes Protokoll der Anwendungsschicht kapseln. Dies trifft z. B. auf den Anwendungsfall von VoIP-Notrufen in Abschnitt 5.2.1 zu.

Als dritte Möglichkeit kann anstelle von Klartextinformationen ein verschlüsseltes Format für die LI genutzt werden. Die Gebiete Kryptographie und sicheres Schlüsselmanagement sind jedoch nicht Bestandteil dieser Arbeit.

IPv4-Optionen vs. IPv6-Erweiterungsheader IPclip wurde auf Basis von IPv4 entwickelt, da dies nach wie vor das zentrale Protokoll im Internet ist. IPv6 spielt zurzeit noch eine untergeordnete Rolle. Die Nutzung von IPv4-Optionen ist aufgrund der Standardkonformität prinzipiell möglich, hat aber auch verschiedene Nachteile. Optionsbehaftete IP-Pakete werden z. B. aufgrund des unregelmäßigen IPv4-Headers in vielen Routern nur in Software im sogenannten *Slow Path* verarbeitet [Spi05, FJ04a]. Dadurch können zusätzliche Verzögerungszeiten entstehen [RW03]. Dennoch ist die Nutzung von IPv4-Optionen für verschiedene Verwaltungs- und Sicherheitszwecke durchaus üblich, da sich die zusätzliche Verzögerung im Schnitt nur in einem geringen Bereich bewegt [RW04].

IPv6 wird zum zentralen Protokoll im Internet werden (siehe Abschnitt 2.5). Optionen werden bei IPv6 in unterschiedlichen Erweiterungsheadern mitgeführt. In der Belegarbeit von Rhinow [Rhi08] wurden die nötigen Anpassungen von IPclip an die Mechanismen von IPv6 spezifiziert.

Eine weitere Variante der Optionsübertragung wird in [FJ04b] diskutiert, welche die Adaption der Idee von IPv6-Erweiterungsheadern in IPv4 über ein noch ungenutztes Bit des IPv4-Headers vorschlägt. In beiden Fällen behalten die IP-Header ihre reguläre Struktur und IP-Pakete müssen nicht im Slow Path verarbeitet werden.

Die entscheidende Einschränkung bzgl. des Gebrauchs von IP-Optionen ist jedoch die Existenz von Firewalls, welche i. Allg. vom Endteilnehmer selbst verwaltet sind und unterschiedlichst (fehl)konfiguriert sein können. Unbekannte IP-Optionen bzw. alle IP-Optionen können geblockt werden. Anders ausgedrückt – privat verwaltete Server und Firewalls müssen entsprechend richtig konfiguriert sein, um mit Optionen, sei es im IPv4-Header oder als IPv6-Erweiterungsheader, umgehen zu können. Dies betrifft nicht nur IPclip.

Wechselwirkungen mit Sicherheitsarchitekturen Ähnlich der in Kapitel 4 vorgestellten MAT-Architektur existieren zwischen IPclip und TLS bzw. SSL keine Wechselwirkungen aufgrund des Bezuges auf andere Schichten des OSI-Referenzmodells. Jedoch können sich IPclip und IPsec gegenseitig beeinflussen. Wird IPsec im Tunnelmodus genutzt (AH oder ESP), ist die Position der IPsec-Endpunkte ausschlaggebend für die Kompatibilität mit IPclip. IPclip-Optionen

müssen bereits vorliegen, wenn IPsec-Header erzeugt werden. Ein nachträgliches Einfügen ist aufgrund der Verschlüsselung nicht möglich. Wird ESP im Transportmodus genutzt, entstehen keine Wechselwirkungen, da der IP-Header sowie IP-Optionen nicht durch ESP abgesichert sind. Wird AH im Transportmodus genutzt, muss festgelegt sein, welche Teile des IP-Headers durch AH abgesichert sind. Je nach Position der IPclip-Instanz relativ zum IPsec-Endpunkt dürfen IPv4-Optionen bzw. IPv6-Erweiterungsheader nicht durch AH authentifiziert sein, da die IPclip-Funktionalität diese Felder nachträglich modifizieren muss.

5.1.7. Zwischenfazit zu Trust-by-Wire und IPclip

Der Kerngedanke des IPclip-Ansatzes ist, Vertrauenswürdigkeit auf Basis von *Trust-by-Wire* auf IP-Ebene zu realisieren. Dies geschieht durch die Anreicherung von IP-Paketen mit zusätzlichen Ortsreferenzen zu ihrem Ursprung in Form von IP-Optionen. Das Einfügen geschieht am Rand des Internets in den Teilnehmerzugangsnetzen. IPclip erfüllt auf diese Weise zwei Funktionen: einerseits die Lokalisierung bzw. Identifikation des Teilnehmers bzw. der TAL auf Basis der Ortsinformationen und andererseits die Schaffung einer Vertrauensgrundlage durch die Verifikation dieser Ortsinformationen.

Ergebnisse zum allgemeinen IPclip-Mechanismus wurden in [KWD+08b] und [DKW+08c] veröffentlicht. Zudem wurde der gesamte IPclip-Mechanismus als voll funktionsfähiger Prototyp in Hardware realisiert [WKD+08] und in [DKW+08a] und [DKW+09] vorgestellt. Weitere Informationen zum Prototyp bietet zudem Abschnitt B.2 im Anhang.

5.2. Anwendungsszenarien für IPclip

In diesem Abschnitt werden in knapper Form drei aktuelle Anwendungsszenarien für IPclip vorgestellt, welche sowohl die Flexibilität als auch die Erfordernis des TBW-Ansatzes belegen. Die Szenarien repräsentieren speziell sicherheitskritische Probleme im Bereich der Telekommunikation, in welchen aus der Schaffung einer Vertrauensbasis und der Bereitstellung von Ortsreferenzen Nutzen gezogen werden kann. Die prinzipielle Funktion von IPclip bleibt unverändert. Die Interpretation der IPclip-Optionen ist aber je nach Anwendung anders. Für tiefgreifende Informationen zu den Anwendungsfällen sei auf die jeweilige Quelle verwiesen.

5.2.1. Notrufe & Voice-over-IP

Hintergrund & Motivation Die Möglichkeit, Notrufe per Festnetz oder Handy abzusetzen, klingt banal und wird heute von Jedermann für eine Selbstverständlichkeit gehalten. Selbst

für den unglücklichen Fall, dass ein Anrufer sich nicht sprachlich oder nur eingeschränkt verständigen kann, wurden z. B. Notruf-Tasten bzw. „Röchel-Rufe“ entwickelt, die keine verbale Interaktion mit der annehmenden Einsatzleitstelle mehr erfordern. Ein Anrufer kann hier durch die Verbindung von Rufnummer und physikalischer Leitung bzw. Funkzelle identifiziert und lokalisiert werden.

Es gibt verschiedene regionale Varianten und Standards klassischer Telefon- und Mobilfunknetze. Diese sind technisch oft inkompatibel, was z. B. an den weltweit unterschiedlichen Notrufnummern erkennbar ist. Durch den Übergang klassischer Telefonie zu VoIP ins Internet heben sich diese regionalen Beschränkungen jedoch auf. Dies macht eine globale Vereinheitlichung und Standardisierung auf der einen Seite unabdingbar und auf der anderen Seite vor allem erst möglich. Ein VoIP-Anrufer ist anhand seiner IP-Adresse nun aber nicht mehr eindeutig identifizierbar und zu lokalisieren. Die Allgemeinen Geschäftsbedingungen (AGBs) des weit verbreiteten VoIP-Tools Skype weisen z. B. darauf hin, dass Notrufe eben *nicht* unterstützt werden [SKY]. Ein Notruf, bei dem u. a. Zeit eine kritische Rolle spielt, kann bei VoIP nicht mehr ohne Weiteres an die korrekte Einsatzleitstelle weitervermittelt werden. Das Fehlen einer eindeutigen und vertrauenswürdigen Referenz zum Anrufer und dessen geographischer Position kann im Fall von Notrufen fatale Folgen haben. Dies ist vor allem bei mobiler VoIP-Nutzung der Fall; und Mobilität und mobile Geräte liegen i. Allg. immer mehr im Trend (siehe Abschnitt 3.2.3).

Bei drahtgebundener Nutzung von VoIP findet die Bestimmung der korrekten Position des Anrufers und die Verbindung mit der entsprechenden Einsatzleitstelle momentan auf Basis einer Reihe komplexer Datenbankanfragen statt [MHRSW05, RP08, Hig07, RSPN08], sogenannter Push/Pull-Verfahren. Daran sind mehrere unabhängige Netzwerkinstanzen beteiligt. Dies resultiert in einem immensen Verwaltungsaufwand. Zudem sind diese Verfahren nicht standardisiert, sondern beschreiben nur die zurzeit günstigste Vorgehensweise. Diese sieht z. T. auch die manuelle Konfiguration der aktuellen Position durch den Nutzer selbst vor. Einerseits sind diese Vorgehensweisen in mobilen Umgebungen nicht anwendbar, da hier die Position eines Nutzers stark variiert. Andererseits ist aufgrund des Fehlens einer einheitlichen Notruf-Regelung für VoIP eine allgemeine Lösung notwendig. Für weitere Details zum Stand der Technik sei auf die genannten Referenzen als auch auf [KWD+08b] verwiesen.

IPclip im VoIP-Szenario In Kontrast zum Stand der Technik auf Basis von Push/Pull-Verfahren werden mittels IPclip zuverlässige Ortsinformationen in größtmöglicher Nähe zum Kunden auf IP-Ebene eingefügt – auf den Linecards, den Eintrittspunkten in die TZN. Da ein Nutzer *nicht* gezwungen ist, LI bereitzustellen, und diese auch fehlkonfiguriert sein kann, sind drei Fälle zu unterscheiden, welche in Abbildung 5.5 illustriert sind.

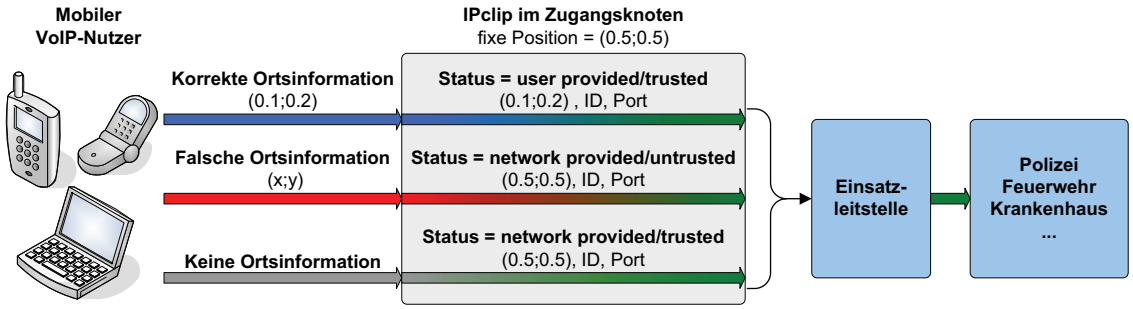


Abbildung 5.5.: Notrufe bei VoIP mit IPclip (normalisierte Koordinaten, siehe Abb. 5.3)

- Durch das CPE eines Nutzers kann korrekte LI eingefügt worden sein (blauer Pfeil). Korrekt bedeutet, dass die LI innerhalb der SCA des entsprechenden Zugangsknotens liegt (siehe Abbildung 5.3). Durch die IPclip-Instanz validierte LI wird um die TAL-Identifikation (ID, Port) erweitert und weitergeleitet.

- Ergibt die Verifikation, dass die vom Nutzer bereitgestellte LI falsch ist (roter Pfeil), d. h. sie liegt außerhalb der SCA, wird sie durch die fixe LI des Zugangsknotens (0.5;0.5) inklusive der TAL-Identifikation ersetzt.

- Sind gar keine IPclip-Optionen durch den Nutzer bereitgestellt (grauer Pfeil), wird in jedem Fall die fixe LI des Zugangsknotens und die TAL-Identifikation eingefügt.

Auf diese Weise verlassen IP-Pakete den Zugangsknoten in Richtung Kernnetz nur mit IPclip-Optionen mit validierter LI (grün markiert). Stammt die LI direkt vom Notrufenden, kann diese in der zentralen Einsatzleitstelle direkt zur Abbildung auf eine zivile Adresse bzw. zur Lokalisierung auf einer Karte genutzt werden, um den Notruf an die entsprechende Leitstelle der Polizei, Feuerwehr, oder des medizinischen Notfalldienstes weiterzuleiten. Wurde die LI auf dem Zugangsknoten ersetzt oder neu eingefügt, ist eine Ortsreferenz mit der Genauigkeit der SCA gegeben inklusive den Angaben zur TAL-Identifikation. Beide Varianten stellen eine Verbesserung der aktuellen Situation dar, insbesondere bei mobiler VoIP-Nutzung. Zudem ist IPclip kompatibel zu bisherigen Verfahren und unterstützt diese durch die zusätzliche LI.

Für den Verbindungsaufbau bei VoIP wird vorrangig das Session Initiation Protocol (SIP) [RFC3261] genutzt, für die eigentliche Sprachverbindung das Real-Time Transport Protocol (RTP) [RFC3550]. Im VoIP-Szenario ist es sinnvoll, nicht alle IP-Pakete mit IPclip-Optionen anzureichern, sondern ausschließlich SIP-Pakete. Da SIP ein Protokoll der Anwendungsschicht ist und eine unregelmäßige Headerstruktur aufweist, ist es jedoch nicht ohne Weiteres möglich, diese Pakete zu filtern. Die Arbeiten in [DKT07, Dan06] bieten jedoch eine geeignete Grundlage zur Text-

& Mustersuche, um in kompletten IP-Paketen nach Hinweisen eines SIP-Verbindungsaufbaus zu suchen, z. B. nach einem `INVITE` oder einer bestimmten SIP-URI für Notrufe.

Mehrwert durch IPclip IPclip stellt glaubwürdige und aktuelle Ortsreferenzen zur Lokalisierung des Anrufers bereit, welche mindestens die Genauigkeit der SCA und TAL-Identifikation haben. Somit werden VoIP-Notrufe generell unterstützt und für die mobile Nutzung von VoIP ermöglicht, ohne dass der Nutzer seine aktuelle Position mit Datenbanken synchronisieren muss. Selbst bei ungewolltem Verbindungsabbruch steht durch die IPclip-Optionen bereits gesendeter IP-Pakete eine Ortsreferenz zur Verfügung. Mit IPclip können Notrufe bei ortsgebundener sowie mobiler VoIP-Nutzung direkt zur verantwortlichen Leitstelle umgelenkt werden.

5.2.2. Bekämpfung von E-Mail-Spam

Hintergrund & Motivation Aufgrund des Erfolgs des Internets hat sich E-Mail zu einem der wichtigsten Kommunikationsmedien entwickelt. Im Vergleich zu herkömmlicher Briefpost können Millionen von Menschen mit einem Bruchteil des zeitlichen und finanziellen Aufwands erreicht werden. Diese Vorteile sind jedoch auch gleichzeitig Gründe für eine der Achilles-Fersen des Internets – *Spam*. Einige Ursachen dafür wurden bereits in Abschnitt 3 genannt. Der Hauptgrund ist jedoch das zentrale E-Mail-Protokoll selbst. Das in die Jahre gekommene SMTP wurde nicht für die derart große Nutzergemeinde des heutigen Internets entwickelt [Kle04]. Es bietet keine Authentifizierungs- und Sicherheitsmechanismen. Details zur Struktur von E-Mails und Funktion von SMTP können [Hal05] und [RFC2821] entnommen werden.

Der Spam-Anteil im gesamten E-Mail-Verkehr beträgt zurzeit ca. 90 % [Nel03, Sym07a]. Seit Auftreten des ersten Spams [Str03] werden E-Mails nicht mehr nur für Werbezwecke missbraucht. Spam gefährdet das Internet auf vielfältige Weise: DDoS-Angriffe auf E-Mail-Server, Nichtverfügbarkeit von E-Mail-Konten, Belästigung durch Werbung, finanzieller Ruin durch dubiose Angebote, Einschleusen von Schadsoftware [Erb05], Kosten im Fall volumenbasierter Online-Tarife, Beeinträchtigung des Vertrauens der Nutzer in das Internet [Fal03] sowie die Verschwendung von Arbeitszeit durch manuelle Spam-Auslese. Zur Detektion und Verhinderung von Spam existiert eine breite Palette an Werkzeugen, jedoch bietet kein Ansatz 100%igen Schutz. Gleichzeitig entwickeln Spammer ständig neue Methoden, E-Mails zu manipulieren und glaubhaft erscheinen zu lassen, damit sie erfolgreich ausgeliefert werden. Dabei erlaubt SMTP die Manipulation fast aller Elemente einer E-Mail [Bon04]. Die Entwickler von Abwehrmaßnahmen und Anti-Spam-Programmen können nur reagieren. Details sowohl zu Techniken von Spammern als auch Methoden zur Spam-Bekämpfung bieten u. a. [KWD+08c], [GHR05] und [Lea07].

Trotz Maßnahmen zur Sensibilisierung der Nutzer und hohen Trefferquoten von Spam-Filtern

kann aufgrund der großen Flut von Spam immer noch ein für den Spammer lukrativer Anteil die Postfächer der Empfänger erreichen. Zudem ist die Detektion von Spam durch Falsch-Positive und -Negative geprägt, so dass E-Mails oft noch einmal per Hand selektiert werden. Da der treibende Faktor bei Spam primär finanzieller Natur ist, gibt es prinzipiell drei Ansätze, um das Spam-Aufkommen zu reduzieren:

- Deutlich weniger bzw. kein Spam erreicht sein Ziel.
- Der Aufwand des Versendens vieler E-Mails wird für den Spammer zu kostspielig.
- E-Mails können eindeutig zurückverfolgt werden, was jedoch gerade im paketvermittelten Internet problematisch ist [San07].

Während Maßnahmen bzgl. des ersten Punktes Informationen zur Analyse nutzen, die durch Spammer manipulierbar sind, können Ansätze bzgl. des zweiten Punktes auch negativen Einfluss auf den Versand regulärer E-Mails haben. Der TBW-Ansatz mit IPclip greift das Spam-Problem von einer anderen Seite auf und nutzt dabei entscheidende Eigenschaften von Spam. Spammer möchten in jedem Fall unerkannt bleiben. Spam-E-Mails geben deshalb einen falschen Ursprung vor. Die wahre Quelle der E-Mail wird verschleiert. Normale und erwünschte E-Mails bzw. deren Absender tun dies nicht. Spammer sind zudem ungeduldig und müssen viele E-Mails in kurzer Zeit versenden, um profitabel zu bleiben.

IPclip im Anti-Spam-Szenario Am Versand und Transport einer E-Mail sind typischerweise verschiedene Instanzen beteiligt. Der Sender erzeugt und versendet eine E-Mail mittels einer E-Mail-Software an den Mail Transfer Agent (MTA), bei dem er registriert ist. Ein MTA ist ein E-Mail-Server eines ISPs, z. B. von Microsoft oder Yahoo. Entweder leitet der MTA die E-Mail anhand der Ziel-Adresse direkt zum Empfänger weiter, wenn dieser beim gleichen MTA registriert ist, oder er sendet sie zum MTA des Empfängers (ggf. über mehrere Zwischenstationen). Die Übertragung einer E-Mail erfolgt dabei mittels SMTP.

Für die Nutzung von IPclip sind an dieser Stelle drei Dinge relevant. Zum einen liegen IPclip und IPclip-Optionen nicht im Einflussbereich der Spammer und können nicht manipuliert werden. Zum anderen findet der E-Mail-Transfer auf SMTP-Ebene statt. Der wesentliche Punkt ist jedoch, dass IP an jedem MTA terminiert und der IP-Header neu aufgesetzt wird. Dies bedeutet, dass IPclip-Optionen auf IP-Ebene im Netz am ersten MTA verloren gehen. Aus diesem Grund muss der Inhalt der ursprünglichen IPclip-Option gesichert werden. Dazu ist vorgesehen, dass im *ersten* MTA auf der Übertragungsstrecke die in jedem Fall vorhandene IPclip-Option in den Header der E-Mail kopiert wird. Die RFCs [RFC0822, RFC2821] definieren eine bestimmte Formatierung für optionale Einträge im Header. Abbildung 5.6 zeigt die Struktur

```
From - <timestamp>
 X-IPclip-Status: 1100
 X-IPclip-Type: GPS
 X-IPclip-LI: <longitude;latitude>
 X-IPclip-Port: <number>
 X-IPclip-DSLAM: <identification>
 X-IPclip-MTA: mx.senderhome.net [86.165.10.2]
Return-Path: <sender@senderhome.net>
Received: from ...
```

Abbildung 5.6.: Mögliche Struktur eines IPclip-Eintrags im E-Mail-Header

der IPclip-Felder im E-Mail-Header. Auf diese Weise wird die IPclip-Option bis zum Empfänger übertragen, welchem nun eine glaubwürdige Ortsreferenz zur Quelle der E-Mail vorliegt.

Dieses Vorgehen wirft verschiedene Fragen auf, für deren Beantwortung auf [KWD+08c] verwiesen wird. Die Literaturquelle enthält u. a. Details zur Interpretation der Status-Bits, zur Handhabung und Manipulierbarkeit der IPclip-Informationen im E-Mail-Header, zur Kompatibilität mit existierenden Ansätzen bzgl. E-Mail-Sicherheit und sie differenziert zwischen verschiedenen möglichen Übertragungswegen einer E-Mail.

Mehrwert durch IPclip IPclip klassifiziert keine E-Mails als Spam und verhindert auch nicht, dass Spam die Mailbox eines Nutzers erreicht! Die primäre Nutzung von IPclip ist nach wie vor die Schaffung einer zweckmäßigen Vertrauensbasis à la Trust-by-Wire, da das Fehlen einer verlässlichen und eindeutigen Referenz auf den Ursprung von E-Mails (und allgemein IP-Daten) einer der Hauptgründe ist, weshalb sich Spammer hinter manipulierten Senderinformationen verstecken können. Existierende Anti-Spam-Tools wie z. B. SpamAssassin [SPA] sind auch weiterhin für die Klassifikation einer E-Mail als erwünscht bzw. unerwünscht zuständig.

Einerseits stellt IPclip mit den Status-Flags im E-Mail-Header (siehe Abb. 5.6) ein zusätzliches, höchst glaubwürdiges Entscheidungskriterium für die bisher angewandten Methoden wie Heuristiken, zentrale Spam-Datenbanken und lernfähige Filter bereit. SpamAssassin würde anhand der Flags z. B. einen entsprechend hohen Wert auf die Spam-Wertung von E-Mails addieren. E-Mails werden dann gemäß ihrer Spam-Wertung erst verzögert zugestellt. Somit ist eine zuverlässigere Klassifikation von Spam möglich. Andererseits bieten die vertrauenswürdigen Ortsinformationen (GPS-Koordinaten & TAL-Identifikation) zusätzlich Aufschluss über die Herkunft von E-Mails. Dadurch können als Spam klassifizierte E-Mails zurückverfolgt werden. IP-Adressen konnten bisher u. a. durch WHOIS-Abfragen aufgelöst werden [RFC3912], jedoch nur mit ungenügender Auflösung. Dieser Aspekt spielt z. B. in Honeypots [Bon04, The06] eine wichtige Rolle. Honeypots sind präparierte Rechner und Netze, um Schadsoftware anzulocken

und zu analysieren.

Mit Blick auf die Massen von E-Mails ist eine schnelle Klassifikation und Abschätzung, ob ein Absender vertrauenswürdig ist oder nicht, notwendig. IPclip stellt geeignete Kriterien dafür bereit. Spammer verstecken sich meist hinter falschen Absender-Adressen und nutzen dynamisch wechselnde IP-Adressen, jedoch ist IPclip unabhängig von diesen Techniken und liegt außerhalb des Einflussbereichs von Spammern.

5.2.3. Schutz vor Phishing im Internet

Hintergrund & Motivation Das Ziel von Phishing ist Identitätsdiebstahl durch Erlangung sensitiver Informationen, z. B. Kreditkartenkenndaten oder Nutzername & Passwort für Online-Banking, welche direkt missbraucht oder an Dritte veräußert werden. Während einer Phishing-Attacke, z. B. initiiert durch Spam (siehe Abschnitt 5.2.2), gibt ein Phisher vor, eine vertrauenswürdige Instanz, Person oder Einrichtung zu sein. Um dies zu erreichen, werden dem potentiellen Opfer manipulierte Hyperlinks und trügerische Webseiten präsentiert, welche den wahren Webseiten täuschend ähnlich sind. Dadurch ist das Opfer der Überzeugung, sich in einem gesicherten Bereich aufzuhalten. Durch Eingabe persönlicher Kundendaten authentifiziert sich ein Nutzer beim Homebanking-Service seiner Bank und stellt ein *einseitiges* Vertrauensverhältnis zwischen sich und der Bank her, welches somit nicht ausreichend ist. Es lässt keinen Rückschluss auf die Vertrauenswürdigkeit der vermeintlichen Webseite zu.

Der Begriff *Phishing* wurde in den 90-iger Jahren durch das Ausspionieren von Kundenkonten von America Online (AOL) geprägt [Oll04], zumeist basierend auf Spam. Mittlerweile stellt Phishing eine kritische Bedrohung im Internet dar und verursacht erheblichen finanziellen Schaden. Nach Angaben des Government Accountability Office [Uni07] beträgt der Schaden allein in den USA „[...] $49.3 billion in 2006 for identity theft and $1 billion annually due to phishing [...]“. Den vielen verschieden Phishing-Methoden steht eine relativ begrenzte Anzahl an Abwehrmaßnahmen gegenüber. Ein Mix verschiedener Sicherheitsmechanismen bietet zwar guten Schutz, erfordert jedoch ein gewisses Maß an Fachwissen. Der Schwachpunkt ist nach wie vor der unbescholtene Durchschnittsnutzer, dem die Gefahren des Internets nicht bewusst sind. In [KWD+08a] und [Oll04] werden detaillierte Informationen über sowohl typische Methoden von Phishern als auch Maßnahmen gegen Phishing-Angriffe gegeben. Am Beispiel des Online-Bankings soll gezeigt werden, wie IPclip in einem Phishing-Szenario Abhilfe schaffen kann.

IPclip im Anti-Phishing-Szenario Eine typische Phishing-E-Mail fordert einen Kunden auf, die Daten seines Bank-Accounts auf einer fingierten Webseite zu aktualisieren. Ohne IPclip würde das Opfer ungewarnt bleiben und der Phisher die eingegebenen Daten abfangen. Mit IPclip kann

die Kommunikationsverbindung zwischen dem tatsächlichen Kreditinstitut und dem Nutzer gesichert werden. Dazu stellt die Bank eine öffentliche, offizielle Ortsinformation in Form einer IPclip-LI – im Folgenden Signatur genannt – zur Verfügung. Diese Signatur identifiziert seine Internet-Präsenz bzw. seinen Webserver *eindeutig*. Der Nutzer eines Online-Banking-Services vergleicht nun die öffentliche Signatur der Bank mit der in den IPclip-Optionen enthaltenen LI, welche jedes IP-Paket mit sich tragen muss, das von der Bank stammt. Sowohl dieser Vergleich als auch die Status-Flags der IPclip-Option, welche ihre Herkunft und Glaubwürdigkeit beschreiben, können verschiedene Aktionen auslösen. Im Fall einer nicht vertrauenswürdigen Internetseite (`untrusted` oder `network provided`) sollte dies die Blockierung der Seite oder zumindest eine deutliche Warnung des Nutzers nach sich ziehen.

Abbildung 5.7 illustriert dieses Szenario. Auf der linken Seite befinden sich der Webserver der Bank und eine manipulierte Phishing-Webseite. Sie sind über die Zugangsknoten A und B mit dem Internet verbunden, wobei ein Phisher üblicherweise nicht über denselben Zugangsknoten an das Internet angeschlossen ist wie die Bank. Kommuniziert ein Nutzer mit dem echten Bankserver, stellt dieser in seinen IP-Paketen korrekte Ortsinformationen bereit (0.2;0.7). Die IPclip-Instanz im Zugangsknoten A verifiziert diese als `user provided/trusted` und fügt zudem die TAL-Identifikation (A, Port X) hinzu. Der Nutzer des Online-Banking-Dienstes vergleicht die mitgelieferte validierte LI mit der öffentlichen Signatur, welche in einer Signatur-Datenbank abgelegt ist. Diese Datenbank ist *statisch*, um eine automatische Generierung gültiger aber manipulierter Signatureinträge durch einen Phisher selbst auszuschließen. Bei Übereinstimmung kann mit dem Online-Banking fortgefahren werden. Ein Phisher hat nun verschiedene Möglichkeiten, LI in Form von IPclip-Optionen bereitzustellen. Kennt er z. B. die Ortsinformation des Bankservers, kann er diese missbrauchen und vortäuschen. Ist er am selben Zugangsknoten wie die Bank angeschlossen, wird die LI aus Sicht dieser IPclip-Instanz zwar als `user provided/trusted` validiert, jedoch unterscheiden sich LI und Signatur in der TAL-Identifikation – ein direktes Beispiel für TBW. Erhält der Phisher über einen anderen Knoten Zugang zum Internet, unterscheiden sich die verifizierten IPclip-Optionen (grüne Pfeile) ebenfalls immer von der in der statischen Datenbank abgelegten Signatur. Somit hat ein Phisher praktisch keine Möglichkeit, eine LI vorzutäuschen, die der Signatur des Bankservers gleich ist. Dies setzt jedoch zwei Dinge voraus. Eine Bank bzw. deren IT-Infrastruktur muss IPclip-Optionen in ausgehendem Datenverkehr einfügen. Weiterhin muss eine Signatur in einer öffentlichen Datenbank wie in Abbildung 5.7 oder auf andere Weise dem Nutzer zum Vergleich zur Verfügung stehen. Es gibt jedoch naheliegende Gründe, dass ein Kreditinstitut seine Signatur, welche Ortsreferenzen und Zugangsportnummern der eigenen IT-Infrastruktur enthält, eben *nicht* offenlegen sollte bzw. dies nicht in einer lesbaren Form tut. Dazu stellt [KWD+08a] u. a. ein angepasstes Szenario mit verschlüsselten Signaturen vor.

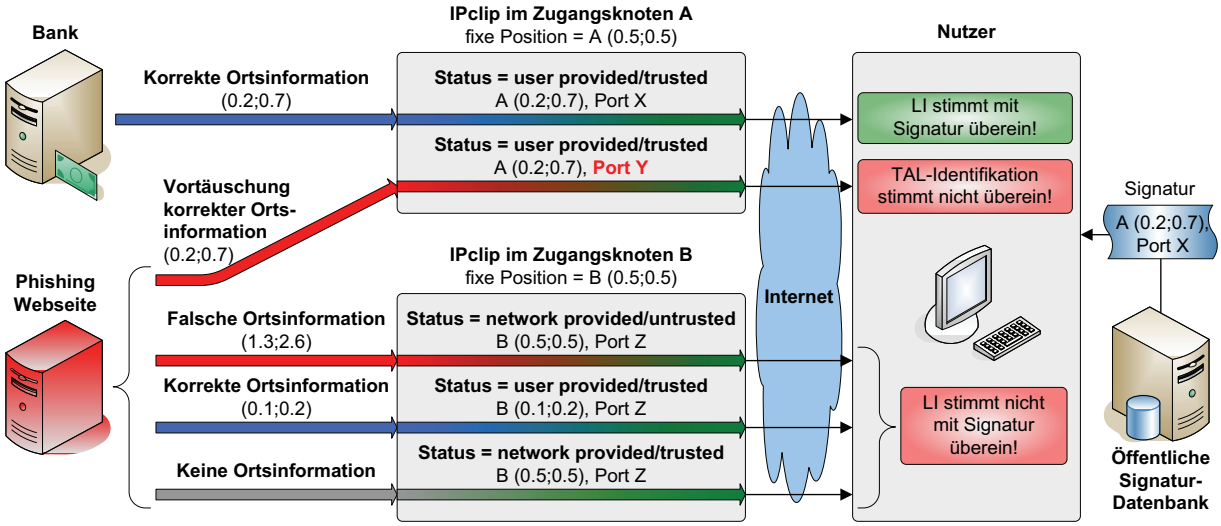


Abbildung 5.7.: Anti-Phishing-Szenario mit IPclip (normalisierte Koordinaten bzgl. der Zugangsknoten A und B, siehe Abb. 5.3)

Mehrwert durch IPclip Bisherige Mechanismen gegen Phishing versuchen, IP- und E-Mail-Adressen, den Inhalt von E-Mails und das Verhalten von Webseiten zu analysieren. Aber auch diese Dinge sind – ähnlich dem Anwendungsfall in Abschnitt 5.2.2 – alle im Einflussbereich der Angreifer. Mit IPclip können jedoch vertrauenswürdige Kennzeichen und Kriterien auf IP-Ebene zur Verfügung gestellt werden, die nicht im Einflussbereich von Phishern liegen, sondern in der Verwaltungsdomäne der Netzbetreiber und ISPs. Da die Markierung von IP-Paketen mit einer LI in den Zugangsknoten nicht umgangen werden kann, werden Phisher in eine Position mit geringem Bewegungsspielraum gedrängt. Bzgl. Phishing besitzt ein IPclip-fähiges Netz somit die Vorzüge, in erster Linie Phishing-Versuche durch den Vergleich von LI und Signatur zu detektieren und zu verhindern. Darüber hinaus sind die Ortsinformationen zur Rückverfolgung zum Ursprung des Phishing-Angriffs nutzbar. IPclip schafft ein *gegenseitiges* Vertrauensverhältnis zwischen Bank und Nutzer. Die Bank authentifiziert sich durch ihre IPclip-LI, der Nutzer durch seine Kundendaten. Ohne IPclip existiert nur eine *unidirektionale* Vertrauensbasis, welche aufgrund der existenten Gefahren im Internet nicht mehr adäquat ist.

5.3. Zusammenfassung des Kapitels

Kapitel 5 hat mit IPclip einen neuartigen Mechanismus zum Einsatz in aktuellen und zukünftigen TZN vorgestellt. Das TBW-Konzept und das IPclip-System sind auf der Netzwerkschicht im OSI-Referenzmodell angesiedelt. Die primären Ziele sind die Realisierung von Vertrauenswür-

digkeit und die Identifikation des Kunden bzw. der TAL anhand eindeutiger geographischer Ortsreferenzen im paketvermittelten Internet. Dadurch wird einerseits ein insgesamt höheres Sicherheitsniveau erreicht und andererseits die Migration neuer Dienste ins Internet unterstützt. Das IPclip-System realisiert dabei die technische Basis zur Bereitstellung der Ortsinformationen. Von vertrauenswürdigen Ortsreferenzen über den Ursprung von IP-Datenverkehr kann dann in verschiedensten Anwendungsfällen profitiert werden. Dazu wurden in Abschnitt 5.2 aktuelle, sicherheitskritische Szenarien vorgestellt, in welchen ein direkter Mehrwert aus den durch IPclip hinzugefügten Orts- und Senderinformationen erzielt wird: Notrufe in VoIP-Umgebungen, Bekämpfung von E-Mail-Spam und Schutz vor Phishing. Tabelle 5.4 fast sie zusammen.

Tabelle 5.4.: Zusammenfassung der Anwendungsbeispiele für IPclip

Anwendungsfall	Notrufe [KWD+08b]	Spam [KWD+08c]	Phishing [KWD+08a]
Wechsel von …zu …	PSTN ⇒ VoIP	Briefpost ⇒ E-Mail	Bank-Filiale ⇒ Online Banking
Adressierung	Rufnummer vs. IP-Adresse & SIP-URI	Postanschrift vs. IP- & E-Mailadresse	Bankangestellter vs. URL & IP-Adresse
Problem?	fehlender Trust-by-Wire		
Lösungsansatz	IPclip		
IPclip-Nutzung	LI	Status-Flags + LI	LI ≡ Signatur?
Zweck	Lokalisierung des Notrufenden	Klassifizierung & Nachverfolgung	Schaffung einer gegenseitigen Vertrauensbasis

Teil III.

Architektonische Untersuchungen für Systems-on-Chip

The network is the computer.
(John B. Gage, Sun Microsystems)

Kapitel 6.

Entwicklung eines Network-on-Chip

Kapitelstruktur

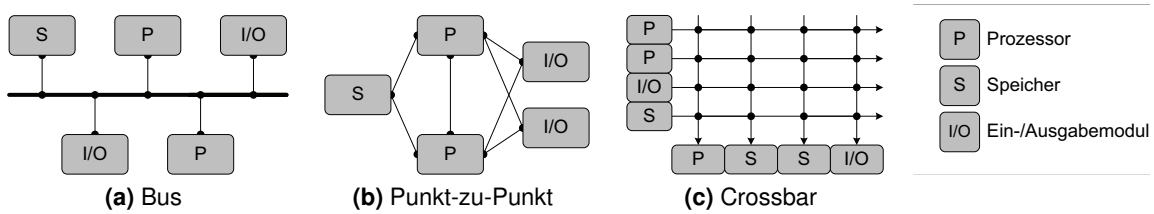
Schwerpunkt dieses Kapitels ist die Entwicklung einer adäquaten Kommunikationsarchitektur für integrierte Systeme. Dazu gibt Abschnitt 6.1 einen Überblick über verschiedene Varianten von Kommunikationsinfrastrukturen. Die Abschnitte 6.2 bis 6.4 stellen die grundlegende Network-on-Chip-Architektur sowie entwickelte Optimierungsansätze vor und bewerten diese.

6.1. Kommunikationsinfrastrukturen für Systems-on-Chip

Ein System-on-Chip (SoC) besteht aus unabhängigen Teilmodulen, sogenannten IP-Cores, welche miteinander kommunizieren [JW05, AH06]. Typische IP-Cores sind z. B. Datenspeicher, Prozessoren und Ein-/Ausgabemodule. Die einem SoC zugrunde liegende Kommunikationsinfrastruktur hat wesentlichen Einfluss auf die Effizienz des Systems. Im Bereich komplexer hoch integrierter SoCs spielen bei der Realisierung der Chip-internen Kommunikation verschiedene Aspekte eine Rolle. Einerseits sind Kosten in Bezug auf Flächen-, Ressourcen- und Energieverbrauch gering zu halten. Andererseits müssen Leistungsparameter wie Bandbreite, erzielbarer Durchsatz und Latenz den Anforderungen der jeweiligen Anwendung genügen. Zudem muss das Kommunikationsmedium flexibel skalierbar sein sowohl mit Blick auf das Mooresche Gesetz als auch bzgl. der Anzahl der anzuschließenden IP-Cores. Weiterhin sind zeitliche und ökonomische Aufwendungen für Entwurf und Test eines SoCs kritisch. Abstraktion und Wiederverwendbarkeit sind deshalb entscheidend. Verschiedene Ansätze sind im Folgenden kurz vorgestellt.

6.1.1. Klassische Ansätze

Zu den klassischen Ansätzen zählen Punkt-zu-Punkt-Verbindungen, Busse und Crossbars [DYN03, DT03, BM06a]. Abbildung 6.1 stellt diese schematisch dar. Im Bereich der SoCs ist ein zentraler geteilter Bus das bisher meistgenutzte Kommunikationsmedium. Bekannte kommerzielle Vertreter sind u. a. AMBA, Wishbone, CoreConnect und Avalon [HP97, SLKH02, MS06], welche standardisierte Schnittstellen bieten. Im Gegensatz dazu erfolgt bei Punkt-zu-Punkt-Verbindungen die Kommunikation zwischen IP-Cores über exklusive, angepasste Kommunikationskanäle. Dies erzielt für eine feste Anwendung immer die beste Leistung, da die jeweilige Bandbreite eines Kanals nicht gemeinsam genutzt und geteilt wird wie bei einem zeitgemultiplexten Bus. Crossbars wiederum sind Schaltmatrizen, die jede mögliche Verbindung von m Datenquellen zu n Datensenken realisieren. Für die Kommunikation in einem SoC bieten sie vollste Flexibilität und eine hohe Kommunikationsleistung.



(a) Bus **(b)** Punkt-zu-Punkt **(c)** Crossbar

Abbildung 6.1.: Klassische Kommunikationsinfrastrukturen für SoCs

Die Notwendigkeit nicht nur leistungsstarker sondern gleichzeitig skalierbarer Kommunikationsinfrastrukturen für SoCs ergibt sich durch die stetige Verkleinerung der Strukturgrößen und der damit wachsenden Komplexität von ICs. In [Mat97, HMH01] werden die Auswirkungen der zunehmenden Integrationsdichte auf lokale und globale Leitungen untersucht. Während die Eigenschaften lokaler Leitungen mit den Technologiegenerationen i. Allg. Schritt halten, haben globale Leitungen konstante Längen und skalieren nicht bzw. nicht in gleichem Maße. Die kapazitive Last C_L einer Leitung berechnet sich nach (6.1) aus der Grundkapazität pro Längeneinheit C_0 und der Leitungslänge l. Der Leitungswiderstand R_L ergibt sich nach (6.2). Bei zunehmender Integration, d. h. sinkenden Strukturgrößen, nimmt R_0 zu und C_0 sinkt. Da C_0 jedoch nur geringfügig sinkt, steigt die Leitungsverzögerung t_D [HMH01]. Eine einfache Approximation für t_D ist durch (6.3) gegeben. Für typische CMOS-Technologien berechnet sich die Verlustleistung P_V einer Leitung nach (6.4) (ohne Berücksichtigung von Leckströmen). α ist die Aktivität, die Wahrscheinlichkeit eines 0-1-Pegelwechsels, und f die Taktfrequenz. Zwischen l und t_D besteht eine quadratische Abhängigkeit ($t_D \sim l^2$). P_V ist linear abhängig von l ($P_V \sim l$). Steigt l, verschlechtern sich t_D und P_V.

$$C_L = C_0 \cdot l \tag{6.1}$$

$$R_L = R_0 \cdot l \tag{6.2}$$

$$t_D = \frac{C_L \cdot R_L}{2} = \frac{C_0 \cdot R_0 \cdot l^2}{2} \tag{6.3}$$

$$P_V = C_L \cdot {V_{dd}}^2 \cdot f \cdot \alpha = C_0 \cdot l \cdot {V_{dd}}^2 \cdot f \cdot \alpha \tag{6.4}$$

Die genannten Zusammenhänge zeigen sich deutlich in den Skalierungseigenschaften bzgl. Leistungsfähigkeit und Energieverbrauch der klassischen Ansätze. Arteris Inc. [Art05], Lee et al. [LCOM07] und Bolotin et al. [BCGK04] analysieren die Skalierungseigenschaften mithilfe asymptotischer Kostenfunktionen und stellen sie neuen Ansätzen gegenüber. Neue Technologieschritte erlauben einerseits die Integration einer höheren Anzahl von IP-Cores in einem SoC, jedoch sinkt mit steigender Anzahl an IP-Cores der erzielbare Gesamtdurchsatz eines gemeinsam genutzten Busses. Darüber hinaus basiert ein Bus auf langen globalen Signalleitungen, deren physikalische Bandbreite durch die Leitungsverzögerung t_D bestimmt ist. Wie oben bereits genannt, skaliert t_D nicht für globale Leitungen. Zudem erhöht sich die kapazitive Last eines Busses zusätzlich mit der Anzahl der an ihn angeschlossenen IP-Cores, wodurch auch die Leistungsaufnahme (P_V) steigt. Hierarchische bzw. segmentierte Busse können Abhilfe schaffen, lösen aber das eigentliche Problem nicht. Mit zunehmender Komplexität von SoCs zeigen Punkt-zu-Punkt-Verbindungen und Crossbars ein ähnliches Verhalten. Insbesondere bei

Punkt-zu-Punkt-Verbindungen ist die Spezialisierung auf eine einzige SoC-Applikation von Nachteil, da nicht von Modularität und Wiederverwendbarkeit profitiert werden kann.

In einem SoC werden zunehmend mehr Transistoren und damit mehr IP-Cores untergebracht. Während die Grenze bisher bei ca. 10 IP-Cores lag [JT03], werden es in Zukunft bis weit über 100 sein. Intel's 80-Prozessor-Chip [VHR+07], Cisco's Carrier Routing System CRS-1 mit 192 Xtensa RISC-Prozessorkernen pro SoC [Wil04] oder Tilera's Tile64 [Til08] sind beeindruckende aktuelle Beispiele. Derartig komplexe Systeme sind nicht mehr verarbeitungsorientiert, sondern *kommunikationsorientiert*. Die klassischen Ansätze sind dafür ungenügend. Ho fasst in [HMH01] zusammen, dass die Entwicklung komplexer SoCs zukünftig vor allem durch Modularität sowie skalierbare und performante Kommunikationsressourcen geprägt sein wird. Deswegen sind neue Konzepte zur Handhabung der steigenden SoC-Komplexität erforderlich.

6.1.2. Networks-on-Chip

Networks-on-Chip (NoCs) sind ein neuartiges Kommunikationsparadigma für SoCs [GG00, BM02, KJM+02]. Der Titel eines der meistreferenzierten Artikel zu Networks-on-Chip beginnt mit „*Route Packets, Not Wires* [...]" [DT01] und beschreibt in prägnanter Form bereits deren Grundprinzip: die Chip-interne Kommunikation basiert nicht mehr auf globalen synchronen Signalen, sondern auf dem asynchronen Austausch von Paketen, welche über ein verteiltes Netzwerk aus Routern und Kanälen übertragen werden. Globale Signalleitungen werden aufgetrennt. Parasitäre physikalische Einflüsse können so reduziert und auf lokale Regionen begrenzt werden, da t_D und P_V durch ein kleineres l ebenfalls skalieren (siehe Formel (6.3) und (6.4)).

In Abbildung 6.2 sind NoCs zusammen mit typischen makroskopischen Netzwerkklassen in ein Diagramm eingeordnet. Interessant erscheint, dass mit geringerer räumlicher Ausdehnung normalerweise auch geringere Datenraten einhergehen. NoCs hingegen sind nur wenige Quadratmillimeter groß, weisen jedoch höchste Datenraten auf, die mit denen von WAN- und GAN-Technologien vergleichbar sind. Diese *hohe Kommunikationsdichte* weist NoCs als eine besondere Art von Netzwerken aus und belegt den Wandel zu kommunikationsorientierten SoCs. Zudem spielt somit die Minimierung des Energieverbrauchs der NoC-Infrastruktur eine wesentliche Rolle im SoC-Entwurf. Low-Power-Aspekte sind jedoch nicht Teil dieser Arbeit. Es wird auf weiterführende Literatur, z. B. [YMB02, Bha05, SCKT06], verwiesen.

Im NoC-Entwurf können Erfahrungen aus dem Internet und makroskopischen Datennetzen sowie aus verteilten Systemen adaptiert werden [SSM+01, Art05]. In einem verteilten System kommunizieren auf vernetzten Rechnern befindliche Komponenten durch den Austausch von Nachrichten und greifen gemeinsam auf Ressourcen zu [CDK05]. In modernen SoCs ist dies ebenso. In [HKHT05, HKM+06] zeigen Hecht et al. die Anwendbarkeit von Mechanismen verteil-

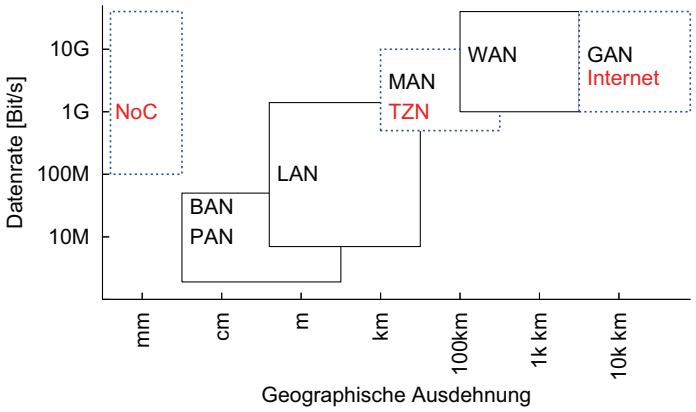


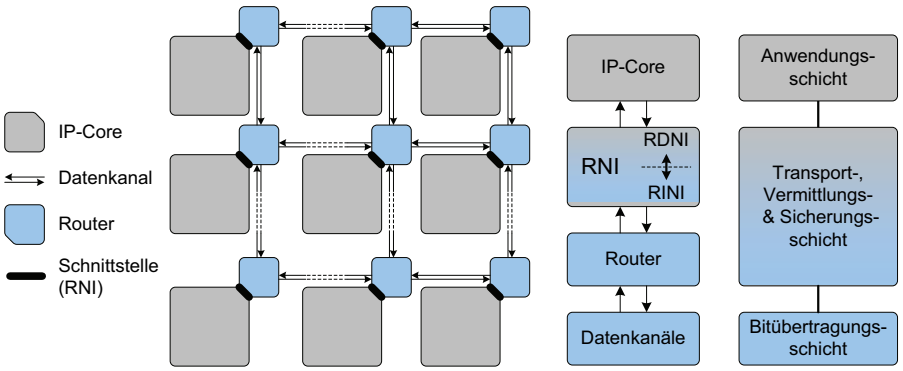
Abbildung 6.2.: Klassifizierung von Networks-on-Chip und typischen Netzwerkklassen

ter Systeme in NoC-basierten dynamisch rekonfigurierbaren Systemen. Ein NoC-basiertes SoC ist somit ein *Mikronetzwerk* von Komponenten. Während klassische Datennetze auf allgemeinen Standards basieren, um verschiedensten Applikationen gleichzeitig zu genügen, sind NoCs insbesondere in ASICs auf eine bestimmte Anwendung bzw. Anwendungsdomäne spezialisiert, bieten jedoch standardisierte Schnittstellen zu den anwendungsorientierten Schichten [JW05]. Aufgaben und Funktionen eines NoC sind ähnlich dem OSI Referenzmodell in Schichten eingeteilt (siehe Abbildung 6.3b). Dadurch wird in SoCs eine strikte Trennung von Datentransport und Datenverarbeitung erreicht – eine Trennung von inter- und intra-IP-Core-Kommunikation.

Grundbausteine eines Network-on-Chip NoCs bestehen aus Routern, Kanälen und Schnittstellen. Abbildung 6.3a zeigt die Anordnung dieser Komponenten anhand einer Gittertopologie und verdeutlicht, dass ein NoC eine verteilte Kommunikationsressource ist.

Router sind die Knotenpunkte im NoC. Zu ihren Aufgaben zählen Wegewahl und Datenflusskontrolle. Der Grad eines Routers ist gleich der Anzahl seiner Ports. Trifft ein neues Paket ein, wird es entweder zum lokalen IP-Core oder an einen entsprechenden Ausgangsport weitergeleitet. Abhängig vom gewählten Flusskontrollverfahren wird ein komplettes Paket oder werden Teile eines Pakets lokal zwischengespeichert, sollte das Paket aufgrund von Blockierungen noch nicht weitergeleitet werden können. Sobald ein Router bereit ist, ein neues Paket entgegenzunehmen, wird dies der Datenquelle signalisiert. Innerhalb des NoC werden i. Allg. jedoch keine Pakete verworfen, da die Nutzung von Fehlerprotokollen oder Sendewiederholungen in einem NoC zu aufwendig und zeitkritisch ist [1]. Dies ist ein wesentlicher Unterschied zu klassischen makroskopischen Datennetzen. NoCs werden deshalb nicht anhand ihrer Verlustrate bewertet,

[1] Zur Auflösung von Deadlocks z. B. können Pakete aber auch innerhalb des NoC verworfen werden.



(a) NoC-Komponenten in einer 2D-Gitterstruktur **(b)** Anbindung eines IP-Cores an das NoC

Abbildung 6.3.: Grundbausteine eines Network-on-Chip

sondern anhand ihrer Sättigungsgrenze bzw. Kapazität.

Router sind über *Kanäle* verbunden. Ein Datenkanal besteht aus zwei unidirektionalen Punkt-zu-Punkt-Verbindungen mit eigener Flusskontrolle. Parallele Kanäle ermöglichen die nebenläufige Übertragung mehrerer Pakete im NoC. Die Gesamtbandbreite eines NoC BW_{NoC} ist die Summe der Kanalbandbreiten aller K unidirektionalen Kanäle nach (6.5). Für ein beliebiges zweidimensionales Gitter (Mesh) mit den Kantenlängen k_x und k_y ist K durch (6.6) bestimmt.

Die *Schnittstellen* (Resource-Network-Interface, RNI) verbinden die IP-Cores mit der NoC-Infrastruktur. Abbildung 6.3b skizziert die weitere Unterteilung eines RNIs in RINI (Resource Independent Network Interface) und RDNI (Resource Dependent Network Interface). Das RINI stellt standardisierte Schnittstellen zum NoC-Router bereit. Das RDNI realisiert die Anbindung an einen speziellen IP-Core. Das RNI übt somit eine Brückenfunktion aus und abstrahiert die zugrunde liegenden kommunikationsorientierten Schichten.

$$BW_{NoC} = \sum_{i=1}^{K} BW_{Channel_i} \tag{6.5}$$

$$K_{Mesh} = 2 \cdot k_x \cdot (k_y - 1) + 2 \cdot k_y \cdot (k_x - 1) \tag{6.6}$$

Nachrichtenformat Pakete sind in mehrere sogenannte Flits (Flow Control Digits) unterteilt. Flits sind die kleinsten Informationseinheiten im NoC, auf denen Flusskontrolle betrieben wird. Ein Flit kann zudem in Phits (Physical Transfer Digits) unterteilt sein. Abbildung 6.4 stellt dies dar. Im Normalfall entspricht ein Flit einem Phit, wodurch sich die Anzahl der Flits n_{flits} eines Pakets

nach (6.7) ergibt. L_{bit} ist die Länge eines Pakets und W_{bit} die Breite des Übertragungskanals in Bit. Das Header-Flit beinhaltet Routing- und Steuerinformationen. Daten-Flits folgen dem Header und stellen die Nutzlast dar. Den Nutzdaten kann zusätzlich ein Tail-Flit mit weiteren Kontroll- und Steuerinformationen folgen.

$$n_{flits} = \left\lceil \frac{L_{bit}}{W_{bit}} \right\rceil \tag{6.7}$$

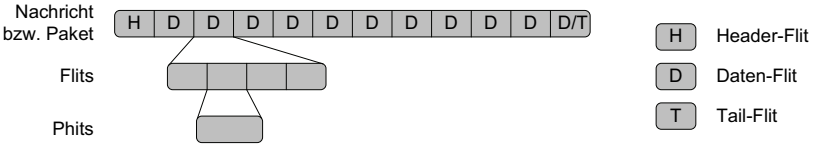


Abbildung 6.4.: Unterteilung von Paketen in Flits und Phits

NoC Design Space Während des NoC-Entwurfs müssen vielschichtige Entscheidungen getroffen werden, da verschiedene Einsatzgebiete unterschiedliche Anforderungen besitzen. Dies spannt einen weiten Parameterraum auf. Abbildung 6.5 zeigt wichtige Aspekte, welche die Komplexität und Charakteristika eines NoC entscheidend beeinflussen.

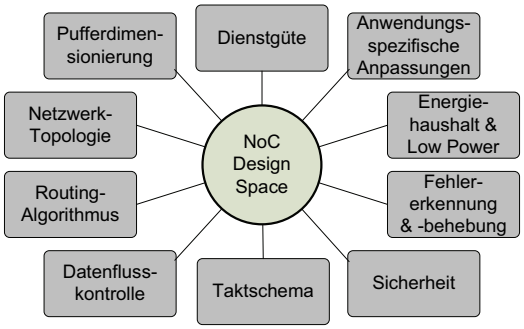


Abbildung 6.5.: Design-Space von Networks-on-Chip

Im NoC-Bereich werden vorwiegend zweidimensionale Gittertopologien eingesetzt (siehe z. B. Abbildung 6.3a), da diese relativ einfach auf die zweidimensionale Oberfläche eines ICs abgebildet werden können. Die Vielfalt existenter Routing-Algorithmen kann prinzipiell in adaptive, deterministische, minimale und nicht-minimale Algorithmen unterteilt werden. Aufgrund ihrer Einfachheit werden häufig deterministische Ansätze genutzt. Die Beleg- und Diplomarbeiten [Sof07] und [Kub04] geben einen Überblick über Routing-Algorithmen. Typische

Flusskontrollverfahren in einem NoC sind Wormhole Switching (WHS) [DS86, NM93] und Virtual Cut-Through Switching (VCTS) [KK79]. QoS-Mechanismen zur Verbesserung bzw. zur Garantie von Dienstgüten beziehen sich vor allem auf die Minimierung der Verzögerung durch das NoC, da SoCs zunehmend kommunikationsorientiert sind. Prinzipiell wird angestrebt, die durchschnittliche Latenz zu verbessern oder bestimmte Verkehrsklassen zu priorisieren.

Weitere Informationen zum Design Space sowie eine allgemeine Übersicht zu NoCs und den vielschichtigen Facetten dieses Themas bieten [JT03], [BM06a], [BM06b] und [Dem07]. Verschiedene kommerzielle und universitäre NoC-Realisierungen werden in [Ern06] analysiert und miteinander verglichen. Aktuelle Arbeiten sind in [KP07], [BKP] und [NOC08] dokumentiert.

6.1.3. GALS – Global Asynchron Lokal Synchron

Mit zunehmender Integrationsdichte zählt die Realisierung eines *synchronen* globalen Taktnetzes zu den Kernproblemen im Schaltungsentwurf. Einerseits steigt t_D, da Taktleitungen globale Leitungen sind und aufgrund des steigenden Entwurfaufwands die maximal erreichbare, synchrone Taktfrequenz limitieren. Andererseits steigt P_V aufgrund der hohen Leitungskapazitäten, die in jedem Taktzyklus umgeladen werden müssen. Als logische Weiterentwicklung komplett synchroner Systeme stellen global asynchrone lokal synchrone (GALS) Architekturen eine Möglichkeit dar, komplexe integrierte Systeme effizient zu realisieren. GALS-Architekturen nutzen dazu verschiedene und voneinander unabhängige Taktsignale, um die Komponenten eines SoCs separat anzusteuern. Ein SoC auf GALS-Basis ist *polychron* und besteht aus sogenannten Taktinseln bzw. Clock Domains. Ähnlich wie bei NoCs werden globale Leitungen aufgesplittet, wodurch t_D und P_V aufgrund der reduzierten Leitungslänge l skalieren (siehe Formel (6.3) und (6.4)) und parasitäre Einflüsse auf die einzelnen Taktinseln begrenzt werden. Der globale Taktversatz (Skew) kann vernachlässigt werden. Während des Entwurfs von GALS-Systemen erfolgt für gewöhnlich eine Partitionierung der jeweiligen Anwendung in unabhängige Teilaufgaben, welche als nebenläufige Blöcke in ein SoC integriert werden. Jeder Block kann mit einer angepassten Taktfrequenz betrieben werden, um einerseits Engpassfunktionen zu beschleunigen und andererseits Leerlaufzeiten zu reduzieren. Dies wirkt sich wiederum vorteilhaft auf Leistungsfähigkeit, Ressourcenauslastung und Energiehaushalt eines SoCs aus.

Abbildung 6.6 zeigt typische Formen der zeitlichen Beziehung zwischen zwei Taktsignalen in einem GALS-System [TGL07, DP98]. clk_1 und clk_2 sind zueinander ...

- ... synchron, wenn für ihre Frequenzen und Phasen gilt $f_1 \equiv f_2$ und $\Delta_\varphi = 0$.
- ... mesochron, wenn gilt $f_1 \equiv f_2$, $\varphi_1 \neq \varphi_2$ aber $\Delta_\varphi = \text{const}$ (und unbekannt).
- ... plesiochron, wenn gilt $f_1 \approx f_2$, $\varphi_1 \neq \varphi_2$ sowie $\Delta_\varphi \neq \text{const}$.

- ...heterochron, wenn gilt $f_1 \neq f_2$, $\varphi_1 \neq \varphi_2$ und $\Delta_\varphi \neq$ const.

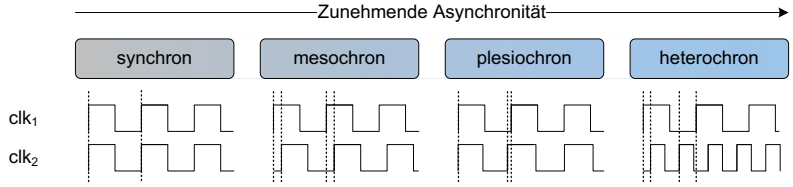


Abbildung 6.6.: Klassifizierung verschiedener Taktschemata

Die Kommunikation zwischen verschiedenen Taktregionen erfordert jedoch Maßnahmen zur Synchronisation von Daten- und Steuersignalen. Das Abgreifen eines asynchronen Signals führt ohne weitere Maßnahmen zum Verlust der Datenintegrität aufgrund metastabiler Signalpegel [Ste03]. Um GALS-Systeme trotzdem sicher verwenden zu können, existieren für verschiedene Technologien und Anwendungsgebiete Schaltungen für den Übergang von Signalen zwischen verschiedenen Taktdomänen, dem sogenannten Clock Domain Crossing (CDC). CDC-Schaltungen takten asynchrone Signale in eine Taktinsel ein [KGGV07, Cad04]. Die Güte einer CDC-Schaltung wird dabei durch die durchschnittliche Zeitspanne bis zum Auftreten eines metastabilen Zustandes (Mean Time Between Failures, MTBF) definiert [Has97, Wak00]. Wesentlichen Einfluss auf die MTBF haben die physikalischen Eigenschaften und das Zeitverhalten, u. a. die Setup- und Holdtime t_{su} und t_h, der genutzten Register-Elemente. t_{su} und t_h sind mittlerweile durch die Fortschritte im Bereich der Fertigungsprozesse und Technologien so gering, dass bereits einfache CDC-Schaltungen eine MTBF im Bereich von mehr als 10^6 Jahren erreichen können [Alt99, Gin03, Alf05]. Da Synchronisationsfehler somit zwar selten aber dennoch unvermeidbar sind, erscheint für bestimmte Applikationen zusätzlich eine Integritätsprüfung sinnvoll, z. B. auf algorithmischer Ebene per CRC.

6.1.4. Zwischenfazit

Klassische Kommunikationsinfrastrukturen zeigen aufgrund der anhaltenden Skalierung der Strukturgrößen und zunehmenden Komplexität von ICs kritische technologische und ökonomische Schwachstellen. Aktuelle Ansätze wie NoC und GALS limitieren die Auswirkungen parasitärer physikalischer Einflüsse auf lokale Regionen. Beide Ansätze sind flexibel skalierbar und erzwingen eine modulare Systemarchitektur, wodurch die Abstraktion und Nachnutzung existenter Komponenten gefördert wird. Dies wirkt sich wiederum vorteilhaft auf ökonomische Aspekte des SoC-Entwurfs aus. Prinzipiell sind NoC und GALS zwei voneinander unabhängige Ansätze, jedoch ergänzen sie sich auf einfache Art und Weise sowohl auf funktionaler als auch

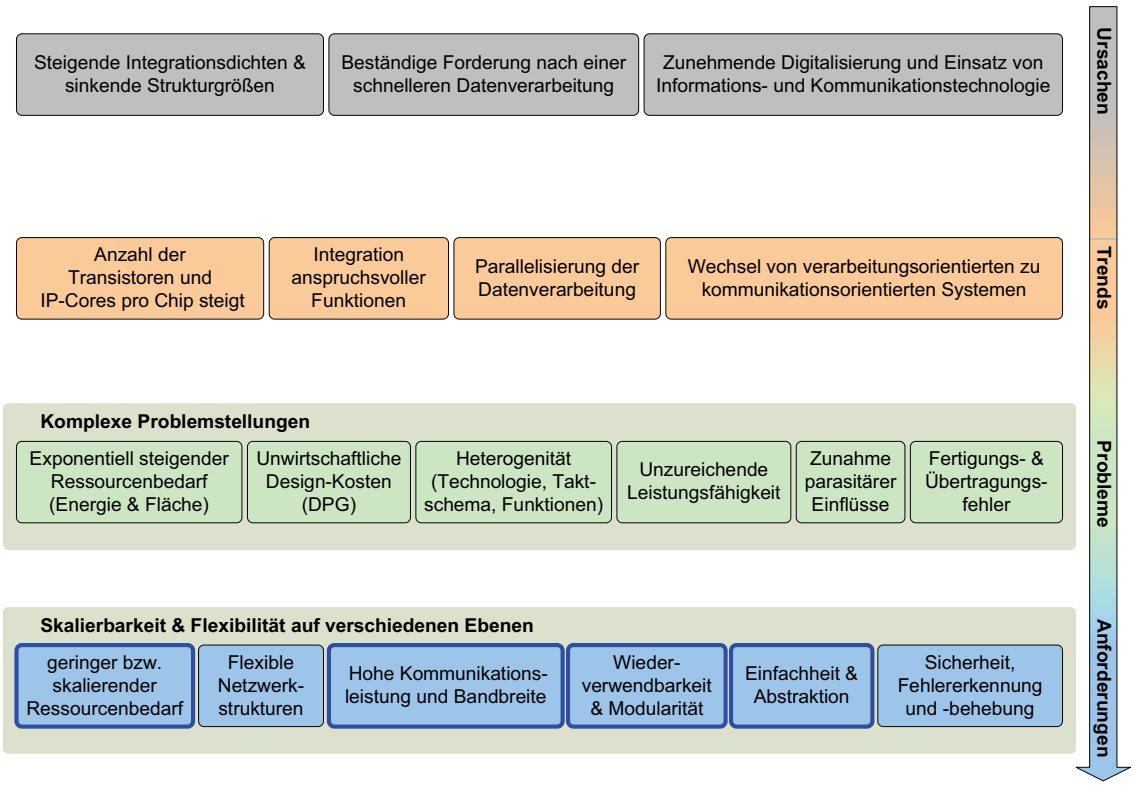


Abbildung 6.7.: Ableitung von Anforderungen an aktuelle und zukünftige Kommunikationsinfrastrukturen aus Trends und Problemen im SoC-Entwurf

auf technologischer Ebene. NoCs realisieren eine paketorientierte Kommunikation zwischen den IP-Cores. Der GALS-Ansatz setzt ein asynchrones Taktschema um. Eine Kombination beider Ansätze ist deshalb erstrebenswert.

Abbildung 6.7 gibt einen komprimierten Überblick über die genannten Ursachen, Trends und Probleme im Bereich Chip-interner Kommunikationsinfrastrukturen. Anforderungen an aktuelle und zukünftige Ansätze werden abgeleitet. In den folgenden Abschnitten diskutierte eigene Lösungsansätze behandeln insbesondere die dick umrandet hervorgehobenen Anforderungen.

6.2. Entwicklung einer schlanken und flexiblen Network-on-Chip-Architektur

Abgrenzung des Anwendungsbereichs Es gibt keine NoC-Implementierung, welche für jeden Anwendungsbereich optimale Leistungsparameter aufweist. In der Literatur werden deshalb

typische Einsatzgebiete für SoCs und damit auch für NoCs unterschieden [CK05, BM06a]:

MPSoCs Multi Prozessor SoCs (MPSoCs) bestehen aus einer Vielzahl homogener Prozessorblöcke, z. B. Intel's 80-Prozessor-Chip [VHR+07]. Durch ihre räumliche Dichte, rechenintensive parallele Applikationen und nicht vorhersagbare Verkehrsmuster erfordern MPSoCs höchste Kommunikationsleistungen und stellen besondere Ansprüche an die Dienstgüte.

Anwendungsspezifische SoCs Anwendungsspezifische SoCs sind auf dedizierte Applikationen zugeschnittene ASICs. A-priori-Wissen über das zu erwartende Kommunikationsmuster einer Anwendung fließt in den Entwurf des Systems und somit in die Gestaltung der NoC-Infrastruktur mit ein, um den Anforderungen dieser Anwendung optimal zu genügen.

Plattform-SoCs Plattform-SoCs sind hingegen auf eine bestimmte Anwendungsdomäne ausgerichtet. Sie sind typischerweise auf Softwareebene programmierbar und stellen gerade auf den unteren Schichten allgemeine abstrakte Kommunikationsmechanismen bereit.

FPGAs FPGAs sind SoCs, deren Funktion erst nach der Fertigung durch Konfiguration und Verbindung elementarer Komponenten und komplexer Makroblöcke bestimmt wird [BL03, Max04]. FPGAs bieten höchste Flexibilität für verschiedenste Anwendungen. Bisher wurden NoCs aufgrund physikalischer und elektrischer Skalierungsprobleme primär in ASICs als fest verdrahtete Strukturen genutzt. Bekannte NoC-Architekturen aus dem ASIC-Bereich sind z. B. Xpipes [BB04], Æthereal [GDR05], SPIN [ACG+03, AG03], Nostrum [MNT+04] oder SoCBus [WL03, Wik05]. Im FPGA stehen jedoch vor allem Modularität, Effizienz und Kommunikationsleistung im Vordergrund, da die elektrischen und physikalischen Eigenschaften eines FPGAs bereits durch den jeweiligen Hersteller abgestimmt worden sind [BL03]. Das zurzeit größte FPGA, das Xilinx Virtex-5 XC5VLX330T [XV5], bietet zudem mit u. a. über 300.000 Logikblöcken, 15 MByte internem Speicher und hochentwickelten Makroblöcken ausreichend Ressourcen, um hunderte IP-Cores zu integrieren. Nach Benini et al. [BM06a] und Ehliar et al. [EEL07] wird die Kommunikation in FPGA-basierten SoCs in Zukunft angesichts der zunehmenden Komplexität von FPGAs ebenfalls durch NoCs realisiert.

Der Anwendungsbereich der im Folgenden vorgestellten NoC-Architektur bezieht sich auf dedizierte Aufgaben im Bereich der Paketverarbeitung in Netzwerken und des Netzwerkmanagements bei höchsten Datenraten, wie sie in Teil II dieser Arbeit beschrieben sind. Der Telekommunikationssektor ist für gewöhnlich von kurzfristigen Veränderungen bzgl. der Protokolle, Technologien und Anwendungen geprägt, so dass Flexibilität zwingend erforderlich ist. Aufgrund ihrer funktionalen Anpassungsfähigkeit durch Rekonfiguration werden FPGAs gerade

in diesem hochdynamischen Bereich als Plattform eingesetzt. Darüber hinaus bieten FPGAs mittlerweile ausreichend Leistungspotential, welches sich aufgrund der Integration fest verdrahteter Makroblöcke der Performanz dedizierter SoCs und ASICs annähert [MSCL06, KR07]. Das Ziel dieses Abschnitts ist deshalb die Entwicklung einer adäquaten NoC-Infrastruktur, welche als logisches Overlay-Netzwerk in einem FPGA der SoC-internen Kommunikation dient.

6.2.1. Simplizitätsprinzip – Internet vs. Network-on-Chip

Sowohl der Anwendungsbereich der Paketverarbeitung als auch die Nutzung von FPGAs als Zielplattform stellen bestimmte Anforderungen an eine Kommunikationsinfrastruktur.

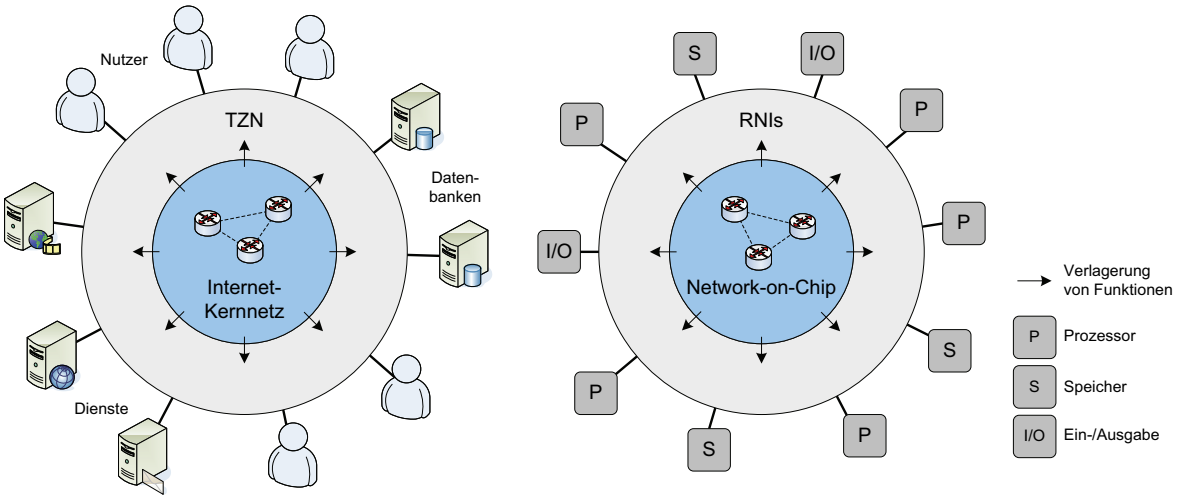
- Eine *hohe Kommunikationsleistung* in Form von Bandbreite ist erforderlich, um Datenraten von mehreren Gbit/s auch innerhalb des SoCs bewältigen zu können.
- Die Nutzung einer NoC-Infrastruktur resultiert in jedem Fall in einem nicht vernachlässigbaren zusätzlichen Hardwarebedarf. Das NoC muss deshalb *einfach* und *ressourcensparend* ausgelegt sein, um den Großteil der limitierten Ressourcen eines FPGAs der eigentlichen Anwendung vorzubehalten.
- Das NoC muss zudem *allgemein* ausgelegt und *wiederverwendbar* sein, um die Entwicklungszeit von SoCs zu verkürzen und Modifikationen bestehender SoCs effizient durchführen zu können.

Um eine hohe Kommunikationsleistung, eine schlanke Implementierung und Wiederverwendbarkeit zu erreichen, ist die Integration komplexer Mechanismen und Funktionen sowie anwendungsspezifischer Anpassungen in den transportorientierten SoC-Schichten, dem NoC, zu vermeiden. Deswegen ist nach Auffassung des Autors dieser Arbeit der Entwurf der hier beschriebenen Klasse von NoCs durch das Simplizitätsprinzip (KISS – „Keep it simple and smart.“) getrieben, welches z. B. auch dem Internet als Beispiel eines makroskopischen Netzwerks zugrunde liegt. Dieser Standpunkt ist sowohl durch Gemeinsamkeiten als auch durch Unterschiede zwischen Internet und NoCs motiviert und begründet [KCHT07]. Isenberg [Ise98] und Saltzer et al. [SRC84, RSC98] fassen o. g. Prinzip als sogenannte Ende-zu-Ende-Argumente zusammen. Gilder sagt [Gil92]: „In a world of dumb terminals and telephones, networks had to be smart. But in a world of smart terminals, networks have to be dumb.“ Es stellt sich somit die Frage, ob Langzeiterfahrungen, die während der Entwicklung und der Ausbreitung des Internets gemacht wurden, ebenso auf NoCs angewandt werden können – sind NoCs und das Internet miteinander vergleichbar?

In Abbildung 6.8 sind aus diesem Grund die generalisierten Strukturen des Internets und eines NoC-Systems gegenübergestellt. Einerseits teilen NoCs und das Internet viele grundsätzliche Gemeinsamkeiten. Beide Netzwerktypen sind dem ISO/OSI-Modell entsprechend entworfen und definieren ähnlich abstrakte Schichten. Verarbeitung und Kommunikation sind separiert. Der Informationsaustausch erfolgt über ein verteiltes Router-Netzwerk. Die Anbindung von Nutzern bzw. IP-Cores erfolgt über die TZN bzw. die RNIs, welche jeweils die intelligente und individuelle Schnittstelle zum Router-Netzwerk repräsentieren. Deshalb sind die im Umgang mit dem Internet gesammelten Langzeiterfahrungen ebenfalls für NoCs anwendbar, selbst wenn Internet und NoCs verschiedene Größenordnungen besitzen. Andererseits existieren offenkundige Unterschiede zwischen beiden Arten von Netzwerken. Das Internet besitzt viel mehr Nutzer und eine komplexere Struktur als ein NoC, welches relativ wenige IP-Cores verbindet und „nur“ aus Routern und Leitungen besteht. Aber es gibt noch einen weiteren fundamentalen Unterschied zwischen der Kommunikation in einem on-Chip Netzwerk und im Internet. Das Internet und die zugehörige Infrastruktur sind meistens nicht überlastet bzw. verstopft[2] und verfolgen eher den „Fat Pipe“-Ansatz – der DE-CIX Knoten ist z. B. nur zur Hälfte ausgelastet [DCX]. Im Gegensatz dazu wurden bisher und werden Chip-interne Kommunikationsinfrastrukturen oft an der Grenze ihrer maximalen Kapazität betrieben [OHM05]. Weitere Unterschiede zeigen sich in den treibenden Kostenfaktoren (finanzieller Gewinn vs. Entwurfskosten und Leistungsoptimierung), den zu erwartenden Arbeitslasten (heterogene Dienste vs. begrenzter Anwendungsbereich) und den physikalischen Implementierungsaspekten (off-Chip vs. on-Chip).

Bezogen auf o. g. Fragestellung können interessante Beobachtungen gemacht werden. Langjährige Erfahrungen im Internet haben gezeigt, dass Komplexitätsprobleme *nicht* mit zusätzlicher Komplexität gelöst werden können. ATM z. B. ist sehr effizient und bietet umfangreiche Möglichkeiten für QoS auf verschiedenen Ebenen. Jedoch ist es gerade deswegen auch sehr umfassend und teuer bezogen auf Konfiguration und Management. Augenscheinlich verschwindet es momentan vom Markt, da einfachere Technologien, z. B. Ethernet (siehe Abschnitt 2.4), bevorzugt werden. Ähnlich verhält es sich mit der komplexen Sicherheitsarchitektur von IPsec [RFC4301]. Die vielen Modi und Parameter führen oftmals zu Fehlkonfigurationen der jeweiligen Endpunkte. Zudem erzeugt IPsec bereits auf den unteren Protokollschichten einen hohen Overhead [Bei06]. Gerade die grundlegenden Schichten müssen jedoch einfach, allgemein und kompakt spezifiziert werden. VPN ist zurzeit die einzige wirklich sichtbare Anwendung für IPsec. Einfachere Alternativen wie TSL/SSL oder PGP haben sich hingegen in den anwendungsbezogenen Schichten etabliert. Ein anderes Beispiel ist IPv4/v6, welches QoS-Funktionen in Form eines Type-of-Service-Feldes (IPv4) bzw. eines Flow Labels (IPv6) anbietet. Jedoch sind beide Möglich-

[2] Überlastsituationen treten vorrangig zu den Hauptverkehrszeiten, z. B. in den Abendstunden auf.



(a) vereinfachte Internet-Struktur **(b)** abstraktes NoC-System

Abbildung 6.8.: Vergleich der vereinfachten Strukturen des Internets und eines NoC-Systems

keiten bisher ungenutzt und werden ignoriert, wodurch reiner IP-Datenverkehr letztendlich auf Best-Effort-Vermittlung basiert. Ein viertes Beispiel ist die FastPath-Option bei DSL-Anschlüssen. Sie erlaubt, QoS-Mechanismen zur Fehlererkennung und -behebung auszuschalten, wodurch sich die Latenz erheblich verbessert. Die genannten Beispiele sind alle auf das Simplizitätsprinzip zurückzuführen. Es funktioniert so gut, da genug Bandbreite vorhanden ist und immer mehr Bandbreite zur Verfügung stehen wird (siehe z. B. [CBP+06] und [YYS+08]). Die Adaption dieses Prinzips auf SoC-Ebene bedeutet, dass NoCs *nicht* an ihrem Limit betrieben werden dürfen, wie es bisher bei klassischen Bussystemen der Fall war. Durch funktionale Unterspezifikation und Bereitstellung von ausreichend Bandbreite kann in den transportorientierten Schichten auf komplexe Mechanismen, z. B. für QoS und anwendungsspezifische Optimierungen, verzichtet werden.

Mechanismen zur Sicherung der Dienstgüte sowohl im Internet als auch in NoCs sind jedoch nach wie vor notwendig. Einerseits verfolgen z. B. Goossens et al. mit dem Æthereal-Projekt eine eher konträre Vorgehensweise [GMPW02, GDR05, RDP+05] und stellen garantierte Dienstgüten in den Mittelpunkt ihres NoC-Ansatzes. Sie beziehen sich jedoch mit Echtzeit- und Multimediaanwendungen in ASIC-basierten komplexen SoCs auf einen Anwendungsbereich mit gänzlich anderen Anforderungen. Andererseits ist Best-Effort in der Telekommunikation aus ökonomischer Sicht ein nicht lebensfähiges Geschäftsmodell, weswegen die Garantie abonnierter Bandbreiten ein Grund für die Existenz von QoS-Mechanismen im Internet ist. Jedoch ist auch hier am Beispiel Internet zu erkennen, wie mit diesen Problemen umgegangen werden

kann. Komplexität wird momentan aus dem Kernnetz heraus in die TZN verlagert. Diese Form der Dezentralisierung erlaubt den Einsatz einfacherer angepasster Mechanismen am Rand des Netzes, wie z. B. die Beiträge in Teil II dieser Arbeit zeigen. In den Abbildungen 6.8a und 6.8b ist diese Funktionsverlagerung anhand der Pfeile in Richtung TZN und RNIs verdeutlicht. Wird ein SoC nach diesem Prinzip entworfen, kann die zugrunde liegende NoC-Infrastruktur einfach und mit geringem Hardware-Overhead gestaltet werden. Eine schlanke NoC-Implementierung kann zudem mit höheren Taktraten betrieben werden und somit eine höhere physikalische Bandbreite zur Verfügung stellen.

Im Folgenden wird eine NoC-Architektur vorgestellt, die mit Blick auf die genannten Annahmen und Anforderungen auf verschiedenen Ebenen optimiert wird. Abschnitt 6.2.2 beschreibt als Ausgangsbasis die allgemeinen Eigenschaften und generellen Mechanismen der NoC-Architektur. In Abschnitt 6.3 und 6.4 werden eigens entwickelte Optimierungsansätze bzgl. der Datenflusskontrolle und Topologie vorgestellt und bewertet.

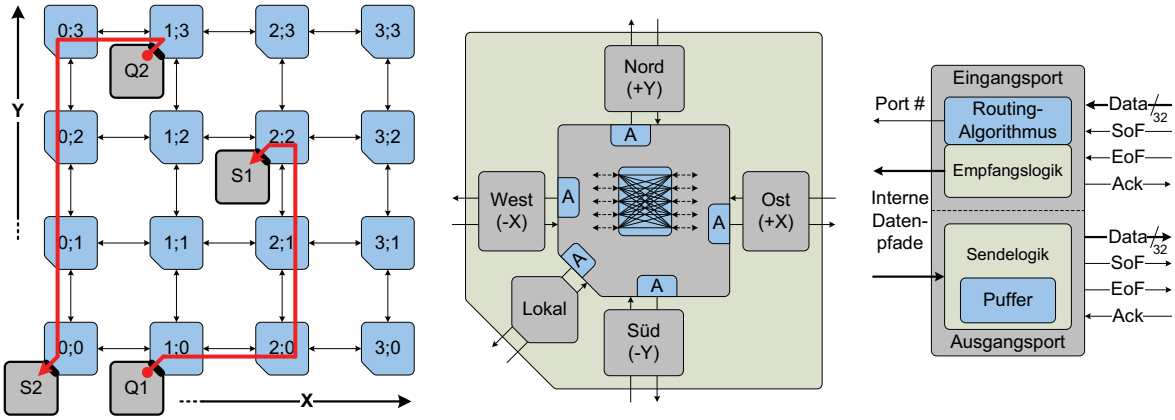
6.2.2. Grundlegende NoC-Parameter

Die primären Charakteristika eines Netzwerks sind seine Topologie, die Wegewahl (Routing) und das Flusskontrollverfahren (Switching) [DYN03, DT03]. Als Ausgangsbasis für die weiteren Betrachtungen wurde eine NoC-Infrastruktur wie folgt spezifiziert und entwickelt.

Topologie Das NoC basiert auf einem k-fachen n-Würfel. n definiert die Dimensionen und k die Kantenlänge des Würfels, wobei rechteckige Topologien mit $k_x \neq k_y$ nicht ausgeschlossen sind. Mit $n = 2$ werden zweidimensionale Gittertopologien beschrieben. Abbildung 6.9a zeigt einen 4-fachen 2-Würfel (4×4-Gitter). Diese auch als Manhattan-Style bezeichnete Topologie bietet eine hohe Regularität, weist sehr gute Skalierungseigenschaften bzgl. des Ressourcenverbrauchs (Fläche & Leitungslänge) auf und kann effizient auf ein FPGA bzw. einen ASIC abgebildet werden. Die maximale Distanz zweier IP-Cores, der Durchmesser d_{max} des Gitters, ergibt sich nach Formel (6.8). d_{max} ist dabei in Anzahl der Router auf diesem Pfad definiert. Das flache Adressierungsschema ordnet jedem Router und IP-Core ein Koordinatenpaar (x;y) zu.

$$d_{max} = 2 \cdot k - 1 \quad \text{bzw.} \quad = k_x + k_y - 1 \tag{6.8}$$

Wegewahl Als Routing-Algorithmus wird Dimension-Ordered Routing (DOR) genutzt, welches im zweidimensionalen Fall auch mit XY-Routing bezeichnet wird. DOR-Algorithmen sind einfach zu implementieren, deterministisch, minimal und deadlockfrei. Deterministisch und



(a) XY-Routing in einem 2D-Gitter **(b)** Aufbau eines NoC-Routers **(c)** Portstruktur und Signale

Abbildung 6.9.: Aufbau eines 2D-Gitters und Routerstruktur

minimal bedeutet, dass für ein Quelle-Senke-Paar immer dieselbe Route gewählt wird, die gleichzeitig den kürzesten Weg zur Senke darstellt. d ist dabei die Länge des Übertragungspfades als Anzahl der Router (Hops) auf diesem Übertragungspfad. Aufgrund der Deterministik können sich Pakete im NoC nicht überholen, weswegen auf Sortiermechanismen an der Senke verzichtet werden kann. Deadlocks werden bei DOR-Algorithmen aufgrund der strikten Reihenfolge der Dimensionen, in der sich ein Paket durch das NoC bewegt, vermieden, da keine zirkulären Abhängigkeiten im NoC entstehen können. Mechanismen zur Vermeidung, Erkennung und Auflösung von Deadlocks sind nicht nötig[3]. In Abbildung 6.9a ist an zwei Beispielen der Weg eines Pakets von $Q1$ bzw. $Q2$ zu $S1$ bzw. $S2$ gezeigt. Zuerst wird in der X-Dimension geroutet, dann in der Y-Dimension.

Router-Struktur Der prinzipielle Aufbau eines NoC-Routers wird z. B. in [Chi98] beschrieben. Davon abgeleitet ist in Abbildung 6.9b die Struktur eines Routers des entwickelten NoC skizziert. Die zentralen Funktionen des Routers sind Wegewahl (XY-Routing) und Arbitrierung und damit die Konfiguration der Crossbar. Die Arbitrierung (A) erfolgt pro Ausgangsport auf Basis eines Round-Robin-Schedulers. Die Crossbar erlaubt eine Verbindung von jedem Eingang zu jedem Ausgang und ist aus Multiplexern aufgebaut. Jeder Router hat 5 Ports, deren Bezeichnungen im zweidimensionalen Gitter an die Himmelsrichtungen angelehnt sind. Der lokale Port verbindet ein RNI und IP-Core mit dem Router. Abbildung 6.9c zeigt die Detailansicht eines Ports und die Daten- und Steuersignale der Kanäle. Der Datenpfad ($Data$) ist 32 Bit breit. Start-of-Frame (SoF)

[3] An dieser Stelle wird von einer intakten und fehlerfreien NoC-Infrastruktur ausgegangen

Tabelle 6.1.: Gegenüberstellung der Eigenschaften von VCTS, GWHS und KWHS

	VCTS	GWHS	KWHS
Ressourcenbedarf	große Paket-Puffer	kleine Flit-Puffer	kleine Flit-Puffer
Arbeitstakt f_{NoC}?	$\approx f_{Router}$, const	$\approx f_{Router}$, const	$= f(k_x, k_y)$
Latenz t	$= f(n_{flits})$	$= f(2 \cdot n_{flits})$	$= f(n_{flits})$
idealer Durchsatz	$\approx BW_{NoC}$	$\approx BW_{NoC}/2$	$\approx BW_{NoC}$

und End-of-Frame (EoF) signalisieren das erste und letzte Flit eines Pakets. Ack dient als rückwärtiges Bestätigungssignal. Die Datenflusskontrolle wird in einem Port durch die Sende- und Empfangslogik realisiert. Letztere beinhaltet auch den Routing-Algorithmus. Output-Buffering wurde gewählt, um die Auswirkungen des sogenannten Head-of-Line-Blockings (HOL) zu reduzieren [MR07]. Dazu sind abhängig vom Flusskontrollverfahren FIFO-Speicher oder einfache Register als lokale Puffer für Pakete bzw. Flits in den Ausgangsports integriert. Die interne Verzögerung eines NoC-Routers setzt sich aus dem zeitlichen Aufwand für die Wegewahl, die Arbitrierung sowie das Zuweisen von Daten an die Schnittstellen der Ausgangsports zusammen und ist durch n_{route} in Anzahl der Taktzyklen beschrieben.

Datenflusskontrolle In der Grundversion des NoC wurden für einen späteren Vergleich verschiedene klassische Flusskontrollmechanismen umgesetzt: VCTS, getaktetes WHS (GWHS) und kombinatorisches WHS (KWHS). Tabelle 6.1 stellt die wichtigsten Eigenschaften dieser Flusskontrollverfahren gegenüber. Nachteilige Eigenschaften sind hervorgehoben.

Bei VCTS wird Flusskontrolle auf Ebene eines *kompletten* Pakets durchgeführt. Ein NoC-Router muss pro Port mindestens Speicherplatz in Höhe der MTU bereitstellen. Einzelne Flits eines Pakets können jedoch sofort weitergeleitet werden, ohne dass wie bei Store-and-Forward-Switching (SAFS) vorher das komplette Paket empfangen werden muss. Dies führt zu einer geringeren Latenz als bei SAFS. Kann das Paket aufgrund von Blockierungen nicht weitergeleitet werden, wird es komplett lokal zwischengespeichert. Zuvor belegte Netzwerkknoten werden wieder freigegeben.

Prinzipiell wird bei WHS Flusskontrolle nicht mehr pauschal auf einem kompletten Paket, sondern auf *einzelnen* Flits angewandt. Der wesentliche Vorteil ist, dass Puffer nun auf Flit-Basis reserviert werden. Dies erlaubt die Verwendung kleiner Flit-Speicher und spart somit Ressourcen. Da ein Paket im Fall einer Blockierung nicht komplett lokal zwischengespeichert werden kann, sind im ungünstigsten Fall alle bisher zugeteilten Puffer auf dem Übertragungsweg belegt, was wiederum zusätzliche Blockierungen anderer Pakete nach sich ziehen kann. Bei GWHS wird jedes Flit mit einem *synchronen* abgetakteten Signal (Ack) bestätigt. Dadurch werden für die

Übertragung eines einzelnen Flits zwei Takte benötigt, wie die Kostenfunktionen der Latenz bzw. Übertragungsdauer in Tabelle 6.1 zeigen. Einerseits bleibt die maximale Arbeitsfrequenz der NoC-Infrastruktur durch die Nutzung getakteter Registerausgänge für die Steuer-Signale stabil, wenn sich die Größe des NoC verändert. Andererseits kann jedoch nur von der Hälfte der maximalen Bandbreite des NoCs profitiert werden. Der Unterschied zwischen KWHS und GWHS ist die Verwendung eines *kombinatorischen* Ack-Signals, welches nicht durch ein Register abgetaktet ist. Der Vorteil von KWHS liegt in der besseren Ausnutzung der Bandbreite der Übertragungskanäle, da pro Takt ein Flit übertragen werden kann. Der theoretisch erzielbare Durchsatz liegt bei KWHS und VCTS ungefähr bei BW_{NoC}. Allerdings verringert sich die maximale Arbeitsfrequenz mit zunehmender Größe der NoC-Infrastruktur, da sich die kombinatorischen Pfade der Schaltung ebenfalls entsprechend verlängern.

Die Abbildungen 6.10a bis 6.10c zeigen das Signalverhalten an einer Schnittstelle für VCTS, GWHS bzw. KWHS. Anhang C.1 bietet weitere detaillierte Simulationsdiagramme des Übertragungsverhaltens der genannten Mechanismen.

Schritt (1) Ein neues Paket wird durch ein gesetztes SoF signalisiert.

Schritt (2) Der Router signalisiert die Bereitschaft zur Verarbeitung des Pakets mit $Ack = 1$.

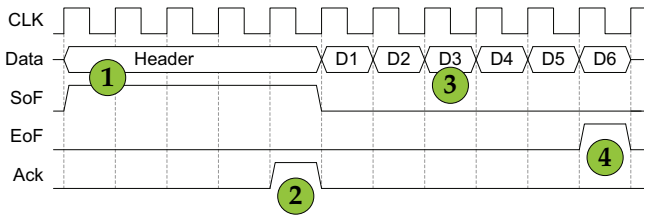
Schritt (3) Daraufhin werden die verbleibenden Daten übertragen. Bei VTCS erfolgt dies taktweise. Es ist keine weitere Bestätigung nötig, da ein Router ausreichend Speicher für ein komplettes Paket zusichert. Bei KWHS wird der Empfang eines Flits sofort mit dem kombinatorischen Ack bestätigt, um pro Takt ein Flit übertragen zu können. Bei GWHS wird jedes Flit durch ein abgetaktetes Ack bestätigt, bevor ein neues Flit an $Data$ zugewiesen werden darf. Dies erfordert mindestens zwei Takte pro Flit. Wird bei GWHS und KWHS der Kopf des Pakets blockiert, wird ein Daten-Flit erst entsprechend verzögert bestätigt, z. B. Daten-Flit $D1$.

Schritt (4) Ein gesetztes EoF signalisiert das Ende des Pakets.

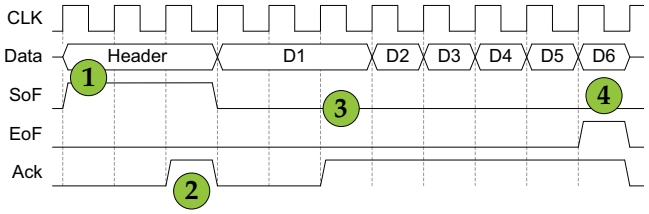
Die Latenz t einer Paketübertragung für die einzelnen Verfahren ergibt sich nach (6.9) bzw. (6.10). Die drei wesentlichen Komponenten der Latenz sind erkennbar: Verzögerungen durch Routing und Arbitrierung ($d \cdot n_{route}$), Serialisierungsverzögerung durch die Paketlänge (n_{flits}) und Wartezyklen durch Blockierungen innerhalb des NoC ($n_{blocked}$) [DT03, Aga91].

$$t_{VCTS}, t_{KWHS} = T_{clk} \cdot (d \cdot n_{route} + n_{flits} + n_{blocked}) \tag{6.9}$$

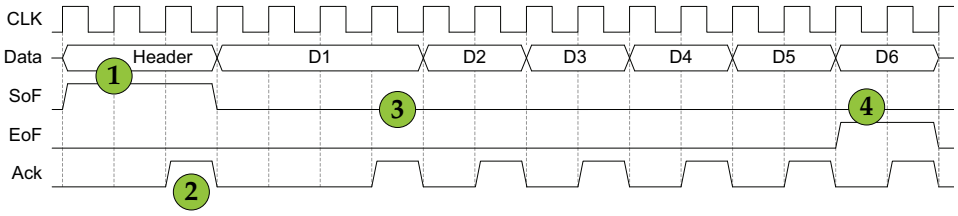
$$t_{GWHS} = T_{clk} \cdot (d \cdot n_{route} + 2 \cdot n_{flits} + n_{blocked}) \tag{6.10}$$



(a) Signalisierung an einer VCTS-Schnittstelle



(b) Signalisierung an einer KWHS-Schnittstelle



(c) Signalisierung an einer GWHS-Schnittstelle

Abbildung 6.10.: Signalisierung verschiedener Flusskontrollverfahren

Eine weitere Form der Datenflusskontrolle ist das Konzept virtueller Kanäle. Unter Virtual Channels (VCs) wird die Unterteilung physikalischer Pufferspeicher in logische Warteschlangen verstanden [Dal92]. Dadurch können blockierte Ressourcen umgangen werden, wodurch die durchschnittliche Latenz des NoC gesenkt werden kann oder bestimmte Verkehrsklassen priorisiert werden können. Zirkuläre Abhängigkeiten im NoC (Deadlocks) können aufgetrennt werden [DS87, Dua93]. Negative Auswirkungen von HOL [PGIB99, BPS99] können durch virtuelle Warteschlangen zusätzlich verringert werden. Auf die Verwendung von VCs wurde aus verschiedenen Gründen jedoch *verzichtet*. Einerseits hängt die relative Leistungssteigerung durch VCs sowohl vom tatsächlichen Verkehrsmuster der Anwendung als auch vom Routing-Algorithmus ab [RE94]. Treten keine Blockierungen auf, kann kein Leistungsgewinn erzielt werden. Andererseits resultiert der Einsatz von VCs in einem erheblichen Mehrbedarf an Logik und Speicherressourcen für die deutlich komplexeren Flusskontroll- und Arbitrierungsaufgaben. In der Studienarbeit von Heinrich [Hei06] wurde die Nutzung von VCs in dem hier vorgestellten NoC untersucht. Die zusätzliche Komplexität eines NoC-Routers senkt dessen maximalen

Arbeitstakt um bis zu 30 %, was wiederum negative Auswirkungen auf BW_{NoC} und damit den absoluten Leistungsgewinn hat [VSD01, MTCM06]. Da sich bei Nutzung von VCs Pakete auch überholen können, sind zusätzlich Mechanismen zur Sortierung der Pakete bzw. Vermeidung des Überholens nötig. Darüber hinaus ist XY-Routing bereits inhärent frei von Deadlocks.

6.3. Hybrider Switching Mechanismus

Motivation Tabelle 6.1 zeigte, das jedes der klassischen Flusskontrollverfahren einen charakteristischen Nachteil besitzt. Bei GWHS kann maximal nur die halbe Bandbreite genutzt werden. Bei KWHS ist f_{NoC} und damit BW_{NoC} eine Funktion von k_x und k_y. VCTS hingegen benötigt große Speicherblöcke, die in einem FPGA jedoch nur begrenzt vorhanden sind. Für eine schlanke und gleichzeitig leistungsstarke NoC-Infrastruktur wird jedoch ein ressourcensparendes und möglichst effizientes Flusskontrollverfahren benötigt:

- Eine hohe Kommunikationsleistung hängt dabei von f_{NoC} und der Effizienz des Flusskontrollverfahrens ab. f_{NoC} darf wiederum nur von der Komplexität eines einzelnen NoC-Routers abhängen und nicht von den Dimensionen der NoC-Infrastruktur. Deshalb wird eine mesochrone Taktstrategie gewählt, wie sie in Abschnitt 6.1.3 beschrieben wurde.
- Die größte Effizienz wird erreicht, wenn pro Takt ein Flit übertragen werden kann.
- Für die Verwendung kleiner Flit-Puffer ist eine Form von WHS notwendig.

Im Folgenden wird der sogenannte Hybride Switching Mechanismus (HSM) vorgestellt, welcher die o. g. Anforderungen erfüllt. HSM ist in der Diplomarbeit von Heinrich [Hei07] dokumentiert und in [KHT07] veröffentlicht.

6.3.1. Modifikationen für ein mesochrones Taktschema

Jeder NoC-Router wird mit einem unabhängigen Takt getrieben, welcher von einem globalen Takt abgeleitet ist. Zur Erzeugung regionaler Takte aus einem Primärtakt wird ein Digital Clock Manager[4] (DCM) genutzt. DCMs sind dedizierte Funktionsblöcke des FPGAs und bieten u. a. Funktionen zur Taktsynthese. Abbildung 6.11 skizziert die Nutzung verschiedener Takttreiber für ein 2×2-NoC. Unterschiedliche Taktdomänen sind durch gestrichelte Linien hervorgehoben. Pro Router und IP-Core werden ein separates Taktnetz und ein separater Treiberbaustein genutzt. Die Takte A_0-A_3 für die NoC-Router werden aus dem gleichen vom DCM erzeugten Takt A abgeleitet. Die Frequenzen f_{A_0}-f_{A_3} sind gleich, jedoch sind die Phasen φ_{A_0}-φ_{A_3} nicht zwingend gleich. Für

[4] Diese Arbeit bezieht sich auf FPGAs der Firma Xilinx. Detaillierte Produktinformationen sind unter [XLX] verfügbar.

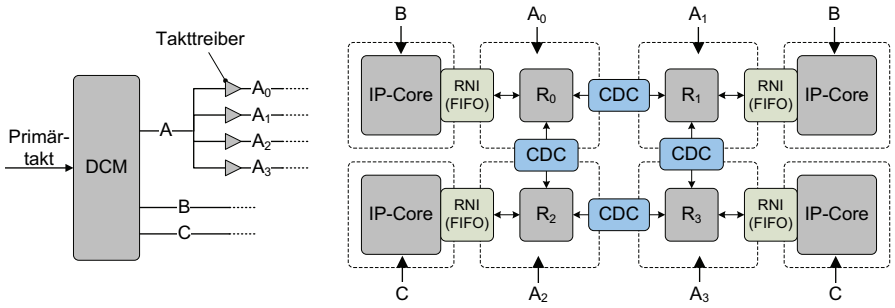
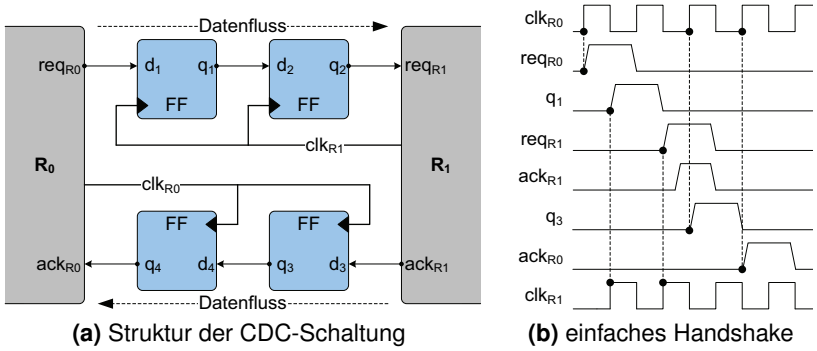


Abbildung 6.11.: Taktdomänen in einem mesochronen 2×2-NoC



(a) Struktur der CDC-Schaltung **(b)** einfaches Handshake

Abbildung 6.12.: CDC-Schaltung auf Basis eines 2-Flipflop-Synchronisierers

den Übergang von der NoC-Infrastruktur zu den IP-Cores werden in den RNIs fest verdrahtete FIFO-Primitive des FPGAs genutzt. Sie erlauben die Ansteuerung mit unterschiedlichen Lese- und Schreibtakten. Dadurch können das RDNI und der IP-Core mit von Takt A abweichenden Frequenzen betrieben werden ($f_{B,C} \neq f_A$). Die NoC-Infrastruktur arbeitet mesochron. Das gesamte NoC-System ist jedoch heterochron ausgelegt.

Da jedoch eine Phasendifferenz $\Delta_\varphi \neq 0$ zwischen verschiedenen Taktinseln im NoC bestehen kann, sind zwischen den Routern CDC-Schaltungen nötig, um Signale in die jeweiligen Taktdomänen einzutakten. Die CDC-Schaltung aus Abbildung 6.12a wird zur Umsetzung einer mesochronen NoC-Infrastruktur genutzt. Die Schaltung basiert auf einem einfachen 2-Flipflop-Synchronisierer [Ste03]. Pro Signal sind zwei Flipflops in Reihe geschaltet. Es werden Registerelemente des FPGAs genutzt. Die Abbildung zeigt die CDC-Schaltung für ein einfaches Handshake-Interface zwischen den Routern R_0 und R_1. Die Register sind durch den Takt der jeweils lesenden Taktdomäne gesteuert, um das eintreffende Datensignal zu synchronisieren. Abbildung 6.12b zeigt das Zeitverhalten der Signale für ein einfaches Bestätigungsprotokoll

auf Basis von Abbildung 6.12a. R_0 muss mindestens 3 Taktzyklen warten, um auf eine Anfrage (Request) die Bestätigung (Acknowledge) von R_1 zu empfangen. Je nach Phasenlage beträgt die Synchronisationsverzögerung n_{sync} für ein einfaches Handshake 3 bis 4 Takte. Derartige Synchronisationsverzögerungen sind ein inhärenter Nachteil von CDC-Schaltungen. Die im vorherigen Abschnitt 6.2.2 vorgestellten Flusskontrollverfahren sind aufgrund dieser Verzögerung nicht mehr ohne weiteres anwendbar, da die effektive Bandbreite erheblich degradiert wird. Aus diesem Grund wurde mit HSM ein angepasstes Flusskontrollverfahren entwickelt.

6.3.2. Funktionsweise von HSM

Die Übertragung eines Pakets ist bei HSM in zwei Phasen unterteilt – *Setup* und *Fast-Transmit*. Die entsprechende Sende- und Empfangslogik ist dabei in den Ein- und Ausgangsports der Router implementiert (siehe Abbildung 6.9c). HSM basiert dabei auf den folgenden Annahmen:

- Die Wegewahl erfolgt durch einen *deterministischen* Routing-Algorithmus, z. B. DOR. In jedem Router ist so die noch verbleibende Anzahl an Routing-Schritten (Hops) bis zur Senke aus der Adresse des Routers und der Zieladresse im Header-Flit ermittelbar.

- In den Ports der Router werden kleine Flit-Puffer genutzt. Die Puffer in den RNIs hingegen sind groß genug, um ein maximal großes Paket aufnehmen zu können. Diese Bedingung ist durch die in den RNIs ohnehin vorhandenen FIFO-Puffer erfüllt.

Setup-Phase Die Bezeichnung deutet bereits auf die Funktion dieser Phase hin – den Aufbau des Übertragungspfads bis zur Senke. Jedoch unterscheidet sich HSM von leitungsvermittelten Ansätzen, da schon *während* des Aufbaus der Verbindung Daten übertragen werden. Es ist *keine* Bestätigung an die Quelle für den erfolgreichen Verbindungsaufbau notwendig. Während der Setup-Phase wird jedes Flit bestätigt, da das Header-Flit auf seinem Weg zur Senke blockiert werden kann. Zu diesem Zweck unterhält jeder Router-Port einen Zähler – Hop-Counter (HC) – welcher die bereits übermittelten Flits eines Pakets zählt. Wenn ein Header-Flit an einem Port eintrifft, wird der Zähler auf n_{hc} gesetzt. n_{hc} ist die Anzahl der Routing-Schritte bzw. Hops, die aus Sicht des aktuellen Routers bis zur Senke noch nötig sind. Während des Weiterleitens der Daten-Flits eines Pakets wird HC pro Flit um Eins dekrementiert. Aufgrund der flachen Adressierung und deterministischen Wegewahl kann n_{hc} durch Formel (6.11) ermittelt werden. Das Maximum $n_{hc,max}$ ergibt sich nach (6.12) aus d_{max} (siehe Formel (6.8)).

$$n_{hc} = |x_{router} - x_{dest}| + |y_{router} - y_{dest}| \tag{6.11}$$

$$n_{hc,max} = d_{max,Mesh} - 1 \tag{6.12}$$

Fast-Transmit-Phase Sobald HC in einem Port den Wert Null erreicht, wechselt die Portlogik in die Fast-Transmit-Phase. Von nun an werden Daten-Flits *ohne* Bestätigung vom empfangenden Router weitergeleitet. Dies ist möglich, da das Header-Flit nach Erreichen der Senke nicht mehr blockiert werden kann und der FIFO-Puffer im RNI der Senke mindestens Platz für ein komplettes Paket bietet. Um einzelne Daten-Flits während der Fast-Transmit-Phase unterscheiden zu können, wird ein zusätzliches Steuersignal – Valid Data Toggle (VDT) – pro Kanal genutzt. An der Senke identifiziert ein wechselnder Signalpegel von VDT ein neues gültiges Daten-Flit. Auf diese Weise kann in der Fast-Transmit-Phase pro Taktzyklus ein Daten-Flit übertragen werden. Dies ist dann der Fall, wenn alle Router-Ports entlang des Übertragungspfades in die Fast-Transmit-Phase gewechselt haben. Das letzte Daten-Flit eines Pakets wird durch EoF gekennzeichnet und gibt die belegten Router-Ports wieder frei. Die Fast-Transmit-Phase ist aus Sicht der Router-Ports nur ein einfaches Weiterleiten von Daten. Die Intelligenz befindet sich am Rand des NoC im RNI der Quelle und Senke (siehe Abbildung 6.8 ⇒ Verlagerung von Funktionen).

Die Latenz wird bei HSM durch (6.13) angenähert. Die einzelnen Komponenten der Latenz für die Übertragung eines Pakets sind hervorgehoben. Da die genauen Phasenverschiebungen Δ_φ zwischen den einzelnen Taktdomänen und damit die Verzögerungen der CDC-Schaltungen unbekannt sind, ist eine exakte Berechnung von t_{HSM} nicht möglich. Eine untere und obere Grenze kann aber durch Einsetzen von $n_{sync} = 3$ bzw. $n_{sync} = 4$ in Gleichung (6.13) ermittelt werden (vgl. Abbildung 6.12b). Für die genutzten NoC-Router gilt weiterhin $n_{route} = 3$. Damit vereinfacht sich (6.13) zu (6.14).

$$t_{HSM} = T_{clk} \cdot \Big(\overbrace{n_{route} \cdot d}^{\text{Routing}} + \underbrace{2 \cdot n_{sync} \cdot (d-1)}_{\text{Synchronisation}} + \overbrace{n_{flits}}^{\text{Serialisierung}} + \underbrace{n_{blocked}}_{\text{Blockierung}} \Big) \tag{6.13}$$

$$t_{HSM} = \begin{cases} T_{clk} \cdot \Big(11 \cdot d + n_{flits} + n_{blocked} - 8 \Big) & \text{für } n_{sync} = 4 \\ T_{clk} \cdot \Big(9 \cdot d + n_{flits} + n_{blocked} - 6 \Big) & \text{für } n_{sync} = 3 \end{cases} \tag{6.14}$$

In Abbildung 6.13 wird die Signalisierung an einer HSM-Schnittstelle anhand der Übertragung eines Pakets zwischen den Routern R_0 und R_1 demonstriert. Es werden die gleichen Indizes wie in Abbildung 6.12 genutzt. Der obere Teil beinhaltet Signale, die zu R_0's Taktdomäne gehören.

Der untere Teil stellt Signale aus R_1's Taktdomäne dar. Der mittlere Teil zeigt die Zwischensignale innerhalb der CDC-Schaltung, welche sensitiv auf CLK_{R_0} oder CLK_{R_1} sind.

Schritte (1) – (3) R_0 hat ein neues Header-Flit verarbeitet und leitet es in Richtung R_1 weiter ($Data_{R_0}$). R_0 wechselt in die Setup-Phase und setzt HC ($Count_{R_0}$) nach (6.11) auf $N = 2$. Der Header wird über die CDC-Schaltung in die Taktdomäne von R_1 übernommen.

Schritte (4) – (6) Nachdem R_1 die Routing- und Arbitrierungsprozesse für das neue Paket abgeschlossen hat, wechselt R_1 ebenfalls in die Setup-Phase und setzt HC ($Count_{R_1}$) auf $N = 1$. R_1 setzt zudem das Bestätigungssignal Ack_{R_1}, welches in die Domäne von R_0 einsynchronisiert wird. Während der Setup-Phase ergibt sich pro Flit eine Verzögerung von mindestens 7 Taktzyklen für Routing und Arbitrierung (n_{route}) sowie Synchronisationsverzögerung (n_{sync}).

Schritt (7) R_0 empfängt die Bestätigung für das Header-Flit (Ack_{R_0}). Daraufhin wird das erste Daten-Flit in Richtung R_1 angelegt. SoF_{R_0} wird zurückgesetzt. $Count_{R_0}$ wird dekrementiert.

Schritt (8) Daten-Flit $D1$ erreicht R_1. Beide Router befinden sich in der Setup-Phase und erwarten das Ack des nächsten Hops hinter R_1 auf dem Weg zur Senke.

Schritte (9) – (11) R_1 empfängt dieses Ack und leitet es an R_0 weiter. Es wird in die Taktdomäne von R_0 einsynchronisiert ($Ack_{R_1} \Rightarrow Ack_{q3} \Rightarrow Ack_{R_0}$). Gleichzeitig wird $Count_{R_1}$ dekrementiert. Da $Count_{R_1}$ null erreicht, wechselt R_1 zu Fast-Transmit. Von nun an werden Daten-Flits ohne Bestätigung weitergeleitet. Am RNI der Datensenke werden bis zum Ende des Pakets die Pegelwechsel von VDT genutzt, um neue gültige Daten-Flits zu erkennen. VDT wird im RNI der Datenquelle generiert und innerhalb des NoC mit dem Paket übermittelt.

Schritt (12) R_0 weist das nächste Daten-Flit an $Data_{R_0}$ zu. Zudem wechselt R_0 in die Fast-Transmit-Phase, da $Count_{R_0}$ Null erreicht hat. Ist der Übertragungsweg komplett zwischen Quelle und Senke etabliert, wird letztendlich pro Takt ein Flit übertragen, z. B. die Flits $D4$-$D6$. Daten-Flit $D3$ benötigt noch 2 Takte, da der komplette Übertragungsweg erst nach Empfang des letzten Ack-Signals an der Quelle aufgebaut ist und somit noch ein zusätzlicher Wartezyklus für das Zuweisen von $D3$ an den Datenbus entsteht.

Schritte (13) – (15) EoF signalisiert das letzte Daten-Flit des Pakets und damit das Ende der Übertragung. Die Router-Ports werden wieder frei gegeben. R_0 und R_1 wechseln in den idle-Zustand und warten auf ein neues Header-Flit.

Anhang C.1 bietet darüber hinaus eine detailliertere Darstellung des Übertragungsverhaltens von HSM anhand der Simulation einer vollständigen Paketübertragung.

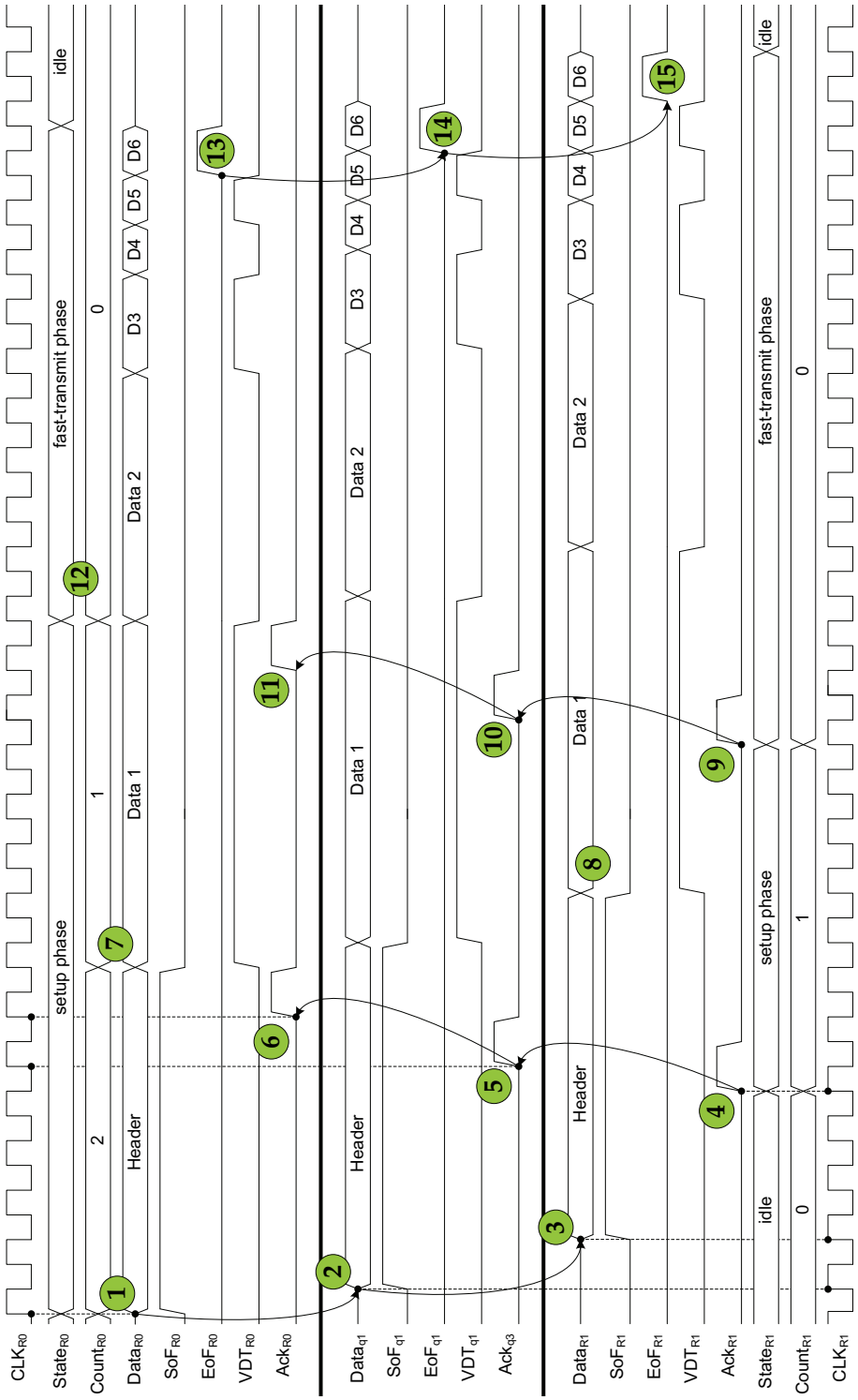


Abbildung 6.13.: Signalisierung an einer HSM-Schnittstelle am Beispiel einer Paket-Übertragung.

6.3.3. Syntheseergebnisse und Bewertung von HSM

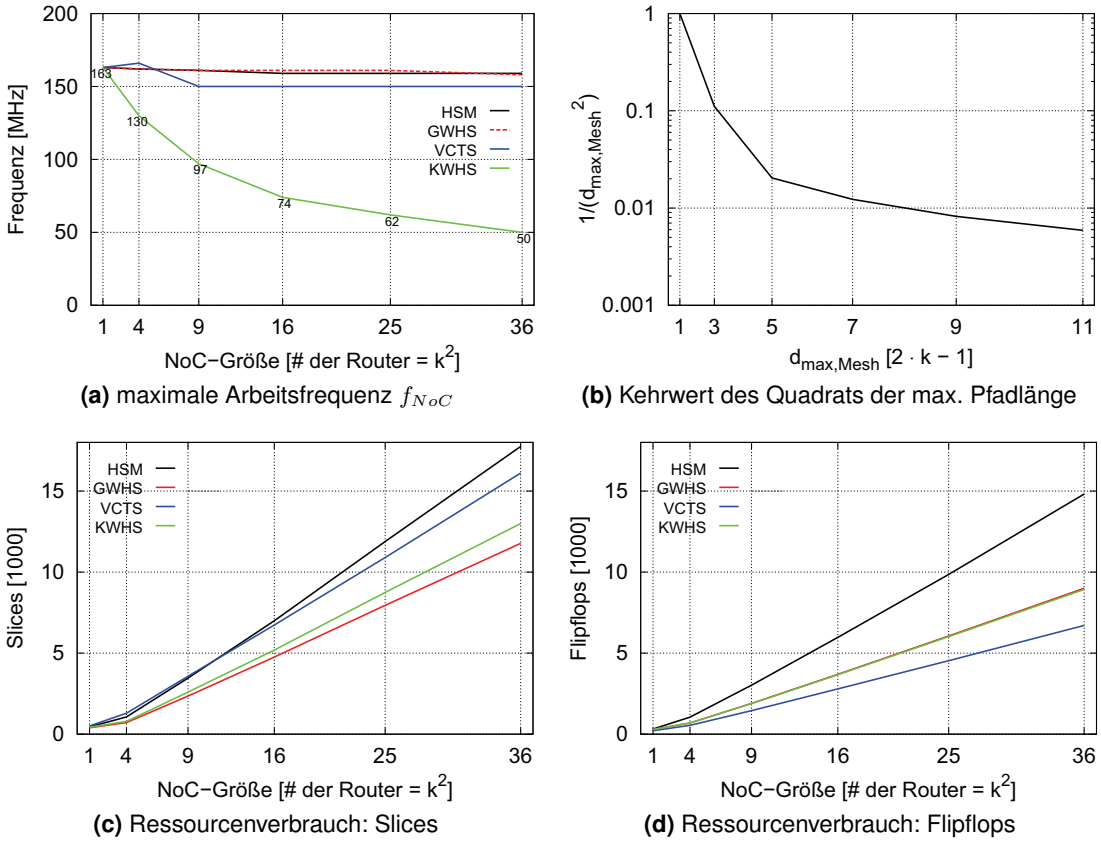
Im Folgenden werden die Syntheseergebnisse verschiedener NoC-Infrastrukturen und die Leistung für den blockierungsfreien Fall diskutiert, um die Vorteile des entwickelten HSM aufzuzeigen. Die Synthese erfolgte – soweit nicht anders angegeben – für ein Xilinx Virtex-4 FX100 FPGA (XC4VFX100-10) mit dem Integrated Software Environment 9.1 unter Standardeinstellungen.

Vergleich von HSM mit klassischen Flusskontrollverfahren Abbildung 6.14a zeigt die Abhängigkeit der maximalen Arbeitsfrequenz eines NoC von der Größe der NoC-Infrastruktur. Zudem ist in Abbildung 6.14b der Kehrwert des Quadrats des NoC-Durchmessers $d_{max,Mesh}$ aufgetragen (siehe Formel (6.8)). Die synthetisierten NoCs besitzen die unter Abschnitt 6.2.2 angegebenen Eigenschaften und unterscheiden sich ausschließlich im Verfahren der Datenflusskontrolle. Aus diesem Grund besitzen einzelne Router der verschiedenen NoC-Varianten dieselbe maximale Frequenz von ca. 163 MHz. Abhängig vom Flusskontrollverfahren verhält sich die jeweilige maximale Frequenz der NoC-Infrastrukturen jedoch unterschiedlich. Die leichten Abweichung im Bereich von $k = 2$ sind durch die Optimierungsprozesse des Synthesetools begründet, da nicht benötigte bzw. nicht verbundene Routerkomponenten am Rand der NoC-Topologien entfernt werden und somit die Komplexität eines einzelnen Routers geringfügig verändert werden kann. Gleiches gilt auch für die Abbildungen 6.14c und 6.14d. Das Augenmerk gilt an dieser Stelle jedoch dem Verlauf der untersuchten Größen bei zunehmendem k.

Bei KWHS stellt das kombinatorische Bestätigungssignal den kritischen Pfad der Schaltung dar. f_{NoC} sinkt mit zunehmender Größe des NoC und verhält sich entsprechend des Verlaufs von ${d_{max,Mesh}}^{-2}$ in Abbildung 6.14b[5]. Diese Ähnlichkeit ist mittels Formel (6.3) zu begründen. Es gilt $f \sim {t_D}^{-1} \sim l^{-2}$. Dabei kann l durch $d_{max,Mesh}$ ersetzt werden. GWHS, VCTS und HSM hingegen zeigen keine Abhängigkeit der Maximalfrequenz von den NoC-Dimensionen. GWHS und VCTS nutzen getaktete Bestätigungssignale. Der kritische Pfad wird durch die Register in separate Teilstücke aufgetrennt. HSM basiert auf einem mesochronen Taktschema mit regionalen Taktdomänen. Mit ca. 160 MHz erreichen HSM und GWHS die höchste Frequenz. VCTS erreicht ca. 150 MHz.

Abbildung 6.14c stellt den Ressourcenverbrauch der verschiedenen NoC-Varianten dar. HSM benötigt ungefähr genauso viele Slices wie VCTS und ca. 40 % mehr Slices als GHWS und KWHS. Dies ist auf die erweiterten Zustandsmaschinen der Sendelogik von HSM zurückzuführen. Der hohe Bedarf an Slices für VCTS ist durch die Logik zur Ansteuerung der FIFO-Puffer begründet. Der zusätzliche Bedarf an Flipflops für die CDC-Schaltungen innerhalb eines HSM-NoC ist in

[5] In Abbildung 6.14b wird eine logarithmische Darstellung genutzt, um den Verlauf des Kehrwertes von ${d_{max,Mesh}}^2$ besser zu verdeutlichen.



(a) maximale Arbeitsfrequenz f_{NoC}

(b) Kehrwert des Quadrats der max. Pfadlänge

(c) Ressourcenverbrauch: Slices

(d) Ressourcenverbrauch: Flipflops

Abbildung 6.14.: Gegenüberstellung der Syntheseergebnisse verschiedener NoC-Varianten

Abbildung 6.14d erkennbar. Registerelemente sind in FPGAs jedoch nicht die kritische Ressource. Ein relativ großes 6×6-NoC auf Basis von HSM nutzt z. B. nur ca. 17 % der Flipflops des Ziel-FPGAs. Aufgrund der Nutzung dedizierter Speicherblöcke (Blockrams) für die Paketpuffer benötigt VCTS die geringste Menge an Registerelementen. Jedoch stellen gerade Blockrams eine knappe und kostspielige Ressource in FPGAs dar.

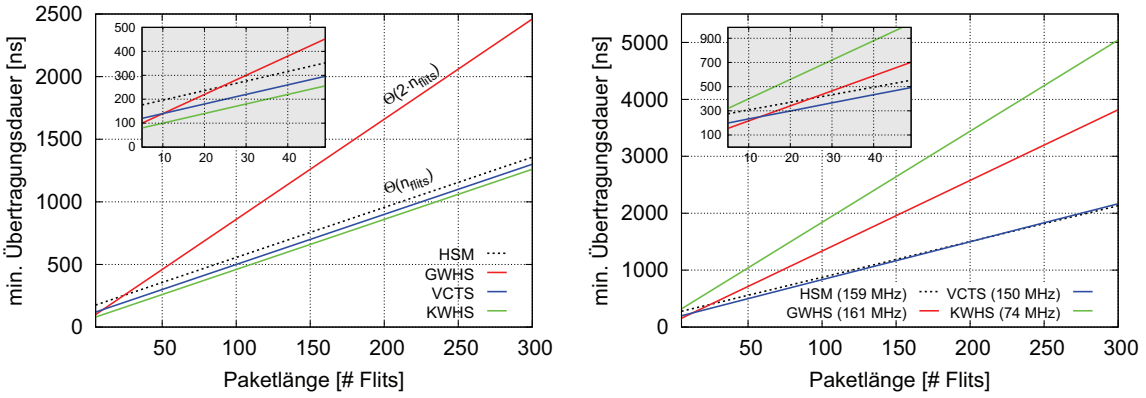
Tabelle 6.2 gibt die Relationen der potentiellen Kommunikationsleistung der verschiedenen NoC-Varianten für einen 4-fachen 2-Würfel wieder. Aus f_{NoC} (siehe Abbildung 6.14a) und W_{bit} (32 Bit) lässt sich die physikalische Maximalbandbreite eines einzelnen unidirektionalen Kanals ermitteln. Im HSM-NoC erreicht ein einzelner unidirektionaler Kanal eine Bandbreite von ca. 5,088 Gbit/s. Die maximale Kanalbandbreite von KWHS ist hingegen stark durch den geringen Arbeitstakt von 74 MHz beeinträchtigt und erreicht nur einen Wert von ca. 2,268 Gbit/s. Mithilfe der Formeln (6.5) & (6.6) ergibt sich für ein 4×4-NoC auf HSM-Basis eine kumulierte physi-

Tabelle 6.2.: Kommunikationsleistung eines 4×4-NoC für verschiedene Flusskontrollverfahren

	VCTS	GWHS	KWHS	HSM
f_{NoC} [MHz]	150	161	74	159
physikalische Kanalbandbreite [Gbit/s]	$\approx$ 4,8	$\approx$ 5,152	$\approx$ 2,368	$\approx$ 5,088
physikalische Gesamtbandbreite [Gbit/s]	$\approx$ 230	$\approx$ 247	$\approx$ 113	$\approx$ 244
effektive Gesamtbandbreite [Gbit/s]	< 230	$< \frac{247}{2}$	< 113	< 244

kalische Gesamtbandbreite von ca. 244 Gbit/s. Auch das VCTS-NoC bietet eine physikalische Gesamtbandbreite von mehr als 200 Gbit/s. Diese *theoretischen* Maximalwerte sind jedoch in der Praxis *nicht* erzielbar. Sie sind vielmehr von sowohl der Effizienz des entsprechenden Flusskontrollverfahrens, von der Paketlänge und dem Routing-Overhead als auch vom tatsächlichen Verkehrsaufkommen bzw. -muster der jeweiligen Applikation und den daraus resultierenden Blockierungen im NoC abhängig. Die effektiv nutzbare Bandbreite von GWHS ist z. B. von vornherein nur auf maximal die Hälfte der in der Tabelle angegebenen physikalischen Gesamtbandbreite von 247 Gbit/s limitiert, da für die Übermittlung eines einzelnen Flits im günstigsten Fall zwei Takte benötigt werden.

Aus diesem Grund vergleicht Abbildung 6.15a die Effizienz der Flusskontrollverfahren anhand der minimalen Latenz t_0 einer Paketübertragung in Abhängigkeit von der Paketlänge. In der Literatur wird t_0 auch als *Zero-Load Latency* bezeichnet, d. h. die Latenz im last- bzw. blockierungsfreien Fall ($n_{blocked} = 0$) [DT03]. Für die Berechnungen wurden die Formeln (6.9), (6.10) und (6.14) genutzt. Für alle Varianten wird in Abbildung 6.15a eine Frequenz $f_{NoC} = 250$ MHz angenommen ($T_{clk} = 4$ ns) sowie eine Übertragungsstrecke mit $d = 5$, um den unterschiedlichen Routing-Overhead bei kleinen Paketen im Diagramm erkennbar zu machen. Für KWHS, GWHS und HSM beträgt die Router-interne Verzögerung $n_{route} = 3$, für VCTS gilt $n_{route} = 5$ (siehe dazu auch Anhang C.1). Desweiteren wird für HSM die untere Grenze von t_0 mit $n_{sync} = 4$ genutzt. Der Kurvenverlauf der Übertragungsdauer von GWHS zeigt deutlich das schnellere Wachstum ($t_{GWHS} = f(2 \cdot n_{flits})$) gegenüber KWHS, VCTS und HSM mit $t = f(n_{flits})$. Die minimale Latenz eines Pakets mit HSM ist etwas höher als bei VCTS bzw. KWHS, was auf die zusätzliche Synchronisationsverzögerung während der Setup-Phase von HSM zurückzuführen ist. Dieser Anteil ist vor allem bei der Übertragung kurzer Pakete relativ hoch, wie der vergrößerte Ausschnitt in Abbildung 6.15a zeigt. Der zeitliche Overhead der Setup-Phase von HSM kann jedoch durch einen hohen Arbeitstakt der NoC-Infrastruktur ausgeglichen werden. Dazu stellt Abbildung 6.15b dieselben Zusammenhänge mit Bezug auf die *tatsächlichen Syntheseergebnisse* der einzelnen NoC-Varianten dar. Für die Berechnungen wurden die erzielten Maximalfrequenzen



(a) Vergleich ohne Synthesebezug, $f_{NoC} = 250$ MHz für alle Verfahren

(b) Vergleich mit Synthesebezug, individuelle f_{NoC} entsprechend der Syntheseergebnisse

Abbildung 6.15.: Vergleich der minimalen Übertragungsdauer eines Pakets bei VCTS, GWHS, KWHS und HSM über der Paketlänge.

genutzt (siehe Tabelle 6.2). Dieser Synthesebezug verändert die Relationen der einzelnen Verfahren zueinander. Bei KWHS und GWHS zeigen sich nun erhebliche Schwächen durch geringe Skalierbarkeit (KWHS) und ineffiziente Bandbreitenausnutzung (GWHS). HSM und VCTS sind in etwa gleich schnell, jedoch beansprucht ein NoC auf VCTS-Basis wertvolle Speicherblöcke (Blockrams) des FPGAs.

Vergleich des HSM-NoC mit dem Stand der Technik Bisher sind aufgrund der primären Ausrichtung auf ASICs nur relativ wenige Forschungsarbeiten zu NoC-Plattformen auf FPGA-Basis veröffentlicht. In Tabelle 6.3 werden Syntheseergebnisse des entwickelten HSM-NoC (jeweils in Klammern) mit ähnlichen Vertretern aus der Fachliteratur verglichen. Dazu wurde das HSM-NoC für den in der jeweiligen Quelle angegebenen FPGA-Typ synthetisiert.

In [SBKV05] wird mit *LiPaR* ein NoC auf 2D-Gitter-Basis mit XY-Routing und Store-and-Forward-Switching (SAFS) vorgestellt. Der Fokus von LiPaR liegt ebenfalls auf einem geringen Ressourcenverbrauch, um die Logikelemente des FPGAs für die eigentliche Applikation nutzen zu können. Ein LiPaR-Router benötigt etwas weniger Slices als ein HSM-Router, aber ca. 28 % mehr Flipflops. Zusätzlich werden pro Router 10 Blockrams für SAFS benötigt. Zudem erreicht LiPaR nur ca. ein Viertel der maximalen Arbeitsfrequenz des HSM-NoC. Das in [ZKS04] vorgestellte *RASoC* nutzt ebenfalls XY-Routing in einem k-fachen 2-Würfel. Das auf einem Altera-FPGA synthetisierte RASoC ist langsamer als das HSM-NoC. Der Hardwarebedarf ist fast doppelt so groß. Das *PNoC* in [HN06] basiert auf einer segmentierten Gittertopologie, nutzt tabellenba-

Tabelle 6.3.: Vergleich des eigenen NoC (Werte jeweils in Klammern) mit anderen FPGA-basierten NoC-Entwicklungen

	LiPaR [SBKV05]	RASoC [ZKS04]	PNoC [HN06]	Open-Source-NoC [EL07]
allg. Eigenschaften	2D-Gitter, XY-Routing, Store-and-Forward Switching	2D-Gitter, XY-Routing, Wormhole Switching, FIFO-Puffer	segmentiertes 2D-Gitter, tabellenbasiertes Routing, leitungsvermittelt	2D-Gitter, XY-Routing, Wormhole Switching, FIFO-Puffer
genutzte FPGA-Plattform	Xilinx XC2VP30-5	Altera EPF10K200S	Xilinx XC2VP30-7	Xilinx XC4VLX80-10
Slices p. Router	437 (487)	1830 (1115)	336 (486)	552 (486)
Flipflops p. Router	478 (373)	745 (326)	k. A. (373)	572 (373)
Besonderes	Blockrams 10 (0)	k. A.	Blockrams 1 (0)	LUTs 1090 (868)
f_{NoC} [MHz]	33 (127)	56 (62)	138 (165)	173 (163)

siertes statisches Routing und ist leitungsvermittelt. Ein 5-Port-PNoC-Router benötigt ca. 25 % weniger Slices, ist jedoch 30 MHz langsamer. Zudem wird ein Blockram für die Routing-Tabelle benötigt. Ehliar et al. stellen in [EL07] den Quellcode eines frei verfügbaren Open-Source-NoC bereit[6]. Es basiert wiederum auf einer 2D-Gittertopologie, XY-Routing und WHS. Ein einzelner Router erreicht unter gleichen Synthesebedingungen auf dem Ziel-FPGA eine ca. 10 MHz höhere Frequenz, erfordert aber auch ca. 15 % mehr Slices und 50 % mehr Register. Dieser Mehrbedarf ist u. a. auf die zusätzliche Logik für die Ansteuerung der genutzten FIFO-Puffer zurückzuführen. Zudem werden spezielle Shift-Register des FPGAs als Speicherelemente für die FIFO-Puffer genutzt, wodurch ein Router des Open-Source-NoC ungefähr 25 % mehr Lookup-Tables (LUTs) als ein HSM-Router benötigt.

In keiner der Referenzen wurden Angaben über die Skalierbarkeit oder lastfreie Latenz des jeweiligen Ansatzes gemacht. Es wird jedoch angenommen, dass t_0 bei LiPaR deutlich höher ist, da SAFS genutzt wird. Das PNoC basiert auf Leitungsvermittlung, wodurch es ähnlich wie HSM bei kurzen Paketen einen hohen Setup-Overhead aufweist. Das RASoC und das Open-Source-NoC nutzen in den Routern größere Puffer, wodurch die Gesamtkapazität dieser NoCs steigt. Prinzipiell hat dies keine Auswirkungen auf t_0, jedoch muss ein Flit abhängig von der Realisierung der Puffer ggf. erst durch alle Pufferstufen propagieren, bevor es weitergeleitet werden kann. Dies kann sich negativ auf t_0 auswirken, lässt sich aus der genannten Literatur aber nicht erschließen.

[6]Der Quellcode ist unter `http://www.da.isy.liu.se/research/soc/fpganoc` verfügbar.

Zusammenfassung der Eigenschaften von HSM Im Folgenden sind die wesentlichen Charakteristika von HSM zusammengefasst. Tabelle 6.4 erweitert Tabelle 6.1 um die Eigenschaften von HSM und stellt diese den Standardverfahren gegenüber.

Tabelle 6.4.: Eigenschaften von HSM und der typischen Flusskontrollverfahren

	VCTS	GWHS	KWHS	HSM
Ressourcenbedarf	große Paket-Puffer	kleine Flit-Puffer	kleine Flit-Puffer	kleine Flit-Puffer
Arbeitstakt f_{NoC}?	$\approx f_{Router}$, const	$\approx f_{Router}$, const	$= f(k_x, k_y)$	$\approx f_{Router}$, const
Latenz t	$= f(n_{flits})$	$= f(2 \cdot n_{flits})$	$= f(n_{flits})$	$= f(n_{flits})$
idealer Durchsatz	$\approx BW_{NoC}$	$\approx BW_{NoC}/2$	$\approx BW_{NoC}$	$\approx BW_{NoC}$

- HSM benötigt nur kleine Flit-Puffer in den Ports der NoC-Router.
- Durch das mesochrone Taktschema wird der maximale Arbeitstakt der NoC-Infrastruktur durch die Komplexität eines einzelnen Routers bestimmt. f_{NoC} ist *keine* Funktion von k_x und k_y.
- $t_{0,HSM}$ ist einfach linear abhängig von n_{flits}. Bei langen Paketen kann die bestmögliche Effizienz von einem Flit pro Takt angenähert werden. Während der Setup-Phase sind jedoch 7 Takte pro Flit notwendig. Dies wirkt sich bei der Übertragung kurzer Pakete nachteilig aus, kann aber durch eine hohe f_{NoC} kompensiert werden.
- HSM ist prinzipiell von der Topologie der NoC-Infrastruktur unabhängig, erfordert jedoch das Vorhandensein eines deterministischen Routing-Algorithmus. Dies ist durch XY-Routing gegeben.
- Die zusätzlichen CDC-Schaltungen erfordern ausschließlich einfache Registerelemente, welche in aktuellen FPGAs ausreichend vorhanden sind.

HSM nutzt somit die Einfachheit des XY-Routings und die Skalierbarkeit von GALS-Systemen. HSM kombiniert dabei sowohl den geringen Ressourcenverbrauch von GWHS/KWHS als auch die hohe Effizienz von VCTS.

6.4. Border-Enhanced Mesh

Motivation Die Topologie einer NoC-Infrastruktur hat wesentlichen Einfluss auf ihren Ressourcenverbrauch und ihre Kommunikationsleistung. Der Hardwarebedarf ist durch die Komplexität eines einzelnen NoC-Routers aber auch durch deren absolute Anzahl bestimmt. Die Kommunikationsleistung ist u. a. durch die Latenz und die tatsächlich nutzbare Gesamtbandbreite der NoC-Infrastruktur bestimmt. Die Optimierung der NoC-Topologie ist daher durch verschiedene Gesichtspunkte beeinflusst:

- Eine Reduktion der Hardwarekosten des NoC ist gerade in FPGAs immer sinnvoll, um die begrenzt vorhandenen Logikressourcen für die eigentliche Applikation aufzusparen.
- Der Grad der NoC-Router, d. h. die Anzahl der Ports, dürfen nicht zusätzlich erhöht werden. In der Diplomarbeit von Pribbernow [Pri07] wird u. a. gezeigt, dass dadurch der Hardwarebedarf der Schaltmatrizen innerhalb der Router ansteigt. Gleichzeitig zeigt die Studienarbeit von Sofke [Sof07], dass mit steigender Komplexität der Schaltmatrix die maximale Arbeitsfrequenz eines einzelnen Routers und damit der NoC-Infrastruktur zunehmend reduziert wird, was sich wiederum schmälernd auf Leistungsparameter wie Latenz und Bandbreite auswirkt.
- Der im vorherigen Abschnitt vorgestellte HSM besitzt bei kurzen Nachrichten einen relativ hohen Routing-Overhead. Eine Verkürzung der zeitintensiven Setup-Phase sowie die Reduzierung der lastfreien Latenz i. Allg. tragen deshalb zur Leistungssteigerung bei.
- Die Beibehaltung einer regulären, allgemeinen und schlanken NoC-Infrastruktur spielt eine wesentliche Rolle. Dally argumentiert in [DT03]: „It is almost always better to use a good general purpose network than to design a topology matched to the problem.“

Im Folgenden wird mit Blick auf die genannten Aspekte eine optimierte anwendungsunabhängige Topologie – das **B**order-**E**nh**A**nced **M**esh (BEAM) – vorgestellt. Ein HSM-basierter k-facher 2-Würfel dient dabei als Ausgangsbasis.

6.4.1. BEAM – Prinzip

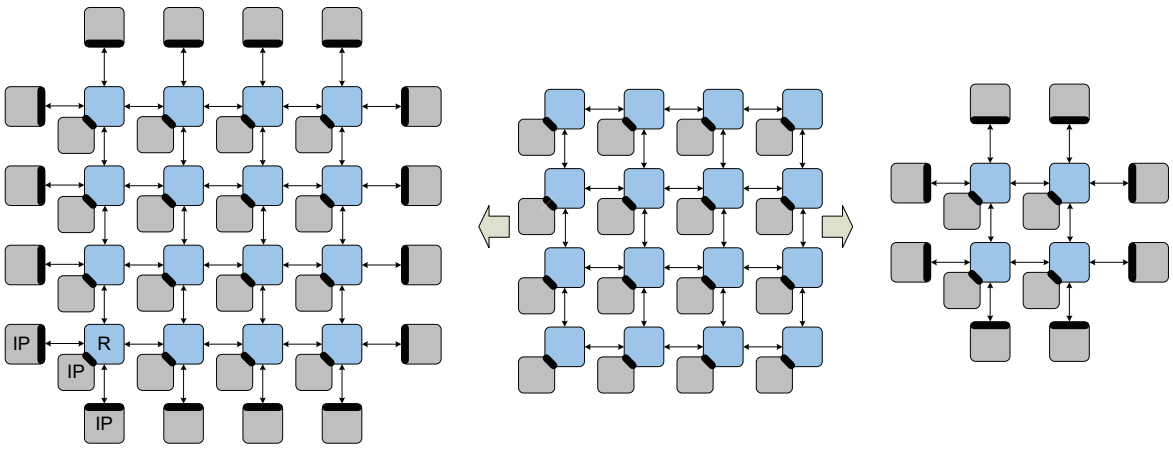
Die prinzipielle Idee von BEAM basiert auf einer geschickten Anordnung von IP-Cores und Routern, welche jedoch nach wie vor auf einer einfachen zweidimensionalen Gitterstruktur beruht. Typischerweise ist in einem k-fachen 2-Würfel an jeden Router ein IP-Core angeschlossen. Das Verhältnis der Anzahl der IP-Cores zur Anzahl der Router – die sogenannte Core-to-Router-Ratio (CRR) bzw. der Konzentrationsfaktor – ist 1. Die CRR ergibt sich nach (6.15). N ist die

Anzahl der IP-Cores und R die Anzahl der Router in einer NoC-Topologie. Für einen k-fachen 2-Würfel sind N und R durch (6.16) bestimmt und können bei gleicher Kantenlänge ($k = k_x = k_y$) zusätzlich vereinfacht werden.

$$CRR = \frac{N}{R} \tag{6.15}$$

$$N_{Mesh} = R_{Mesh} = k_x \cdot k_y = k^2 \tag{6.16}$$

In einer BEAM-Topologie werden die bisher offenen Router-Ports im Randbereich der NoC-Infrastruktur als direkte Schnittstelle zu einem IP-Core genutzt, ohne dass noch ein weiterer Router dazwischengeschaltet ist. Ausgehend von einem klassischen 4-fachen 2-Würfel (4×4-Gitter) in Abbildung 6.16b zeigen die Abbildungen 6.16a und 6.16c die daraus abgeleiteten BEAM-Topologien. Abbildung 6.16a zeigt eine 4×4-BEAM-Topologie, welche durch Anfügen zusätzlicher IP-Cores (IP) an die freien Ports der Router (R) entstanden ist. Es können 32 IP-Cores an die gleiche Anzahl an Routern angeschlossen werden (CRR = 2). Abbildung 6.16c hingegen zeigt eine 2×2-BEAM-Topologie, welche durch die Entnahme von Routern im Randbereich des 4-fachen 2-Würfels entstanden ist. An die vier verbleibenden Router können immer noch 12 IP-Cores angeschlossen werden (CRR = 3). In einer BEAM-Topologie existieren im Randbereich Router mit zwei bzw. drei direkt verbundenen IP-Cores. Die Router an sich spannen jedoch nach wie vor einen regulären k-fachen 2-Würfel auf.



(a) BEAM, $k = 4$ CRR = 2 **(b)** 4-facher 2-Würfel, CRR = 1 **(c)** BEAM, $k = 2$, CRR = 3

Abbildung 6.16.: Anordnung von IP-Cores und Routern in der BEAM-Topologie

Tabelle 6.5.: CRR k-facher 2-Würfel und verschiedener BEAM-Topologien

k	R	k-fache 2-Würfel		BEAM	
		N_{Mesh}	CRR_{Mesh}	N_{BEAM}	CRR_{BEAM}
1	1	1	1	5	5
2	4	4		12	3
3	9	9		21	2,33
4	16	16		32	2
5	25	25		45	1,8
6	36	36		60	1,66
7	49	49		77	1,57
8	64	64		96	1,5
9	81	81		117	1,44
10	100	100		140	1,4

$$R_{BEAM} = k_x \cdot k_y = k^2 \tag{6.17}$$

$$N_{BEAM} = 2 \cdot k_x + 2 \cdot k_y + k_x \cdot k_y = 4 \cdot k + k^2 \tag{6.18}$$

Tabelle 6.5 enthält Werte der CRR verschiedener NoC-Topologien nach den Formeln (6.15) bis (6.18). CRR_{BEAM} ist immer größer als 1, strebt jedoch mit zunehmenden NoC-Dimensionen gegen 1. BEAM zielt dabei vor allem auf die *Einsparung von NoC-Routern* und damit von Logikressourcen im FPGA, was sich insbesondere bei praktikablen NoC-Größen mit $2 \leq k \leq 10$ bemerkbar macht. Es können bei einer deutlich geringeren Anzahl an Routern ähnlich viele bzw. sogar mehr IP-Cores in ein NoC-System integriert werden, wie z. B. die hervorgehobenen Werte in Tabelle 6.5 zeigen. In einer 3×3-BEAM-Topologie können bis zu 21 IP-Cores integriert werden, während verglichen mit einem 4×4-Gitter 6 Router eingespart werden. Auch rechteckige Topologien mit ungleicher Kantenlänge in X- und Y-Richtung sind realisierbar. Ein durchaus möglicher Sonderfall ist ein aus einem NoC-Router bestehendes „Netzwerk“, welches fünf IP-Cores miteinander verbinden kann. Im Folgenden werden aus Gründen der Einfachheit jedoch nur Topologien mit $k = k_x = k_y$ betrachtet.

6.4.2. Adressierung und Routing in einer BEAM-Topologie

Abbildung 6.17 zeigt das Adressierungsschema der BEAM-Topologie am Beispiel eines 3×3-BEAM-NoC. Die Kantenlänge des durch die Router aufgespannten Gitters beträgt $k = 3$. Die

maximale Kantenlänge des gesamten NoC-Systems beträgt jedoch $x_{max} = y_{max} = k + 2 = 5$. Die flache Adressierung eines 2D-Gitters bleibt erhalten. Der wesentliche Unterschied zu einem klassischen k-fachen 2-Würfel besteht darin, dass das komplette Router-Netzwerk um jeweils einen Schritt in X- und Y-Richtung verschoben ist. D. h., die Adressierung der Router beginnt nicht bei Koordinate (0;0), sondern bei Koordinate (1;1). In gleicher Weise endet die Adressierung der Router bei $(x_{max} - 2; y_{max} - 2)$. Die individuellen Adressen der IP-Cores liegen im Bereich $(0 \ldots x_{max} - 1; 0 \ldots y_{max} - 1)$. Ein an den lokalen Port eines Routers angeschlossener IP-Core besitzt dieselbe Adresse wie der jeweilige Router. Die IP-Cores am Rand der Topologie besitzen die Adressen $(0;y)$ an der West-Seite, $(x;0)$ an der Süd-Seite, $(x_{max} - 1;y)$ an der Ost-Seite und $(x;y_{max} - 1)$ an der Nord-Seite. In einem BEAM-basierten NoC existieren weder IP-Cores noch Router mit den Adressen (0;0), $(x_{max} - 1;0)$, $(0;y_{max} - 1)$ oder $(x_{max} - 1;y_{max} - 1)$.

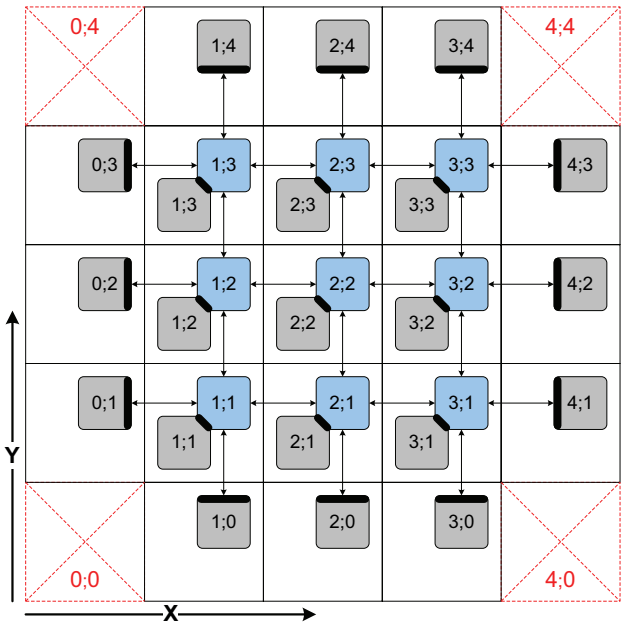
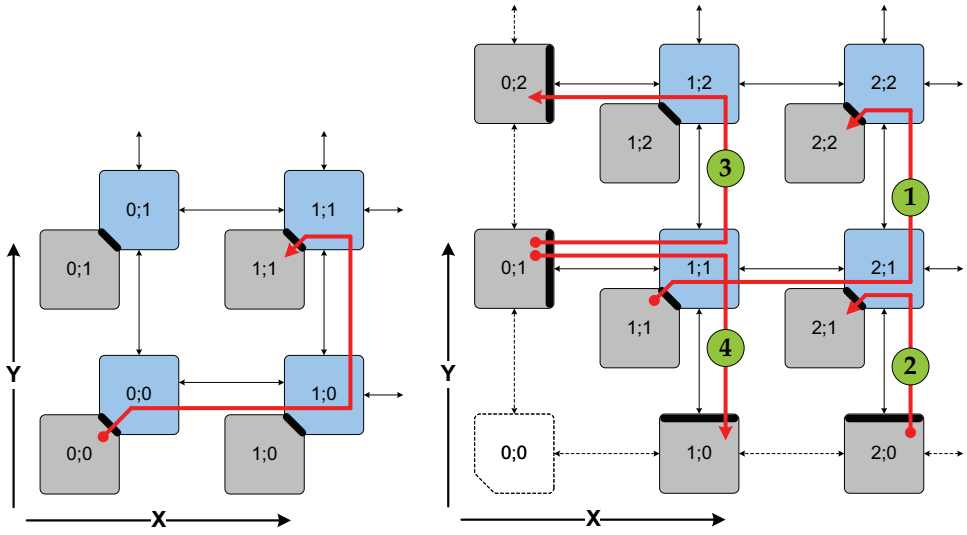


Abbildung 6.17.: Adressierungsschema einer BEAM-Topologie am Beispiel $k = 3$

Dieses Adressierungsschema hat den Vorteil, dass nach wie vor die Nutzung von einfachem XY-Routing mit geringfügigen Modifikationen am Rand des NoC möglich ist. Empfängt ein Randrouter ein Paket für einen der an ihn angeschlossenen äußeren IP-Cores, wird das Paket prinzipiell in Richtung dieses IP-Cores geleitet, als wäre dort noch ein weiterer Router vorhanden. Das Fehlen des entsprechenden Routers ist dabei völlig transparent für den sendenden Router. Dies ist möglich, da an allen Router-Ports die gleichen Schnittstellen zum Einsatz kommen (siehe Abbildung 6.9c). Die Entnahme von Routern und damit auch von Übertragungskanälen am Rand

eines 2D-Gitters erfordert nur in den westlichen und östlichen Randroutern eine Anpassung der XY-Routing-Logik an die veränderte Topologie. Nördliche, südliche und zentrale Router in der NoC-Infrastruktur, z. B. Router (2;2) in Abbildung 6.17, erfordern keine Anpassungen.



(a) XY-Routing im k-fachen 2-Würfel **(b)** XY-Routing in einer BEAM-Topologie

Abbildung 6.18.: XY-Routing in einem k-fachen 2-Würfel und einer BEAM-Topologie

Abbildung 6.18 verdeutlicht anhand von Beispielpfaden die Unterschiede im Routing, welche am Rand der BEAM-Topologie eine Modifikation des XY-Routings erforderlich machen. In Abbildung 6.18a ist der normale Pfad eines Pakets in einem klassischen 2D-Gitter dargestellt (siehe auch Abbildung 6.9a). Mit XY-Routing ergibt sich für das Paket von (0;0) nach (1;1) der dargestellte Übertragungsweg über Knoten (1;0) mit $d = 3$. Im Gegensatz dazu demonstriert Abbildung 6.18b für vier verschiedene Pakete die Wegewahl in einer BEAM-Topologie. Dargestellt ist nur die südwestliche Ecke der NoC-Infrastruktur. An den gegenüberliegenden Ecken bzw. Kanten verhält sich die Wegewahl analog. Die gestrichelten Elemente sind in der BEAM-Topologie *nicht* mehr vorhanden, weswegen die Routinglogik an diesen Stellen anzupassen ist. Die westlichen und südlichen IP-Cores sind ohne einen eigenen Router direkt mit dem jeweils nächsten Router in X- bzw. Y-Richtung verknüpft. Der Pfad von Paket (1) ähnelt dem klassischen Beispiel in Abbildung 6.18a, da weder *von* noch *zu* einem IP-Core am Rand der BEAM-Topologie geroutet wird. Die Übertragung von (1;1) nach (2;2) erfolgt über Router (2;1) mit $d = 3$. Für die Übertragung von Paket (2) sind ebenfalls keine Modifikationen des XY-Routings notwendig, da das Paket nur in der Y-Dimension geroutet werden muss. Die Übertragung erfolgt direkt von IP-Core (2;0) über

Router (2;1) zu IP-Core (2;1) mit $d = 1$. Hingegen wäre in einem klassischen 2D-Gitter $d = 2$. Die Wegewahl für Paket (3) erfordert jedoch eine Modifikation des Routing-Algorithmus. Empfängt Router (1;1) das Paket von IP-Core (0;1), würde es mittels herkömmlichem XY-Routing wieder in Richtung (0;1) zurückgesendet werden, da normalerweise zuerst die X-Dimension ausgeglichen wird. Da IP-Core (0;1) jedoch unmittelbar an Router (1;1) angeschlossen ist, ist kein direkter Datenkanal in Y-Richtung zwischen (0;1) und (0;2) vorhanden. Deshalb muss Router (1;1) das Paket zuerst in Y-Richtung zu Router (1;2) weiterleiten, welcher es dann in Richtung (0;2) umlenkt. Trotz des Umwegs von Paket (3) über (1;1) und (1;2) ist $d = 2$. Der Pfad ist somit genauso lang wie in einem normalen k-fachen 2-Würfel. Als letztes Beispiel wird der Übertragungspfad von Paket (4) von (0;1) nach (1;0) skizziert, welcher prinzipiell wieder dem typischen Weg nach klassischem XY-Routing entspricht. Der signifikante Unterschied zu einem normalen 2D-Gitter ist jedoch die Reduzierung von $d = 3$ auf $d = 1$.

Die Erkennung derartiger Sonderfälle bei der Wegewahl erfolgt innerhalb der Routingfunktion durch einen Vergleich der eigenen Adresse des jeweiligen Routers mit der Zieladresse des Pakets unter Berücksichtigung von x_{max} und y_{max}. Die Anpassungen des Routing-Algorithmus sind im VHDL-Quellcode (Very High Speed Integrated Circuit Hardware Description Language) des NoC generisch integriert und werden abhängig von der relativen Position des Routers im NoC zum Synthesezeitpunkt ausgewählt. Waschki [Was08] zeigt, dass die Deadlockfreiheit des klassischen XY-Routings durch diese Modifikationen nicht verletzt wird. Im Anhang unter Abschnitt C.2 sind Auszüge aus dem Quellcode der Routingfunktion gegeben, welche die Modifikationen für die BEAM-Topologien zusätzlich erläutern.

6.4.3. Bewertung des BEAM-Ansatzes

Bezogen auf die unter Abschnitt 6.4 genannten Motivationsgründe erfolgt an dieser Stelle eine ausführliche Evaluation des BEAM-Ansatzes. Nach Dally et al. [DT03] und Duato et al. [DYN03] wird ein Verbindungs- und Kommunikationsnetzwerk durch die beiden Leistungsparameter Latenz und Durchsatz sowie durch die Hardwarekosten bewertet. Es erfolgt ein Vergleich mit klassischen k-fachen 2-Würfeln, da sich BEAM-Topologien aus zweidimensionalen Gitterstrukturen ableiten (siehe Abbildung 6.16). Zunächst werden verschiedene Kenngrößen und grundlegende Zusammenhänge erläutert. Danach erfolgt eine Diskussion von einerseits analytischen Zusammenhängen und andererseits von Simulationsergebnissen, um BEAM-Topologien und k-fache 2-Würfel bzgl. Latenz und Durchsatz gegenüberzustellen. Daraufhin werden Syntheseergebnisse genutzt, um aufzuzeigen, inwiefern der BEAM-Ansatz die Hardwarekosten reduziert. Abschließend wird die BEAM-Topologie mit dem Stand der Technik verglichen.

Allgemeine Kenngrößen und Zusammenhänge Für einen direkten Leistungsvergleich verschiedener NoC-Topologien werden i. Allg. synthetische Verkehrsmuster genutzt. Am verbreitetsten ist die Nutzung eines *uniformen zufälligen* Verkehrsschemas, welches auch im Weiteren verwendet wird. Jeder IP-Core generiert dabei mit einer definierten Eingangsdatenrate Pakete konstanter Größe. Die jeweilige Datensenke wird zufällig und mit der gleichen Wahrscheinlichkeit aus den übrigen IP-Cores ausgewählt. Abbildung 6.19 zeigt dazu unabhängig von einer konkreten Topologie und einem bestimmten Verkehrsschema den allgemeinen charakteristischen Verlauf der durchschnittlichen Latenz t_{avrg} über der Eingangsdatenrate.

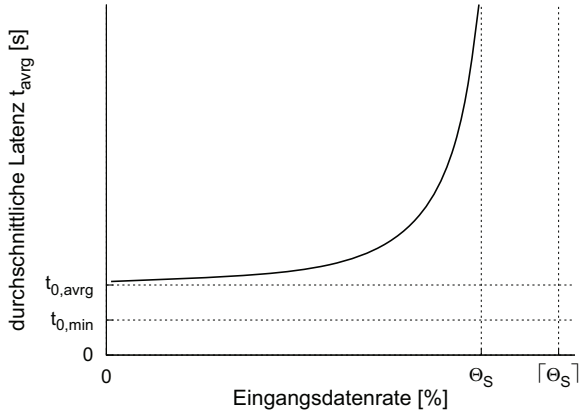


Abbildung 6.19.: Charakteristischer Verlauf der Latenz über der Eingangsdatenrate

In Abschnitt 6.3.3 wurde bereits der Begriff der lastfreien Latenz t_0 als untere Schranke der Übertragungsdauer für ein bestimmtes Paket und einen konkreten Pfad der Länge d eingeführt. Die *minimale* lastfreie Latenz $t_{0,min}$ ergibt sich ebenfalls aus den Formeln (6.9), (6.10) und (6.14) unter Nutzung der minimalen Pfadlänge d_{min} einer Topologie. $t_{0,min}$ stellt die absolute untere Grenze der Latenz in Abbildung 6.19 dar. Die *durchschnittliche* lastfreie Latenz $t_{0,avrg}$ ergibt sich mittels der durchschnittlichen Pfadlänge d_{avrg}. Die tatsächliche durchschnittliche Latenz t_{avrg} nähert sich bei geringen Eingangsdatenraten $t_{0,avrg}$ asymptotisch an, da kaum Blockierungen im NoC auftreten. Die Parameter d_{min} und d_{avrg} sind abhängig von der jeweiligen Topologie. Allgemein kann d_{avrg} durch (6.19) ermittelt werden. Für einen k-fachen 2-Würfel gilt (6.20).

$$d_{avrg} = \frac{\sum\limits_{A,B \in N} d_{A \Rightarrow B}}{N \cdot (N-1)} \qquad \text{mit } A \neq B \tag{6.19}$$

$$d_{avrg,Mesh} = \frac{2 \cdot k}{3} + 1 \qquad \text{mit } k_x = k_y \tag{6.20}$$

Wird die Eingangsdatenrate erhöht, vergrößert sich auch t_{avrg}, da sich Pakete im NoC zunehmend gegenseitig blockieren und es zu Verzögerungen kommt. An einem bestimmten Sättigungspunkt strebt die Latenz gegen unendlich. Dieser Punkt ist abhängig von den Eigenschaften der Topologie, vom Verkehrsmuster, vom Routing-Algorithmus sowie vom Flusskontrollverfahren. Die Eingangsdatenrate am Sättigungspunkt wird auch als *Sättigungsdurchsatz* Θ_S bezeichnet. Ist Θ_S erreicht, akzeptiert das NoC auch bei weiterer Erhöhung der Eingangsdatenrate keine zusätzlichen Pakete über Θ_S hinaus. Eine entscheidende Einflussgröße auf Θ_S und die Kommunikationsleistung i. Allg. ist dabei die *Bisektion* B einer Topologie. B ist ein Schnitt durch das Netzwerk, bei dem dieses in zwei annähernd gleich große Teilnetze zerlegt wird. B definiert dabei die Anzahl der Übertragungskanäle, die durch diesen Schnitt aufgetrennt werden. Für k-fache n-Würfel ergibt sich B nach (6.21). Für zweidimensionale Gitter und BEAM-Topologien vereinfacht sich (6.21) zu (6.22). Die *Bisektionsbandbreite* BW_B ist Summe der Einzelbandbreiten der durch die Bisektion aufgetrennten Kanäle. Sind alle Kanäle identisch, gilt Gleichung (6.23). Bei einem zufälligen gleichverteilten Verkehrsaufkommen muss im Durchschnitt die Hälfte aller Pakete die Bisektion einer Topologie queren. Jeder IP-Core trägt dabei den gleichen Anteil zur Gesamtlast bei. Demnach ist die obere Grenze von Θ_S für dieses Verkehrsmuster, der sogenannte ideale Durchsatz, durch (6.24) bestimmt [DT03, DYN03]. Aufgrund von Blockierungen innerhalb der NoC-Infrastruktur und der Eigenschaften des Routing-Algorithmus und des Flusskontrollverfahrens ist Θ_S tatsächlich jedoch *deutlich* niedriger und kann nur simulativ ermittelt werden. Für weitere Informationen dazu sei auf die genannte Literatur verwiesen.

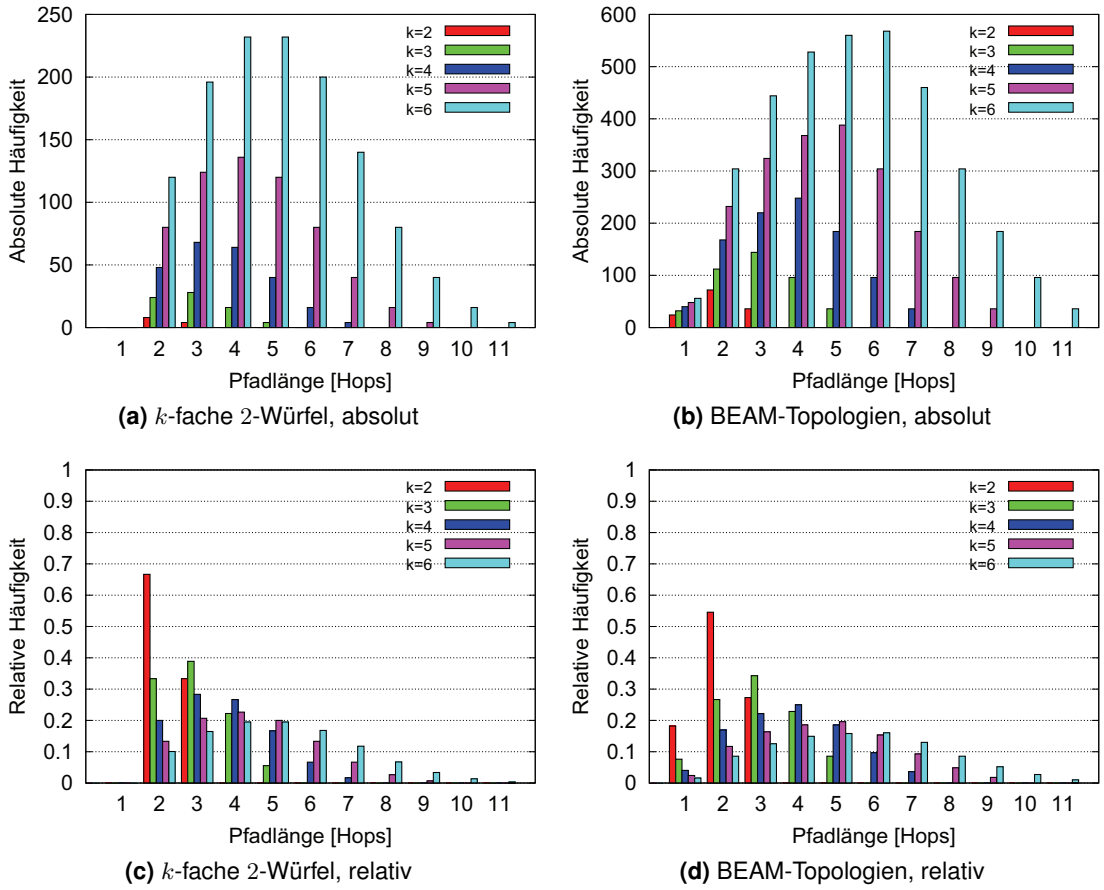
$$B = 2 \cdot k^{n-1} \tag{6.21}$$

$$B_{Mesh} = B_{BEAM} = 2 \cdot k \tag{6.22}$$

$$BW_B = B \cdot BW_{Channel} = B \cdot f_{NoC} \cdot W_{bit} \tag{6.23}$$

$$\lceil \Theta_S \rceil = \frac{2 \cdot BW_B}{N} \tag{6.24}$$

Analytischer Vergleich der Latenz Die Pfadlänge d hat direkten Einfluss auf die Latenz. Die Pfade verschiedener Quelle-Senke-Kombinationen können unterschiedliche Pfadlängen besitzen. In einem NoC existieren bei N IP-Cores $N \cdot (N-1)$ mögliche Quelle-Senke-Kombinationen (siehe Formel (6.19)). Die Diagramme in Abbildung 6.20 stellen in diesem Zusammenhang die Verteilung der minimalen Pfadlängen aller Quelle-Senke-Kombinationen verschiedener Topologien dar. Das verwendete XY-Routing zählt zur Klasse minimaler Routing-Algorithmen, so dass der Informationsaustausch im NoC tatsächlich über diese minimalen Übertragungswege erfolgt. Abbildung 6.20a zeigt anhand der absoluten Häufigkeit die Pfadlängenverteilung für



(a) k-fache 2-Würfel, absolut

(b) BEAM-Topologien, absolut

(c) k-fache 2-Würfel, relativ

(d) BEAM-Topologien, relativ

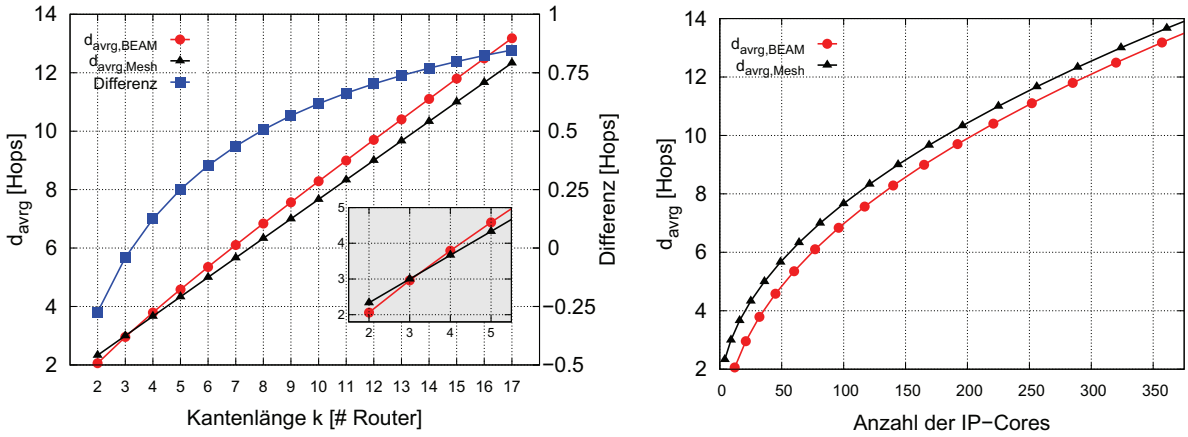
Abbildung 6.20.: Verteilung der Pfadlängen in k-fachen 2-Würfeln und BEAM-Topologien

k-fache 2-Würfel, Abbildung 6.20b für unterschiedlich große BEAM-Topologien. Aufgrund der höheren CRR existieren in BEAM-Topologien absolut mehr Pfade als in k-fachen 2-Würfeln mit gleiche Kantenlänge k. Die maximale Distanz d_{max} ist durch Formel (6.8) bestimmt und für beide Topologien identisch. Die minimale Pfadlänge d_{min} ist jedoch unterschiedlich. In einem k-fachen 2-Würfel ist $d_{min} = 2$. Hingegen ist in einer BEAM-Topologie $d_{min} = 1$. Diese kurzen Wege ergeben sich durch den direkten Anschluss mehrerer IP-Cores an denselben Router. Da dies nur an den äußeren Routern einer BEAM-Topologie möglich ist (siehe z. B. Abbildung 6.17), steigt die Anzahl der Wege mit $d = 1$ mit zunehmender NoC-Größe nur *linear* an. Gleichzeitig steigt jedoch die Anzahl der längeren Pfade in einer BEAM-Topologie *quadratisch* an, da N_{BEAM} schneller wächst als R_{BEAM}. Für kleine k ist deshalb die Pfadlängenverteilung einer BEAM-Topologie durch den Einfluss der kurzen Übertragungswege bestimmt. Für größere k überwiegt hingegen

der Einfluss längerer Pfade. Diese Zusammenhänge sind auch in den Abbildungen 6.20c und 6.20d anhand der relativen Häufigkeit der Pfadlängen sichtbar.

Tabelle 6.6.: Durchschnittliche Pfadlängen k-facher 2-Würfel und BEAM-Topologien

k	2	3	4	5	6	7	8	9	10	11	12	13
$d_{avrg,Mesh}$	2,33	3	3,66	4,33	5	5,66	6,33	7	7,66	8,33	9	9,66
$d_{avrg,BEAM}$	2,06	2,96	3,79	4,58	5,35	6,10	6,84	7,56	8,28	8,99	9,70	10,4



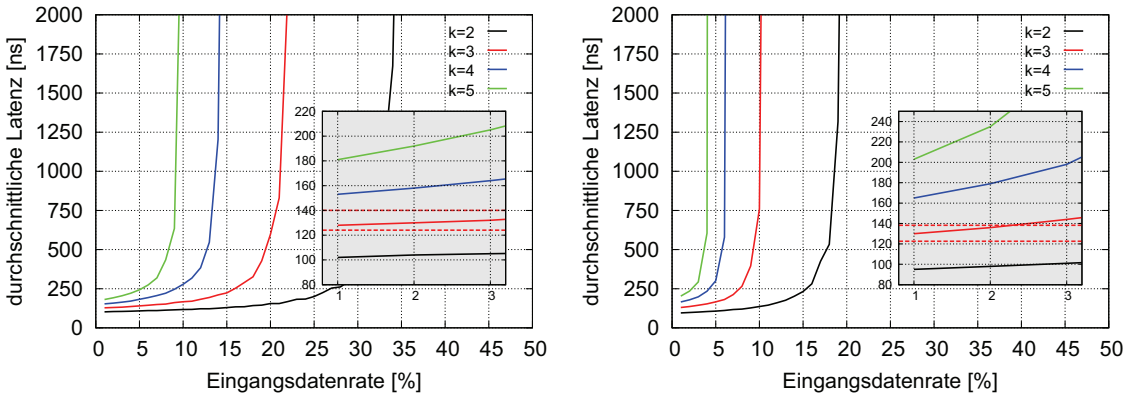
(a) d_{avrg} über der Kantenlänge k **(b)** d_{avrg} über der Anzahl der IP-Cores

Abbildung 6.21.: Durchschnittliche Pfadlängen k-facher 2-Würfel und BEAM-Topologien

Zur Verdeutlichung sind in Abbildung 6.21a die durchschnittlichen Pfadlängen d_{avrg} für beide Topologien über k aufgetragen. Tabelle 6.6 enthält dazu die konkreten Werte nach (6.19) bzw. (6.20). Für kleine NoC-Größen gilt $d_{avrg,BEAM} < d_{avrg,Mesh}$ (siehe vergrößerter Ausschnitt). Ab einer Kantenlänge von $k = 4$ gilt $d_{avrg,BEAM} > d_{avrg,Mesh}$. Dazu ist auf der sekundären Y-Achse die Differenz $d_{avrg,BEAM} - d_{avrg,Mesh}$ dargestellt, welche letztendlich gegen 1 strebt. D. h., für große NoC-Dimensionen gilt $d_{avrg,BEAM} \approx d_{avrg,Mesh} + 1$. In einer BEAM-Topologie der Kantenlänge k ist die durchschnittliche Pfadlänge somit um einen Hop größer als in k-fachen 2-Würfeln. Da die Latenz eine Funktion von d ist (siehe z. B. Formel (6.14)), ist auch t_{avrg} größer. Ein zentrales Ziel von BEAM ist jedoch die Verringerung der durchschnittlichen Pfadlänge d_{avrg}, um die zeitintensive Setup-Phase von HSM zu verkürzen. Dies erfolgt beim BEAM-Ansatz durch die *Einsparung von Routern*. Eine Einsparung von Routern ist prinzipiell möglich, da durch die hohe CRR_{BEAM} auch bei verringertem k annähernd gleich viele IP-Cores in das NoC integriert werden können wie in k-fachen 2-Würfeln mit größerer Kan-

tenlänge (siehe Tabelle 6.5). Dieser Zusammenhang ist im Diagramm in Abbildung 6.21b dargestellt, welches d_{avrg} über der Anzahl der IP-Cores aufträgt. Es ist erkennbar, dass für die gleiche Anzahl von IP-Cores $d_{avrg,BEAM} < d_{avrg,Mesh}$ gilt. Dieser wichtige Aspekt wird im Weiteren noch öfter aufgegriffen. Für einen direkten Vergleich von BEAM-Topologien mit k-fachen 2-Würfeln müssen aus diesem Grund Varianten herangezogen werden, welche sich in der Kantenlänge k um mindestens Eins unterscheiden. Die folgenden Betrachtungen beziehen sich der Einfachheit wegen jedoch auf $\Delta_k = 1$. Bei einer Kantenlänge von $3 \leq k \leq 12$ eines k-fachen 2-Würfels gilt dann $d_{avrg,BEAM}(k-1) < d_{avrg,Mesh}(k)$. Somit *muss* ebenfalls gelten $t_{avrg,BEAM}(k-1) < t_{avrg,Mesh}(k)$. Die Intention des folgenden Abschnitts ist es, diesen analytischen Zusammenhang simulativ nachzuweisen.

Simulation der Latenz und des Durchsatzes Den Simulationen dieses Abschnitts liegen NoC-Topologien auf Basis eines mesochronen Taktschemas und des in Abschnitt 6.3 vorgestellten Hybriden Switching-Mechanismus zugrunde, d. h., die Phasendifferenzen $\Delta_\varphi(CLK_i, CLK_j)$ zwischen den regionalen Takten innerhalb der jeweiligen NoC-Infrastruktur sind unbekannt. Die absolute Frequenz beträgt $f_{NoC} = 250$ MHz ($T_{clk} = 4$ ns).



(a) k-fache 2-Würfel, $2 \leq k \leq 5$, $n_{flits} = 10$ **(b)** BEAM, $2 \leq k \leq 5$, $n_{flits} = 10$

Abbildung 6.22.: Durchschnittliche Latenz t_{avrg} für verschiedene Kantenlängen

Genutzt wird zunächst ein uniformes zufälliges Verkehrsmuster. Die Diagramme in Abbildung 6.22 zeigen den Verlauf der durchschnittlichen Latenz für verschiedene NoC-Topologien in Abhängigkeit von der Eingangsdatenrate und Kantenlänge k bei konstanter Paketgröße ($n_{flits} = 10$). Sie weisen den typischen Verlauf der Latenz nach Abbildung 6.19 auf. Der Anfangsbereich bei geringen Eingangsdatenraten ist zusätzlich vergrößert dargestellt. In Abbildung 6.22a sind die Simulationsergebnisse k-facher 2-Würfel dargestellt, in Abbildung 6.22b für BEAM-Topologien.

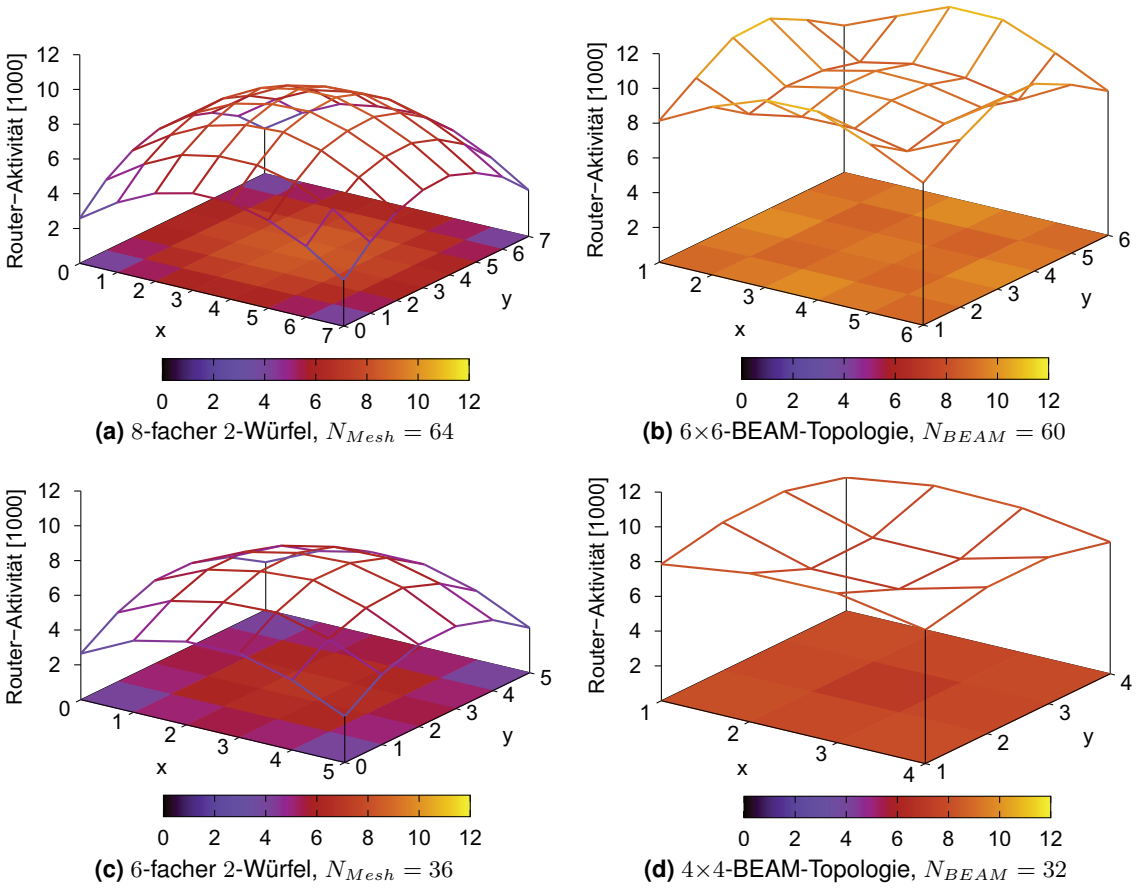
Aus den Kurven ist erkennbar, dass t_{avrg} bei geringen Eingangsdatenraten ungefähr linear mit Kantenlänge k steigt. Dieses Verhalten lässt sich durch den Einfluss der ebenfalls zunehmenden durchschnittlichen Pfadlänge d_{avrg} erklären und soll am Beispiel $k = 3$ mit den analytischen Werten untermauert werden. Nach Gleichung[7] (6.14) ergeben sich mit d_{avrg} aus Tabelle 6.6 die oberen und unteren Grenzen von $t_{0,avrg}$ eines 3-fachen 2-Würfels bzw. einer 3×3-BEAM-Topologie zu 124 ns $\leq t_{0,avrg,Mesh} \leq$ 140 ns bzw. 122,56 ns $\leq t_{0,avrg,BEAM} \leq$ 138,24 ns. In den vergrößerten Ausschnitten der Abbildungen 6.22a und 6.22b sind diese Limits mit gestrichelten Linien veranschaulicht. Solange das jeweilige NoC noch nicht gesättigt ist, befindet sich t_{avrg} bei $k = 3$ aufgrund des uniformen zufälligen Verkehrsmusters jeweils innerhalb dieses Bereichs. Mit sinkender Eingangsdatenrate nähert sich t_{avrg} an die untere Grenze von $t_{0,avrg}$ asymptotisch an. Dies gilt ebenso für die übrigen Simulationsdurchläufe mit anderen Kantenlängen.

$$\lceil \Theta_{S,Mesh} \rceil = \frac{2 \cdot BW_B}{N_{Mesh}} = \frac{4 \cdot k \cdot BW_{Channel}}{k^2} = \frac{4}{k} \cdot BW_{Channel} \tag{6.25}$$

$$\lceil \Theta_{S,BEAM} \rceil = \frac{2 \cdot BW_B}{N_{BEAM}} = \frac{4 \cdot k \cdot BW_{Channel}}{4 \cdot k + k^2} = \frac{4}{4 + k} \cdot BW_{Channel} \tag{6.26}$$

Während t_{avrg} mit zunehmendem k steigt, sinkt hingegen die Grenze des Sättigungsbereichs Θ_S. Nach Einsetzen von (6.16) und (6.18) sowie (6.22) und (6.23) in Gleichung (6.24) ergibt sich der ideale Durchsatz für k-fache 2-Würfel bzw. BEAM-Topologien zu (6.25) bzw. (6.26). Für beide Topologien hat $\lceil \Theta_S \rceil$ eine Komplexität von $\mathcal{O}(n^{-1})$, weswegen der Sättigungsdurchsatz Θ_S mit wachsender Kantenlänge k sinkt. Da $N_{Mesh}(k) < N_{BEAM}(k)$ ist, jedoch $BW_{B,Mesh}(k) = BW_{B,BEAM}(k)$ ist, gilt $\Theta_{S,BEAM}(k) < \Theta_{S,Mesh}(k)$. Dies bedeutet, dass durch die höhere CRR in BEAM-Topologien pro IP-Core ein geringerer Anteil an der Gesamtbandbreite BW_{NoC} bzw. an der Bisektionsbandbreite BW_B zur Verfügung steht. Aufgrund des zufälligen gleichverteilten Verkehrsmusters und der daraus resultierenden hohen Anzahl an Blockierungen ist der Sättigungsbereich in BEAM-Topologien früher erreicht als in k-fachen 2-Würfeln *gleicher* Kantenlänge. Für $k = 5$ z. B. liegt $\Theta_{S,Mesh}$ bei ungefähr 9 %, während $\Theta_{S,BEAM}$ lediglich ca. 4 % beträgt. Werden jedoch Topologien verglichen, die sich in k unterscheiden ($1 \leq \Delta_k \leq 2$), aber ungefähr die gleiche Anzahl an IP-Cores besitzen ($N_{BEAM}(k - \Delta_k) \approx N_{Mesh}(k)$), so ist festzustellen, dass auch $\Theta_{S,BEAM}(k - \Delta_k) \approx \Theta_{S,Mesh}(k)$ ist. In Abbildung 6.22 ist dies z. B. an den Kurvenverläufen eines 5-fachen 2-Würfels ($N_{Mesh} = 25$) und einer 3×3-BEAM-Topologie ($N_{BEAM} = 21$) zu erkennen. Beide Topologien besitzen ungefähr dieselbe Anzahl von IP-Cores und weisen einen ähnlichen Sättigungsdurchsatz von ca. 10 % auf. Ähnliches gilt z. B. für einen 3-fachen 2-Würfel ($N = 9_{Mesh}$) und eine 2×2-BEAM-Topologie ($N_{BEAM} = 12$). Θ_S beträgt hier für

[7] Formel (6.14) gilt für k-fache 2-Würfel und BEAM-Topologien, da in beiden Fällen HSM zum Einsatz kommt.

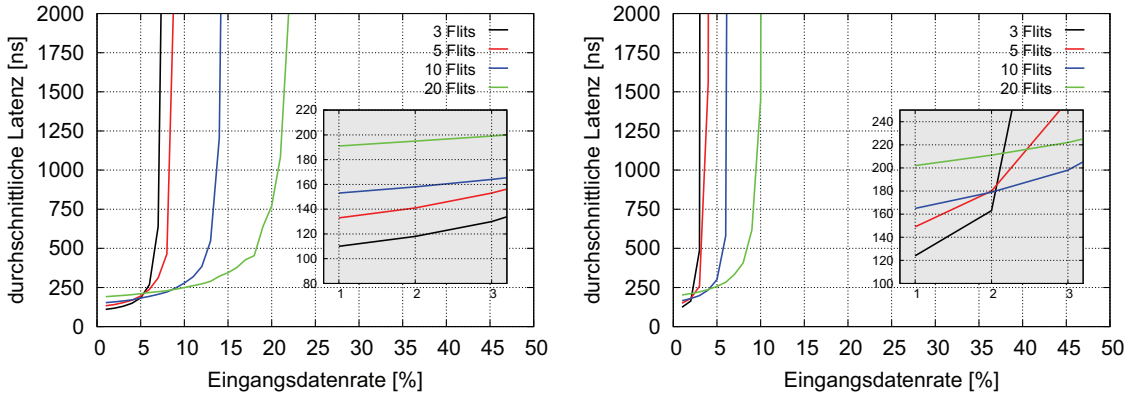


(a) 8-facher 2-Würfel, $N_{Mesh} = 64$ **(b)** 6×6-BEAM-Topologie, $N_{BEAM} = 60$

(c) 6-facher 2-Würfel, $N_{Mesh} = 36$ **(d)** 4×4-BEAM-Topologie, $N_{BEAM} = 32$

Abbildung 6.23.: Verteilung der Router-Aktivität in k-fachen 2-Würfeln und BEAM-Topologien

beide Topologien ca. 20 %. Dieses Verhalten hat zwei Ursachen. Einerseits resultiert eine ähnliche Anzahl von IP-Cores bei einem zufällig gleichverteilten Verkehrsmuster auch in ungefähr der gleichen Gesamtdatenlast, welche die NoC-Infrastruktur zu bewältigen hat. Andererseits besitzt eine BEAM-Topologie mit kleinerer Kantenlänge zwar sowohl eine reduzierte Bisektionsbandbreite BW_B als auch eine geringere Gesamtbandbreite BW_{NoC}, jedoch werden die vorhandenen Kommunikationsressourcen in einer BEAM-Topologie *effizienter* ausgenutzt. Um dies zu belegen, wurde die sogenannte Aktivität der NoC-Router für verschiedene Topologien simulativ ermittelt. Die Router-Aktivität beschreibt, wie oft die Prozesse der Wegewahl und Arbitrierung in einem Router angestoßen werden. Für jedes der an den Eingangsports der Router eintreffenden Pakete muss genau einmal die Wegewahl erfolgen. Simuliert wurde jeweils über einen Zeitraum von 5000 µs bei 10 % Eingangsdatenrate. Die Ergebnisse sind in Abbildung 6.23 illustriert. Die X-



(a) 4-facher 2-Würfel, $n_{flits} = 3, 5, 10, 20$ **(b)** 4×4-BEAM, $n_{flits} = 3, 5, 10, 20$

Abbildung 6.24.: Durchschnittliche Latenz t_{avrg} für verschiedene Paketgrößen

und Y-Achsen repräsentieren die Adressen der NoC-Router. In den Arbeiten von Sofke und Daum [Sof07, Dau08] wurde gezeigt, dass k-fache 2-Würfel auf Basis von XY-Routing bei zufällig gleichverteiltem Datenaufkommen eine Konzentration des Datenverkehrs in der Mitte und eine deutliche Unterauslastung der Kommunikationsressourcen am Rand der Topologie aufweisen. Dieses Verhalten ist auch in den Abbildungen 6.23a und 6.23c sichtbar. Im Gegensatz dazu ist in den Abbildungen 6.23b und 6.23d das Verhalten von BEAM-Topologien mit jeweils einer ähnlichen Anzahl von IP-Cores unter denselben Simulationsbedingungen dargestellt. Die Auslastung der Kommunikationsressourcen, insbesondere am Rand der Topologie, ist deutlich höher als in k-fachen 2-Würfeln. Die zusätzlichen IP-Cores am Rand einer BEAM-Topologie erzeugen in diesen Bereichen eine höhere Last, stellen jedoch gleichzeitig Datensenken dar und konsumieren Pakete. Dadurch weisen BEAM-Topologien einerseits eine höhere und andererseits eine gleichmäßig verteilte Auslastung auf. Die Effizienz der NoC-Infrastruktur steigt. Dies ist wiederum der Grund dafür, dass Θ_S beider Topologien trotz geringerer Gesamt- und Bisektionsbandbreite der BEAM-Topologien ähnlich groß ist.

Die Abbildungen 6.24a und 6.24b zeigen die Abhängigkeit von t_{avrg} und Θ_S von der Paketgröße. Die Kantenlänge ist konstant ($k = 4$), um nur die Auswirkungen der steigenden Paketgröße kenntlich zu machen. Steigt n_{flits} muss $t_{0,avrg}$ nach Formel (6.14) ebenfalls ansteigen. Dieser Einfluss ist in den vergrößerten Diagrammausschnitten erkennbar, da sich auch hier t_{avrg} an $t_{0,avrg}$ annähert. Gleichzeitig mit n_{flits} steigt auch Θ_S. Dies ist einerseits darauf zurückzuführen, dass die Effizienz des Flusskontrollverfahrens mit steigender Paketgröße ebenfalls zunimmt. Der relative Anteil der zeitintensiven Setup-Phase von HSM an der gesamten Übertragungsdauer eines Pakets nimmt mit zunehmender Paketgröße ab (siehe Abschnitt 6.3.2). Andererseits befin-

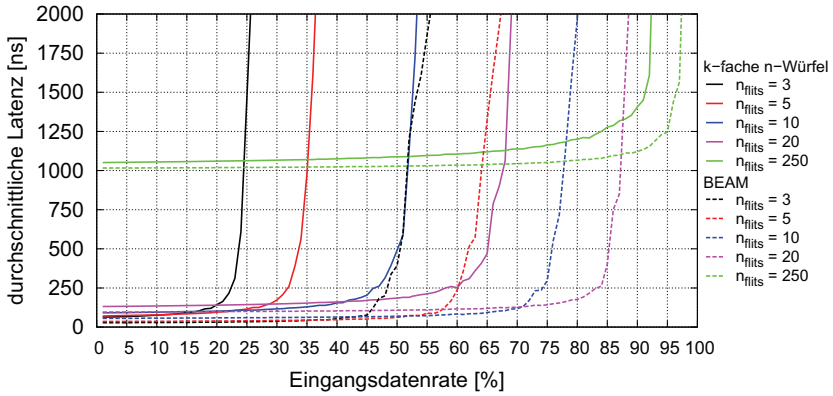


Abbildung 6.25.: Durchschnittliche Latenz t_{avrg} bei blockierungsfreiem Verkehrsmuster

den sich bei großen Paketlängen weniger Pakete gleichzeitig innerhalb der NoC-Infrastruktur als bei geringeren Paketlängen (bei annähernd identischer Eingangsdatenrate). Es kommt deshalb weniger häufig zu konkurrierenden Anfragen auf dieselben Kommunikationsressourcen (Puffer und Kanäle) und damit zu Blockierungen. Dadurch sinkt wiederum die Anzahl der Wartezyklen $n_{blocked}$. Für die Abbildungen 6.24a und 6.24b gilt $\Theta_{S,BEAM} \approx \frac{1}{2} \cdot \Theta_{S,Mesh}$, da bei $k = 4$ gilt $N_{BEAM} = 2 \cdot N_{Mesh}$ (siehe Tabelle 6.5).

In Kontrast zu den Simulationen mit zufällig gleichverteiltem Datenaufkommen stellt Abbildung 6.25 eine Reihe von Simulationen vor, in denen ein blockierungsfreies Verkehrsmuster genutzt wird. Dieses Muster ist an das sogenannte Neighbor Traffic Pattern [DT03] angelehnt, welches ursprünglich in Ringstrukturen genutzt wird, z. B. in Tori, um die Kommunikationsleistung zwischen direkt benachbarten IP-Cores zu analysieren. Die Adresse der Senke berechnet sich dabei aus der Adresse der Quelle ($DST_x = (SRC_x + 1) \ mod \ k$). Dies verhindert, dass sich unabhängige Datenströme gegenseitig blockieren können. Die genutzten Pfade in k-fachen 2-Würfeln haben dann stets eine Pfadlänge von $d = d_{min,Mesh} = 2$. In BEAM-Topologien werden nur Pfade der Länge $d = d_{min,BEAM} = 1$ und $d = 2$ genutzt. Abbildung 6.25 zeigt Θ_S sowie den Verlauf von t_{avrg} für beide Topologien. Die Paketgröße wurde variiert ($n_{flits} = 3, 5, 10, 20, 250$). Da jeweils nur die kürzesten Pfadlängen genutzt werden, sind Θ_S und t_{avrg} unabhängig von k. Bei Eingangsdatenraten im Bereich von 0 bis ca. Θ_S nähert sich t_{avrg} an $t_{0,min}$ an und liegt exakt innerhalb des durch Formel (6.14) mit d_{min} ermittelbaren Bereichs. Verglichen mit den Simulationen mit zufällig gleichverteiltem Verkehrsmuster ist eine erhebliche Steigerung von Θ_S für beide Topologien zu erkennen, was auf die Blockierungsfreiheit zurückzuführen ist ($n_{blocked} = 0$). Θ_S ist in diesen Fällen ausschließlich durch die Effizienz des Flusskontrollverfahrens bestimmt. Mit steigender Paketgröße verschiebt sich Θ_S in Richtung der maximalen

Tabelle 6.7.: Verhältnis der Leistungsparameter k-facher 2-Würfel und BEAM-Topologien

Verkehrsmuster	zufällig gleichverteilt	blockierungsfrei
durchschn. Pfadlänge d_{avrg}	$d_{avrg,BEAM}(k-1) < d_{avrg,Mesh}(k)$	für $3 \leq k \leq 12$
durchschn. lastfreie Latenz $t_{0,avrg}$	$t_{0,avrg,BEAM}(k-1) < t_{0,avrg,Mesh}(k)$	für $3 \leq k \leq 12$
durchschn. Latenz t_{avrg}	$t_{avrg,BEAM}(k-1) \lessapprox t_{avrg,Mesh}(k)$	$t_{avrg,BEAM} < t_{avrg,Mesh}$
Sättigungsdurchsatz Θ_S	$\Theta_{S,BEAM}(k-\Delta_k) \approx \Theta_{S,Mesh}(k)$ für $N_{BEAM}(k-\Delta_k) \approx N_{Mesh}(k)$	$\Theta_{S,BEAM} > \Theta_{S,Mesh}$

Eingangsdatenrate, da einerseits die Übertragungsdauer von HSM eine geringe Wachstumskomplexität besitzt ($\Theta(n_{flits})$) und andererseits der Routing-Overhead im Vergleich zur Paketgröße an Einfluss verliert (vgl. Formel (6.13)). Bei einer Paketlänge von 250 Flits kann in k-fachen 2-Würfeln z. B. ein maximaler Durchsatz von ca. 85 % der physikalischen Kanalbandbreite $BW_{Channel}$ erzielt werden, in BEAM-Topologien sogar ca. 95 %. Mit Blick auf Formel (6.13) ist zu erkennen, dass für beide Topologien der Routing-Anteil der Latenz durch die kurzen Pfade minimiert wird. Für BEAM-Topologien entfällt darüber hinaus der Synchronisationsanteil für Pfade der Länge $d = d_{min,BEAM} = 1$ gänzlich. Somit gilt für ein blockierungsfreies Verkehrsmuster $t_{avrg,BEAM} < t_{avrg,Mesh}$ und $\Theta_{S,BEAM} > \Theta_{S,Mesh}$, wie die Simulationen in Abbildung 6.25 belegen.

Die diskutierten analytischen und simulativen Ergebnisse beziehen sich auf rein synthetische Verkehrsmuster, welche allein dem allgemeinen Vergleich und der Erläuterung der Zusammenhänge und internen Prozesse beider Topologien dienen. Die wichtigsten der in diesem Abschnitt herausgestellten Relationen verschiedener Parameter von BEAM-Topologien und k-fachen 2-Würfeln sind in Tabelle 6.7 zusammengefasst. Die beiden genutzten Verkehrsmuster stellen Extremfälle dar. Zwischen synthetischen und realen Verkehrsmustern bestehen jedoch deutliche Unterschiede [VM04, OHM05]. Blockierungen können z. B. durch die entsprechende Platzierung von IP-Cores innerhalb der NoC-Infrastruktur minimiert bzw. verhindert werden, um den blockierungsfreien Fall anzunähern. Dazu wird jedoch auf Kapitel 7 dieser Arbeit verwiesen.

Syntheseergebnisse Die Synthese erfolgte unter den gleichen Bedingungen wie für HSM in Abschnitt 6.3.3. Die Synthesewerte werden mit denen k-facher 2-Würfel verglichen. Beide NoC-Varianten sind aus funktionaler Sicht identisch (siehe u. a. Abschnitt 6.2.2) und unterscheiden

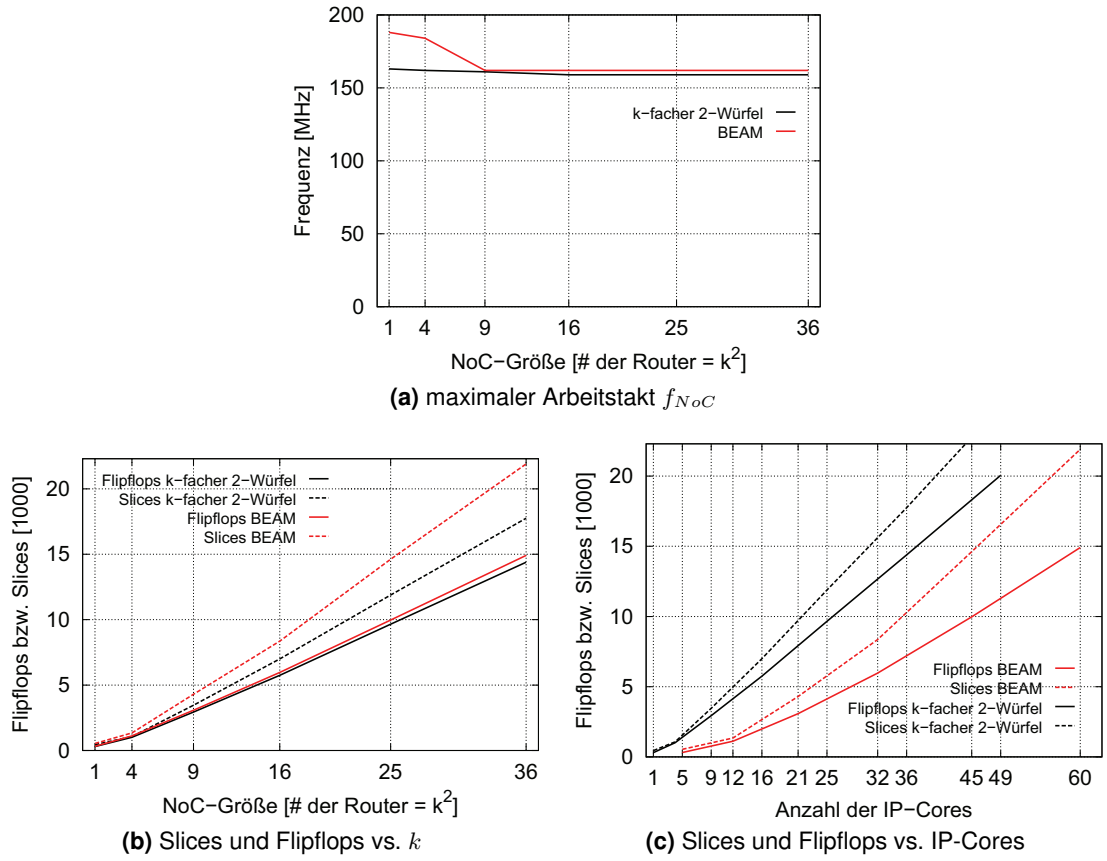
sich ausschließlich in ihrer Topologie.

In Abbildung 6.26a ist die maximale Arbeitsfrequenz f_{NoC} über der NoC-Größe aufgetragen. Durch die mesochrone Architektur und Nutzung des in Abschnitt 6.3 vorgestellten HSM ist f_{NoC} auch in BEAM-Topologien unabhängig von der NoC-Größe und allein durch die Komplexität eines einzelnen Routers bestimmt. f_{NoC} beträgt ähnlich wie im k-fachen 2-Würfel ungefähr 160 MHz. Dies gilt an dieser Stelle vorerst für die Werte nach der reinen Synthese des VHDL-Codes auf dem Ziel-FPGA. Nach dem Vorgang des Platzierens und Verdrahtens (Place & Route) *kann* die absolute Frequenz jedoch von den genannten Werten abweichen, da die tatsächlichen Leitungsverzögerungen mit in die Berechnung einfließen. Dieser endgültige Syntheseschritt ist für die größeren NoC-Infrastrukturen (ab ca. $k = 3$) jedoch nicht möglich, da das FPGA nicht über ausreichend freie I/O-Pins verfügt. Nicht verbundene Komponenten werden in diesen Fällen durch die Optimierungsprozesse des Synthesetools entfernt, so dass die Ergebnisse nicht repräsentativ sind.

Jedoch zeigen sich Unterschiede im Hardwarebedarf, insbesondere bei gleicher Kantenlänge k. Dazu stellt Abbildung 6.26b den Ressourcenverbrauch über der NoC-Größe dar. Eine BEAM-Topologie der Kantenlänge k benötigt mehr Slices als ein k-facher 2-Würfel. Dies ist durch die Anpassungen des Routing-Algorithmus an die BEAM-Topologie bedingt. Die Modifikationen des XY-Routings (siehe Abschnitt 6.4.2 und C.2) erfordern zusätzliche Logik in Form von Slices zur Umsetzung der Funktionalität. Bzgl. der Anzahl der Flipflops ist der Ressourcenbedarf für k-fache 2-Würfel und BEAM-Topologien ungefähr gleich, da die Modifikationen keine zusätzlichen Registerelemente erfordern.

Ein weiterer zentraler Aspekt des BEAM-Ansatzes ist jedoch, Ressourcen in Form von Routern einzusparen und gleichzeitig eine hohe Anzahl von IP-Cores in einem NoC-basierten SoC integrieren zu können (siehe Tabelle 6.5). Aus diesem Grund müssen – ähnlich wie bei der Diskussion der Simulationsergebnisse – Topologien miteinander verglichen werden, deren Kantenlängen k sich um mindestens Eins unterscheiden ($\Delta_k \geq 1$). In Abbildung 6.26b ist ersichtlich, dass z. B. eine BEAM-Topologie mit $k = 5$ ca. 15000 Slices benötigt. Ein 6-facher 2-Würfel erfordert hingegen ungefähr 17500 Slices. Somit können mit einer BEAM-Topologie in etwa 2500 Slices eingespart werden. Gleichzeitig erlaubt der BEAM-Ansatz die Integration von mehr bzw. zumindest gleich vielen IP-Cores in ein NoC. Dies rechtfertigt den geringen Mehrbedarf an Logik für einen einzelnen an die BEAM-Topologie angepassten Router.

Vergleich der BEAM-Topologie mit existenten Ansätzen In der Literatur sind verschiedene von k-fachen 2-Würfeln abgeleitete Topologien zu finden. Einerseits haben diese die Verringerung der durchschnittlichen Latenz und die Durchsatzsteigerung zum Ziel. Andererseits sollen die Hardwarekosten bzw. der Energieverbrauch reduziert werden.



(a) maximaler Arbeitstakt f_{NoC}

(b) Slices und Flipflops vs. k

(c) Slices und Flipflops vs. IP-Cores

Abbildung 6.26.: Syntheseergebnisse verschiedener k-facher 2-Würfel und BEAM-Topologien

In [Dal91] stellt Dally mit *Express Channels* ein allgemeines Konzept zur Verringerung der Latenz in Kommunikationsnetzen vor. In einem Express-Cube – ein k-facher n-Würfel auf Basis von Express Channels – existieren parallel zu den Direktverbindungen zwischen benachbarten Routern noch zusätzliche direkte Kanäle zu jedem m Schritte entfernten Router. Durch die Einbindung von Express Channels wird die Bisektionsbandbreite einer Gittertopologie erhöht. *Long-range Links* stellen eine anwendungsspezifische Variante von Express Channels dar. Marculescu et al. [HM05a, OMLC06, OM06] verwenden diese Methode zur punktuellen Modifikation der NoC-Topologie für eine bestimmte Applikation. Jedoch wird dadurch die Regularität und Allgemeinheit der Kommunikationsinfrastruktur reduziert. Beide Konzepte erfordern Maßnahmen zur Anpassung des Zeitverhaltens der langen Signalleitungen der Express-Kanäle. Wesentlicher Nachteil dieser Ansätze ist jedoch der hohe Grad der Router, da zusätzliche Router-Ports benötigt werden. Dadurch steigen der Hardwarebedarf und die Komplexität der Schaltmatrix eines

Routers an, wodurch f_{NoC} und die physikalische Bandbreite wiederum reduziert werden. Als Abhilfe stellen Kumar et al. in [KPKJ08] mit *Express Virtual Channels* ein vergleichbares Konzept auf virtueller Ebene vor. Der Vorteil ist, dass die Anzahl der physikalischen Ports eines Routers nicht erhöht wird. Jedoch setzt dieser Ansatz das Vorhandensein virtueller Kanäle einschließlich zusätzlicher Paket-Puffer und der entsprechenden Arbitrierungslogik voraus.

Ein der BEAM-Topologie ähnliches Konzept stellt die Klasse der *Concentrated Meshes* dar [BD06, GMB+08], welche ebenfalls eine $CRR > 1$ aufweist. In einem Concentrated Mesh sind die Router in einem regelmäßigen Gitter angeordnet. Jedoch sind an *jeden* Router m IP-Cores angeschlossen. Während die CRR bei BEAM mit zunehmender NoC-Größe gegen 1 strebt, gilt in einem Concentrated Mesh $CRR = m$, unabhängig von k. Durch den direkten Anschluss von m IP-Cores an einen Router werden insgesamt weniger Router benötigt und die durchschnittliche lastfreie Latenz der Topologie sinkt. Gleichzeitig sinkt die Bisektionsbandbreite, da weniger Übertragungskanäle im NoC vorhanden sind. Zudem ist abhängig von m das flache Adressierungsschema eines typischen k-fachen n-Würfels nicht ohne Weiteres anwendbar. Nach Bertozzi et al. [BMF08] ist jedoch primär der hohe Grad der Router der limitierende Faktor einer Concentrated Mesh-Topologie, da die Komplexität der Router die tatsächlich erreichbare maximale Arbeitsfrequenz des NoC und damit den Leistungsgewinn wieder begrenzt. In einem Concentrated Mesh mit CRR = 4 sind z. B. Router mit 8 Ports notwendig. In einer BEAM-Topologie ist der Grad der Router konstant 5. Um den Hardwarebedarf eines einzelnen Routers möglichst gering zu halten, verlagern Bertozzi et al. die Komplexität der zusätzlichen Ports in die Schnittstelle zum NoC und schlagen als Kompromiss zwischen Kommunikationsleistung und Hardwarebedarf ein zeitgemultiplextes RNI vor (Network Interface Sharing) [BMF08]. Mehrere IP-Cores sind an demselben Router-Port angeschlossen und teilen sich gemeinsam dessen physikalische Bandbreite.

Einen weiteren interessanten Ansatz zur Reduzierung der durchschnittlichen Latenz beschreiben Park et al. mit einem *Distributed Multipoint Network Interface* [PNK+06]. IP-Cores sind dabei innerhalb eines k-fachen 2-Würfels mit einer verteilten Netzwerkschnittstelle gleichzeitig an die jeweils vier benachbarten Router angeschlossen. Auf diese Weise besitzt ein IP-Core bis zu vier verschiedene Eintrittspunkte in das NoC und jeder Router ist gleichzeitig mit vier IP-Cores verbunden. Die CRR ist dadurch etwas größer als 1. Die Nachteile dieses Ansatzes sind jedoch die komplexere Netzwerkschnittstelle sowie wiederum die Erhöhung der Anzahl der Router-Ports und damit der Komplexität eines einzelnen Routers.

Die genannten alternativen Lösungsansätze lassen sich prinzipiell in zwei verschiedene Klassen einteilen. Einerseits werden ausgehend von einem einfachen k-fachen 2-Würfel zusätzliche Kommunikationsressourcen in Form von Übertragungskanälen, Pufferspeichern oder Schnittstellen in ein NoC integriert. Diese Ansätze erhöhen zumeist den Grad der Router und vergrößern

in jedem Fall den Hardware-Overhead der Kommunikationsinfrastruktur. Andererseits wird versucht, Kommunikationsressourcen einzusparen, z. B. Router und Router-Ports. Für eine höhere CRR mit dem Ziel einer besseren Auslastung der vorhandenen Kommunikationsressourcen wird dadurch ebenfalls der Grad der Router erhöht bzw. wird die Komplexität in die Netzwerkschnittstellen verlagert. Die in diesem Abschnitt vorgestellte BEAM-Topologie ist jedoch u. a. durch das Simplizitätsprinzip motiviert (siehe Abschnitt 6.2.1 und 6.4). Das Ziel ist, die CRR zu erhöhen, jedoch primär um den Hardware-Overhead der NoC-Infrastruktur zu verringern und ohne dabei die Komplexität einzelner NoC-Komponenten zu erhöhen. Deswegen skaliert CRR_{BEAM} nicht, sondern nähert sich mit zunehmender NoC-Größe an 1 an (siehe Tabelle 6.5).

Darüber hinaus existieren neben k-fachen n-Würfeln verschiedene andere Topologien. Dazu gehören u. a. Tori, Bäume und Butterfly-Strukturen [DT03, DYN03, Tut06]. Ein k-facher 2-Torus ist ein k-facher 2-Würfel mit direkter Verbindung der jeweils gegenüberliegenden Kanten einer Dimension. Dadurch sinken der Durchmesser und die durchschnittliche Pfadlänge bei doppelter Bisektionsbandbreite verglichen mit einem k-fachen 2-Würfel. Einerseits besteht jedoch bei einem Torus die Gefahr von Deadlocks aufgrund zirkulärer Verbindungen. Andererseits ist die Abbildung einer Torus-Struktur auf einen zweidimensionalen IC nicht trivial, da die Kantenverbindungen lange Signalleitungen erfordern, die über den gesamten IC verlaufen. Durch den Einsatz von Repeatern und Pipelining muss das Zeitverhalten angepasst werden. Alternativ kann ein verschachtelter Torus [DS86] genutzt werden. Bäume und Butterfly-Strukturen weisen insbesondere eine geringe und skalierbare durchschnittliche Latenz bei hoher Bisektionsbandbreite auf. Der grundlegende Nachteil dieser Strukturen liegt jedoch in der Abbildung der komplexen und unterschiedlich langen Signalleitungen auf die Oberfläche eines zweidimensionalen ICs. Dazu stellen Kim et al. einen *Flattened Butterfly* als optimierte zweidimensionale Anordnung eines Butterflys vor [KBD07, KDA07]. Zur Realisierung der zweidimensionalen Anordnung werden ein Konzentrationsfaktor von vier ($CRR = 4$) sowie komplexe 10-Port-Router genutzt. Zudem existieren nach wie vor lange Signalleitungen, welche nicht unmittelbar benachbarte Router miteinander verbinden. Pipelining und zusätzliche Repeater sind notwendig, um das Zeitverhalten dieser langen Leitungen zu regulieren.

Zusammenfassung der Eigenschaften von BEAM Der BEAM-Ansatz ist eine nahezu kostenfreie Modifikation einer NoC-Topologie auf Basis k-facher 2-Würfel. Die zusätzlichen Hardwarekosten pro Router beziehen sich allein auf die Anpassung des XY-Routing-Algorithmus an die neuen Gegebenheiten im Randbereich einer BEAM-Topologie und sind aufgrund der Einsparung von Routern vernachlässigbar. Der Kompromiss, der mit der Verwendung einer BEAM-Topologie eingegangen wird, ist die Reduzierung der Bisektionsbandbreite BW_B und der Gesamtbandbreite BW_{NoC} eines NoC, da durch die Einsparung von Routern auch Übertra-

gungskanäle im Randbereich wegfallen. Im Folgenden sind die wesentlichen Charakteristika des BEAM-Konzepts zusammengefasst.

- Die CRR einer BEAM-Topologie ist stets größer als 1. Ist R konstant, können mehr IP-Cores in das NoC integriert werden. Ist N konstant, können Router eingespart werden. CRR_{BEAM} ist jedoch nicht konstant, sondern strebt mit zunehmender NoC-Größe gegen 1 (siehe Tabelle 6.6). Deshalb ist der BEAM-Ansatz insbesondere für praktikable NoC-Größen in FPGAs im Bereich $2 \leq k \leq 10$ geeignet.

- Durch die hohe CRR ergibt sich eine geringe durchschnittliche Pfadlänge d_{avrg}. Dies ist insbesondere für HSM von Vorteil, da früher aus der zeitineffizienten Setup-Phase in die Fast-Transmit-Phase gewechselt werden kann (vgl. Abschnitt 6.3.2). Abhängig vom Verkehrsmuster und der Häufigkeit von Kollisionen bzw. Blockierungen reduziert dies ebenfalls die durchschnittliche Latenz t_{avrg} (siehe Tabelle 6.7).

- Aufgrund der hohen Anzahl von IP-Cores (CRR > 1) kann der Sättigungsdurchsatz Θ_S insbesondere bei Verkehrsmustern mit häufigen Blockierungen deutlich geringer sein als in k-fachen 2-Würfeln. Jedoch existieren erhebliche Unterschiede zwischen synthetischen und realen bzw. anwendungsspezifischen Verkehrsmustern. Dazu sei auf das nächste Kapitel verwiesen.

- Abhängig vom Datenaufkommen und Verkehrsmuster werden die Kommunikationsressourcen in BEAM-Topologien effizienter ausgenutzt. Die Anpassungen des XY-Routing-Algorithmus an den BEAM-Ansatz resultieren in einer gleichmäßig verteilten Router-Aktivität (siehe Abbildung 6.23).

- Die Regularität und Einfachheit von k-fachen 2-Würfeln und XY-Routing bleiben erhalten. Um eine CRR > 1 zu erreichen, wird der Grad der Router und damit ihre Komplexität nicht erhöht, wodurch f_{NoC} konstant bleibt (siehe Abbildung 6.26a).

- Durch die Einsparung von Routern werden die Hardwarekosten der NoC-Infrastruktur gesenkt (siehe Abbildung 6.26b).

- Die Entwicklung der NoC-Architektur erfolgte zwar vor allem mit Blick auf FPGAs als Ziel-Plattform, jedoch kann die NoC-Architektur ebenso in ASIC-basierten SoCs verwendet werden. Aufgrund der Ressourceneffizienz des BEAM-Konzeptes kann insbesondere in ASICs durch die Senkung der Hardwarekosten (= Fläche sowie Anzahl der Transistoren) auch der statische Leistungsverbrauch[8], z. B. durch Leckströme, reduziert werden. Der

[8] Der dynamische Leistungsverbrauch ist hingegen durch die Aktivität und Auslastung der Komponenten der NoC-Infrastruktur bestimmt, welche wiederum von der Applikation und deren Kommunikationsmuster abhängen.

statische Leistungsverbrauch besitzt bei anhaltender Skalierung der Strukturgrößen einen zunehmend relevanten Anteil am Gesamtleistungsverbrauch. Für weitere Informationen dazu sei jedoch auf die Literatur verwiesen, z. B. [Gra05], [Sil07], [WH05] und [RCN03].

6.5. Zusammenfassung des Kapitels

Im Bereich der Chip-internen Kommunikation zeichnet sich eine ähnliche Entwicklung ab, wie sie bereits in der Einleitung der Arbeit in Abbildung 1.1 dargestellt wurde. Im Bereich der Telekommunikation hat sich der Nutzwert von Netzwerken und Diensten stetig gesteigert (Rundfunknetze ⇒ leitungsvermittelte Sprach- und Datennetze ⇒ paketvermitteltes Internet). Der Wandel der Kommunikationsparadigmen von dedizierten Punkt-zu-Punkt-Verbindungen über geteilte synchrone Bussysteme zu asynchronen und modularen Networks-on-Chip ist ein Abbild dessen im SoC-Bereich. Abbildung 6.27 illustriert diese Parallelen. Beide Bereiche profitieren von der Vernetzung der „Teilnehmer“.

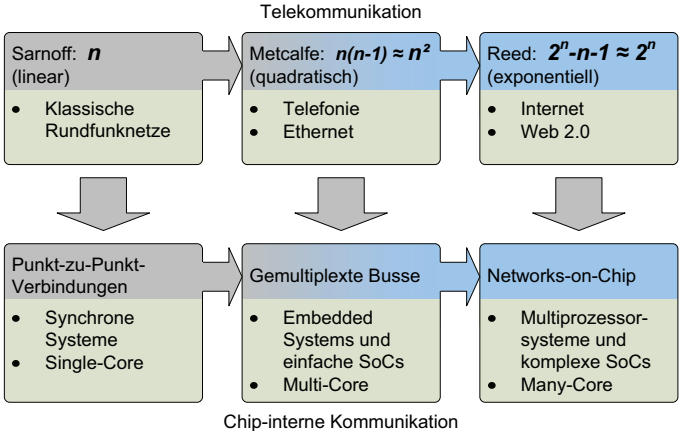


Abbildung 6.27.: Entwicklungsstufen Chip-interner Kommunikationsstrukturen in direktem Vergleich zu Entwicklungen im Telekommunikationsmarkt (n = Anzahl der Teilnehmer)

In Abschnitt 6.1 wurden verschiedene Ansätze für Verbindungs- und Kommunikationsstrukturen für SoCs vorgestellt. Primär aufgrund der unzureichenden Skalierbarkeit in verschiedenen Bereichen sind klassische Ansätze jedoch nicht mehr für zukünftige komplexe SoCs und deren Anforderungen geeignet (siehe Abbildung 6.7). Deshalb wurde das vielversprechende Konzept der NoCs als skalierbare und flexible Lösung für die Chip-interne Verdrahtung und Kommunikation eingeführt. NoCs leiten sich ähnlich wie makroskopische Datennetze vom

OSI-Referenzmodell ab und sind in verschiedene Abstraktionsebenen eingeteilt, welche einen umfangreichen Parameterraum mit unzähligen Entwurfsoptionen aufspannen.

In diesem Kapitel wurde in Anlehnung an das Simplizitätsprinzip die Entwicklung einer schlanken skalierbaren NoC-Architektur beschrieben und diskutiert. Der Einsatzbereich des entwickelten NoC bezieht sich auf Aufgaben der Paketverarbeitung im Bereich der Telekommunikation in feldprogrammierbaren integrierten Schaltkreisen. Ausgehend von einem einfachen NoC in Abschnitt 6.2 wurden mit HSM (Abschnitt 6.3) und BEAM (Abschnitt 6.4) zwei Optimierungsansätze bzgl. des Flusskontrollverfahrens der Vermittlungsschicht und der Topologie vorgestellt, die den gestellten Anforderungen und Rahmenbedingungen des Einsatzumfelds genügen. HSM ist ein effizientes Flusskontrollverfahren für ein mesochron angesteuertes NoC. BEAM erlaubt die Integration einer hohen Anzahl an IP-Cores in ein NoC und reduziert gleichzeitig die Anzahl von Routern, ohne den Grad der Router zu erhöhen und damit f_{NoC} zu senken. Es ist eine allgemeine und schlanke NoC-Architektur entstanden, welche für verschiedene Aufgaben und Projekte als Verbindungs- und Kommunikationssubsystem (wieder)verwendet werden kann. Ergebnisse dieser Arbeiten konnten z. T. in [KWHT07, KCHT07, KHT07, Kub08] veröffentlicht werden. Im folgenden Kapitel 7 wird das in Abschnitt 4.3 vorgestellte MATMUNI-System auf das entwickelte NoC abgebildet. Beide Architekturkonzepte – eine synchrone gepipelinete Struktur und ein asynchrones NoC-basiertes System – werden gegenübergestellt.

Fat pipes are the best QoS.
(David Newman, Data Communications, 16.01.1998)

Kapitel 7.

Anwendungsabbildung und Systemvergleich

Kapitelstruktur

Das NoC-Paradigma ist u. a. durch eine strikte Trennung von Kommunikation und Verarbeitung charakterisiert. Aus diesem Grund stellte Kapitel 6 eine angepasste NoC-Architektur vor, ohne sich dabei auf eine konkrete Applikation zu beziehen. In diesem Kapitel wird die Anwendbarkeit und Tauglichkeit dieser Architektur anhand der Abbildung einer Anwendung aus dem Bereich der Paketverarbeitung in der Telekommunikation untersucht. Dazu wird das in Abschnitt 4.3 bereits vorgestellte MATMUNI-System genutzt, da es mit seinen vernetzten Hardware-Komponenten ein qualifiziertes Beispiel eines SoC darstellt und verschiedene typische Arten von IP-Cores beinhaltet: I/O-Schnittstellen, Speichermodule und Recheneinheiten. Im weiteren Verlauf wird zunächst die formale Vorgehensweise der Anwendungsabbildung erläutert. Analyse und Abbildung von MATMUNI auf die entwickelte NoC-Architektur werden in Abschnitt 7.2 beschrieben. Danach wird das NoC-basierte MATMUNI-System in Abschnitt 7.3 evaluiert und dem Referenzsystem aus Abschnitt 4.3 gegenübergestellt.

7.1. Formale Systemnotation und Problemformulierung

Motivation Die topologische Zuordnung bzw. Abbildung einer konkreten Anwendung auf eine Kommunikationsarchitektur wird im Bereich digitaler integrierter Systeme allgemein als *Mapping* bezeichnet. Die einzelnen logischen Komponenten einer Applikation und deren Kommunikationsbeziehungen untereinander werden den physikalischen Kommunikations- und Verarbeitungsressourcen einer bestimmten Architektur zugewiesen. Dies erfolgt zumeist unter dem Gesichtspunkt einer konkreten Zielvorgabe bzw. definierter Randbedingungen. Die Minimierung des Energieverbrauchs bzw. der Latenz, eine ausgewogene Auslastung der Ressourcen oder das Einhalten von Echtzeitbedingungen sind dabei typische Optimierungsziele [MM04, HM05c, ACN+07].

Für klassische Kommunikationsstrukturen wie Busse (siehe Abschnitt 6.1.1) spielte das Mapping bisher eine untergeordnete Rolle. Aus Sicht der Kommunikationsleistung ist es prinzipiell irrelevant, wo ein Teilnehmer, Prozessor oder IP-Core an einen Bus angeschlossen ist, da ein Bus ein gemeinsam genutztes zeitgemultiplextes Kommunikationsmedium ist. Zur Leistungsoptimierung werden z. B. spezielle Transaktionsarten genutzt oder Buszugriffe mittels Prioritäten arbitriert. Ähnliches gilt für Punkt-zu-Punkt-Verbindungen, da die Kommunikation ohnehin über exklusive Kanäle erfolgt. In einem NoC-basierten SoC spielt das Mapping hingegen eine wesentliche Rolle und zählt zu den Verfahren mit den größten Optimierungspotentialen [MOZ06], da das Verkehrsmuster durch die topologische Platzierung direkt beeinflusst wird. Ein NoC ist eine verteilte Kommunikationsressource (vgl. Abschnitt 6.1.2), welche aus einer Vielzahl unabhängiger Router und separater „Busse“ (= Kanäle) besteht. Diese erlauben die simultane Übertragung mehrerer Pakete. NoCs sind somit sowohl zeit- als auch raumgemultiplext. Dadurch ist es möglich, durch eine angepasste Partitionierung und Platzierung unabhängiger Funktionen einer Anwendung z. B. die Leistungsfähigkeit des Gesamtsystems zu optimieren, da Blockierungen innerhalb der NoC-Infrastruktur und somit Wartezyklen in den einzelnen Funktionsmodulen reduziert werden. Ein wesentlicher Vorteil ist, dass die zugrunde liegende Kommunikationsinfrastruktur durch diese anwendungsspezifische Optimierung nicht verändert wird. Dadurch wiederum kann das NoC allgemein, einfach und mit dem Ziel, BW_{NoC} zu maximieren, gestaltet werden, wie es in Kapitel 6 dieser Arbeit beschrieben wurde.

Formale Notation Zur formalen Darstellung von Applikationen und Architekturen werden in der Literatur ein Vielzahl ähnlicher Notationen vorgeschlagen [HM05b, OHM05, HM05c, MM04, WCL05]. Im Weiteren werden die folgenden formalen Darstellungsformen verwendet:

Communication Task Graph Ein Communication Task Graph (CTG) symbolisiert eine Anwen-

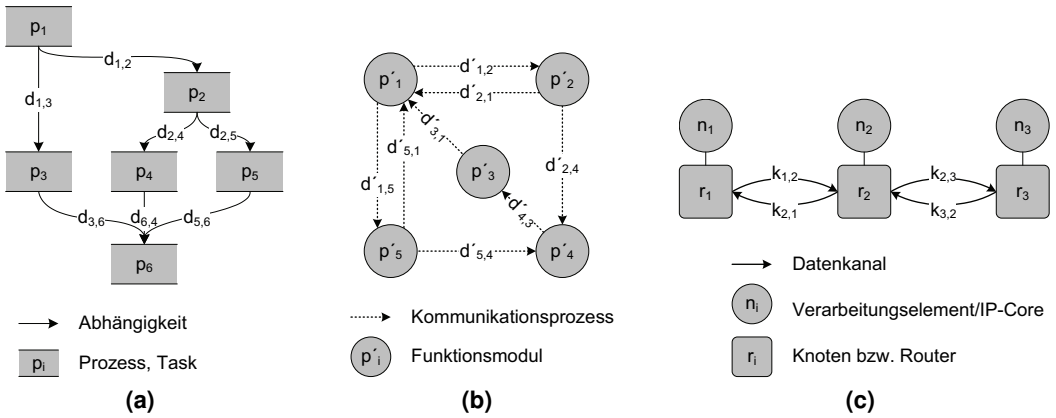


Abbildung 7.1.: Beispielgraphen der formalen Notation: (a) CTG (b) APCG (c) ARCG

dung auf Prozessebene und ist als gerichteter azyklischer Graph $\mathcal{G}(P, D)$ definiert. P ist die Menge aller Prozesse bzw. Teilaufgaben p_i der Anwendung. D ist die Menge aller Abhängigkeiten $d_{i,j}$ zwischen den Prozessen p_i und p_j. Unter einer Abhängigkeit wird dabei entweder ein Austausch von Informationen oder eine Kontrollfunktion bzw. Ablaufsteuerung verstanden. Jede $d_{i,j} \in D$ kann mit Parametern, z. B. dem jeweiligen Kommunikationsvolumen, annotiert sein. Abbildung 7.1a skizziert einen einfachen CTG.

Application Characterization Graph Ein Application Characterization Graph (APCG) leitet sich durch logische Partitionierung aller $p_i \in P$ aus dem CTG einer Anwendung ab und ist als Graph $\mathcal{G}'(P', D')$ definiert. P' ist die Menge aller unabhängigen abgeschlossenen Funktionsmodule p'_i. Dabei werden logisch verwandte bzw. zusammenhängende Prozesse des CTG in autarken Funktionsmodulen des APCG zusammengefasst. Ein APCG ist i. Allg. grobkörniger als ein CTG, jedoch können beide Notationen identisch sein. D' ist die Menge aller Kommunikationsprozesse $d'_{i,j}$ zwischen den Funktionsmodulen p'_i und p'_j. Ein Kommunikationsprozess $d'_{i,j}$ kann wiederum mit verschiedenen anwendungsbezogenen Parametern ausgezeichnet bzw. gewichtet sein. Abbildung 7.1b zeigt einen APCG.

Architecture Characterization Graph Ein Architecture Characterization Graph (ARCG) beschreibt eine NoC-Infrastruktur und ist als Graph $\mathcal{G}(N, R, K)$ definiert. In Abbildung 7.1c ist als Beispiel der ARCG eines 3-fachen 1-Würfels dargestellt. N ist die Menge aller IP-Cores bzw. physikalischen Verarbeitungselemente n_i im NoC. R ist die Menge aller Knoten bzw. Router r_i. Ist CRR > 1 (siehe Formel (6.15)), können an einen Router auch mehrere IP-Cores angeschlossen sein. K ist die Menge aller physikalischen unidirektionalen Übertragungskanäle $k_{i,j}$ zwischen den Routern r_i und r_j.

Ein konkretes Mapping ist mit $\mathcal{M}$ bezeichnet. Gesucht wird ein $\mathcal{M}$, welches den APCG einer Anwendung auf den ARCG einer Kommunikationsinfrastruktur unter Berücksichtigung eines Optimierungsziels $\mathcal{Z}$ abbildet:

$$\mathcal{M} : \mathcal{G}'(P', D') \overset{max(\mathcal{Z})}{\longmapsto} \mathcal{G}(N, R, K).$$

Dies erfolgt durch Platzierung aller $p'_i \in P'$ auf die verfügbaren $n_i \in N$. Diese Abbildung ist nur durchführbar, wenn gilt $|\mathcal{G}'(P', D')| \leq |\mathcal{G}(N, R, K)|$. Dass bedeutet, die NoC-Infrastruktur muss ausreichend Kommunikations- und Verarbeitungsressourcen in Form von IP-Cores und Bandbreite für die entsprechende Applikation bieten. Für einfache Anwendungen mit überschaubarem APCG kann ein den Randbedingungen genügendes $\mathcal{M}$ per Hand ermittelt werden. Hingegen erfolgt das Mapping für komplexe SoCs abhängig von $\mathcal{Z}$ und vom Parameterraum zumeist auf algorithmischer Ebene, z. B. mithilfe evolutionärer Algorithmen [ACP05].

7.2. Analyse und Abbildung des MATMUNI-Systems

In diesem Abschnitt wird das MATMUNI-System anhand der eingeführten Notationen beschrieben. Das MATMUNI-System kann zwar mit bis zu vier bidirektionalen GbE-Kanälen konfiguriert sein, jedoch wird aus Gründen der Übersicht und Einfachheit das Mapping im Folgenden nur anhand eines einzelnen bidirektionalen GbE-Kanals durchgeführt.

7.2.1. Abhängigkeitsanalyse und Partitionierung

Abbildung 7.2a skizziert den CTG des MATMUNI-Systems. An dieser Stelle stehen die Beziehungen der einzelnen Aufgaben und Prozesse untereinander im Vordergrund. Für weitere Details zu den Aufgaben und zur Funktionsweise von MATMUNI sei auf Abschnitt 4.3 bzw. auf die Grundlagen zur Paketverarbeitung in Abschnitt 2.3 verwiesen. Ein bidirektionaler GbE-Kanal besteht aus je einem Datenpfad in Up- und Downstream mit einem Datenaufkommen von jeweils ≤ 1 Gbit/s. Der CTG ist für beide Datenflussrichtungen identisch. Es existiert jeweils ein Hauptdatenpfad (rot hervorgehoben) zur Weiterleitung der zu verarbeitenden Nutzdaten (Ethernet-Frames). Parallel dazu erfolgt die Suche nach Regeln im Speicher anhand der aus den Frames extrahierten Schlüssel. Abhängig davon, ob für einen Schlüssel eine Regel gefunden werden konnte oder nicht, wird der entsprechende Frame weiter im Hauptdatenpfad bzw. zur CPU geleitet. Speicher und CPU sind zentral verwaltet, weswegen gleichzeitige Anfragen der Up- und Downstreamkanäle an diesen Ressourcen auftreten können. Diese müssen arbitriert werden. Je nach Datenflussrichtung und Konfiguration können Anordnung und Anzahl der

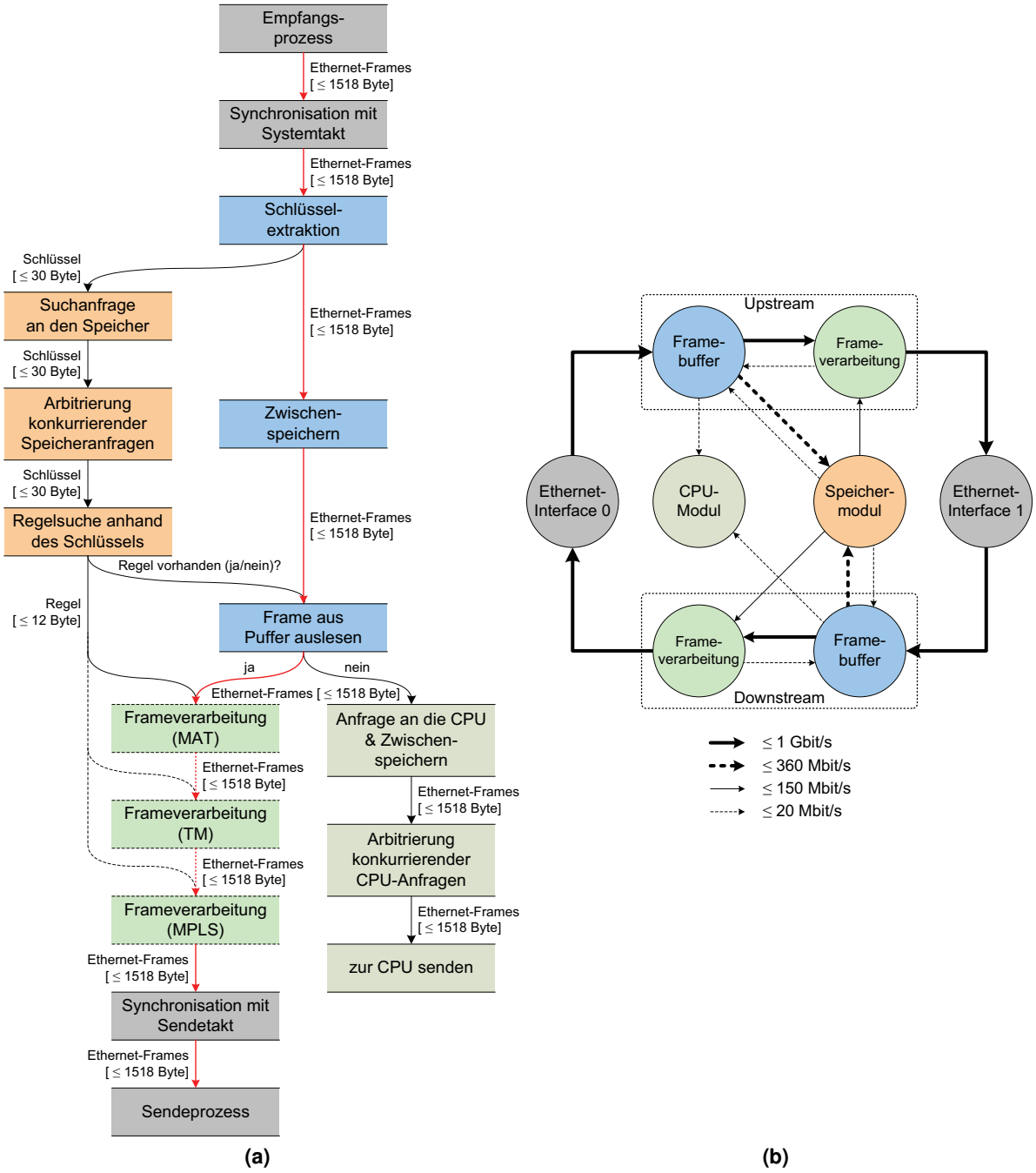


Abbildung 7.2.: (a) Communication Task Graph und (b) Application Characterization Graph des MATMUNI-Systems

Prozesse für die eigentliche Frameverarbeitung (MAT, TM, MPLS) variieren, weswegen diese im CTG gestrichelt dargestellt sind.

Abbildung 7.2b zeigt den APCG des MATMUNI-Systems für einen GbE-Kanal. Dieser leitet sich aus dem CTG ab. Funktional eng miteinander verknüpfte Aufgaben und Prozesse sind in jeweils einem Funktionsmodul zusammengefasst, um die Anzahl der benötigten IP-Cores und damit die erforderliche NoC-Größe zu reduzieren. In den Abbildungen ist dies anhand gleicher Farben symbolisiert.

- Schlüsselextraktion und Pufferung von Frames sind im Framebuffer vereint.
- Die einzelnen Prozesse MAT, TM und MPLS können sowohl aufgrund ihrer seriellen Anordnung als auch aufgrund ihrer geringen Komplexität (vgl. Tabelle 4.1) gemeinsam in einem Modul (Frameverarbeitung) integriert werden.
- Alle auf den Speicher bezogenen Aufgaben sind im Speichermodul kombiniert.
- CPU und CPU-Arbiter sind zusammengefasst.
- Ethernet-Schnittstelle 0 dient als Eingangsport des Upstream- und als Ausgangsport des Downstream-Datenpfads; Ethernet-Schnittstelle 1 in gleicher Weise anders herum.
- Pro bidirektionalen GbE-Kanal sind somit jeweils zwei Ethernet-Schnittstellen, zwei Framebuffer und zwei Frameverarbeitungsmodule nötig.
- Die Zwischenspeicherung von Frames im CPU-Datenpfad und die Taktsynchronisierung entfallen komplett. Diese Aufgaben werden bereits durch die in den NoC-Schnittstellen, den RNIs (siehe Abbildungen 6.3), enthaltenen asynchronen FIFO-Puffer erfüllt.

Im APCG sind die Kommunikationsprozesse mit dem jeweils erforderlichen Bandbreitenbedarf in Form unterschiedlicher Strichstärken annotiert. Dick markierte $d'_{i,j}$ haben einen Bandbreitenbedarf von ≤ 1 Gbit/s und repräsentieren die Hauptdatenpfade des Systems. Die anderen Kommunikationsprozesse besitzen entsprechend der ausgetauschten Informationen einen geringeren Bandbreitenbedarf, der sich auf die mögliche maximale Framerate des GbE-Kanals bezieht. Bei minimal großen Ethernet-Frames (64 Byte) beträgt die maximale Framerate 1488096 Frames/s (siehe Tabelle C.1 im Anhang). Abhängig von der Struktur der Schlüssel kann ein einzelner Suchschlüssel, d. h. pro Frame, bis zu 30 Byte umfassen. Dies ergibt einen Bandbreitenbedarf von ca. 360 Mbit/s für die Kommunikation des Framebuffers mit dem Speichermodul. Für die anderen Kommunikationsprozesse errechnet sich die erforderliche Bandbreite analog.

7.2.2. Mapping von MATMUNI auf ein Network-on-Chip

Für die topologische Abbildung der Funktionsmodule wird eine NoC-Infrastruktur genutzt, welche auf den in Kapitel 6 vorgestellten Ansätzen HSM und BEAM basiert. Das Ziel der Abbildung von MATMUNI auf ein NoC ist einerseits die Minimierung der Latenz der inter-IP-Core-Kommunikation und andererseits die Vermeidung von Konkurrenz bzgl. der gemeinsam genutzten Kommunikationsressourcen des NoC. Aus diesem Grund werden für ein sowohl effektives als auch effizientes $\mathcal{M}$ verschiedene Faktoren und Rahmenbedingungen berücksichtigt:

Verkehrsmuster der Anwendung Die Aufgaben des MATMUNI-Systems beziehen sich auf die Paketverarbeitung in TZN und Internet. Wie Abbildung 2.3 (S. 11) bereits zeigte, ist die Paketverarbeitung i. Allg. durch eine primäre Datenflussrichtung charakterisiert. Diese Eigenschaft spiegelt sich ebenfalls im Blockschaltbild der konventionellen MATMUNI-Architektur in Abbildung 4.12 (S. 58) in Form der Upstream- und Downstream-Datenpfade wieder. Auch CTG und APCG des MATMUNI-Systems zeigen deutlich die Hauptdatenpfade mit einem Bandbreitenbedarf von jeweils ≤ 1 Gbit/s. Die dazu orthogonal verlaufenden Kommunikationsprozesse der Kontrollpfade, z. B. von und zum Speichermodul, sind für jeden Frame identisch. Somit weist die MATMUNI-Anwendung ein sehr reguläres und lineares Verkehrsmuster auf, welches auf dem Hauptdatenpfad durch eine hohe Bandbreite und in den Kontrollpfaden durch wiederkehrende Operationen mit jeweils geringem Datenvolumen geprägt ist.

Ausnutzung von Lokalität Stark miteinander kommunizierende Funktionsmodule sind möglichst nah beieinander zu platzieren, so dass Cluster bzw. sogenannte Kommunikationsinseln entstehen. Nah bedeutet, dass die Kommunikationspfade im NoC zwischen diesen Modulen eine geringe bis minimale Länge d aufweisen, so dass in etwa gilt $d_{min} \leq d \leq d_{avrg}$. Pfade der Länge d_{max} sind zu vermeiden. Dadurch kann der absolute Routing-Overhead der Setup-Phase des HSM minimiert und $t_{0,min}$ angenähert werden. Sind Pfade mit $d > d_{avrg}$ nicht zu vermeiden, sollten primär lange Pakete über diese Pfade versendet werden, da sich der Routing-Overhead bei großen Paketen relativiert (siehe auch Abbildung 6.15).

Minimale NoC-Größe Die Nutzung einer NoC-Architektur impliziert immer einen gewissen zusätzlichen Hardwarebedarf für das NoC-Subsystem. Um diesen Overhead gering zu halten, wurde in Kapitel 6 eine schlanke NoC-Architektur mit angepasster Topologie für Gitter-basierte NoCs eingeführt. Für das Mapping sind die Kantenlängen k_x und k_y der NoC-Topologie unter der Bedingung $|\mathcal{G}'(P', D')| \leq |\mathcal{G}(N, R, K)|$ minimal zu dimensionieren.

Router-Auslastung Ein einzelner NoC-Router ist fähig, bis zu 5 unabhängige Paketübertragungen simultan zu verwalten, solange diese sich nicht gegenseitig blockieren. Da in einer

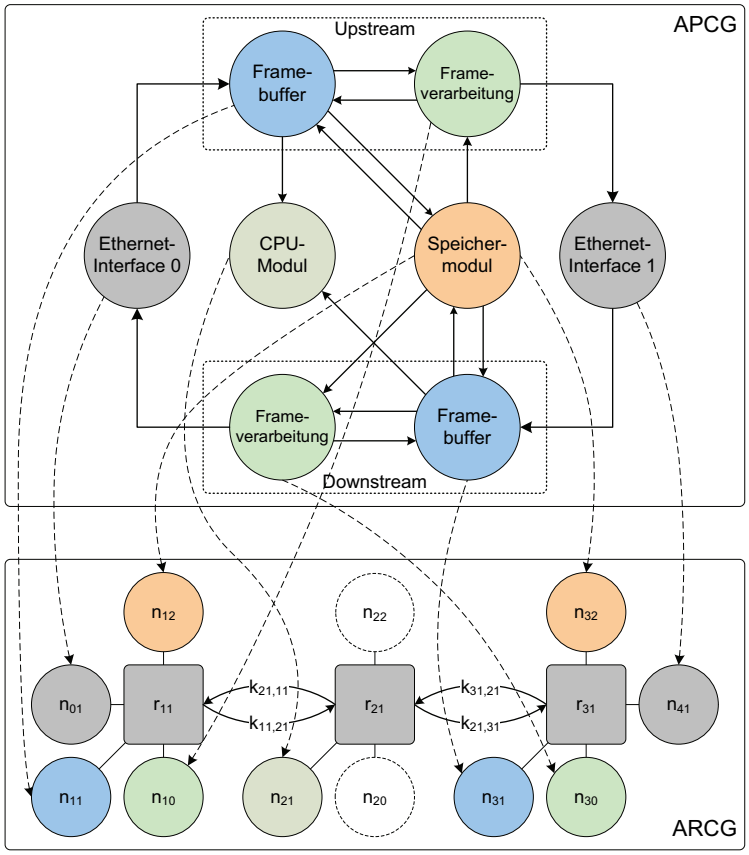


Abbildung 7.3.: Abbildung von MATMUNI auf ein NoC auf Basis einer 3×1-BEAM-Topologie

BEAM-Topologie CRR > 1 ist, können darüber hinaus mehrere IP-Cores an denselben Router angeschlossen sein. Dadurch kann neben der Gesamtbandbreite der einzelnen Übertragungskanäle (= BW_{NoC}) auch die gesamte Router-interne Bandbreite BW_{Router} ausgenutzt werden. Die genutzten 5-Port-Router verfügen über ein maximales internes Bandbreitenreservoir von $BW_{Router} = 5 \cdot BW_{Channel} = 5 \cdot f_{NoC} \cdot W_{bit}$. In einem klassischen k-fachen 2-Würfel kann dies durch die nicht vorgesehenen Ports am Rand der Topologie nicht in allen Routern ausgeschöpft werden. Unter Beachtung der anderen genannten Aspekte und durch gezielte Ausnutzung der minimalen Übertragungswege (d_{min}) kann die Auslastung der Router erhöht werden. Ihre Effizienz steigt (vgl. auch Abbildung 6.23). Gleichzeitig kann die Verkehrslast auf den Übertragungskanälen $k_{i,j} \in K$ reduziert werden, da ein Großteil der inter-IP-Core-Kommunikation innerhalb der einzelnen Kommunikationsinseln stattfindet.

Routing-Algorithmus Zur Vermeidung von Konkurrenz beim Zugriff auf die geteilten Kom-

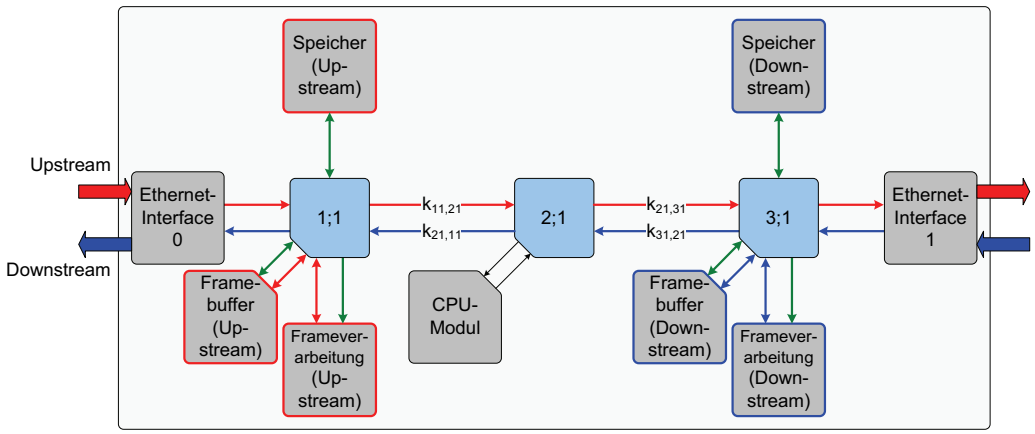


Abbildung 7.4.: Blockschaltbild des NoC-basierten MATMUNI-Systems (vgl. Abbildung 4.12)

munikationsressourcen ist in jedem Fall die Modalität des Routing-Algorithmus in Betracht zu ziehen. Insbesondere bei deterministischen Routing-Algorithmen wie dem verwendeten XY-Routing kann a-priori-Wissen über die tatsächlichen Übertragungswege im NoC mit in die Anwendungsabbildung einfließen.

Mit Bezug auf die genannten Faktoren wurde das Mapping für ein MATMUNI-System mit einem bidirektionalen GbE-Kanal durchgeführt. Aufgrund der überschaubaren Komplexität der Anwendung erfolgte das Mapping per Hand. Abbildung 7.3 zeigt die Zuordnung der einzelnen Funktionsmodule des APCG aus Abbildung 7.2b zu den IP-Cores des ARCG einer NoC-Infrastruktur. Gewählt wurde eine BEAM-Topologie mit den Kantenlängen $k_x = 3$ und $k_y = 1$. Die gestrichelten IP-Cores bleiben ungenutzt.

Abbildung 7.4 zeigt eine detailliertere Struktur des durch das Mapping in Abbildung 7.3 entstandenen NoC-basierten MATMUNI-Systems. Die Teilung des Speicher-Moduls in separate Module für den Up- und Downstream wird in diesem Blockschaltbild deutlich. Diese Teilung ist bereits eine wesentliche Veränderung im Vergleich zum konventionellen MATMUNI-System. Im Zuge dessen Evaluation in Abschnitt 4.3.2 wurden der zentrale Speicher und der Speicherarbiter als limitierende Faktoren sowohl aus Sicht der Leistungsfähigkeit des Systems als auch aus Sicht des maximalen Arbeitstaktes beschrieben. Aus diesem Grund erfolgt an dieser Stelle eine physikalische Trennung des Speichermoduls in zwei separate Teilspeicher für alle Up- bzw. alle Downstream-Datenpfade. Dies reduziert bei mehreren bidirektionalen GbE-Kanälen die Anzahl gleichzeitiger Zugriffe auf einen zentralen Speicher, resultiert aber *nicht* in einer Verdopplung des erforderlichen Speichers, da die Adresstabellen für Up- und Downstream ohnehin unabhängig voneinander zu verwalten sind. Zudem ändert auch eine NoC-basierte Architektur nichts an

der Tatsache, dass ein zentraler Speicher einen Engpass bzgl. der Systemleistung darstellt! Deshalb darf diese Speicheraufteilung nicht als prinzipielle Lösung dieses Engpasses angesehen werden, sondern als reine Systemoptimierung. Aufgrund standardisierter Schnittstellen und einer hohen Modularität erlaubt ein NoC vielmehr eine flexible Systemerweiterung sowie einfache Einbindung und Platzierung zusätzlicher Funktionsmodule. Dies ist ein wesentlicher Vorteil der NoC-basierten Architekturvariante, von welchem an dieser Stelle Gebrauch gemacht wurde.

Abbildung 7.4 zeigt weiterhin, dass die Funktionsmodule derart in den IP-Cores angeordnet sind, dass sich die zwei Hauptdatenpfade nicht gegenseitig blockieren. Up- und Downstream sowie die jeweils zugehörigen IP-Cores sind rot bzw. blau hervorgehoben. Die Übertragungskanäle $k_{11,21}$ und $k_{21,31}$ werden nur vom Upstream-Datenpfad genutzt. Die Kanäle $k_{31,21}$ und $k_{21,11}$ werden ausschließlich vom Downstream-Datenpfad genutzt. Gemäß der Syntheseergebnisse aus Abbildung 6.26a kann das NoC-Subsystem mit einem Arbeitstakt von ca. $f_{NoC} = 160$ MHz angesteuert werden. Demnach besitzt ein einzelner Kanal $k_{i,j}$ eine physikalische Bandbreite von ungefähr 5,12 Gbit/s. Im blockierungsfreien Fall können von dieser maximalen Bandbreite in Abhängigkeit von der Größe der Pakete im NoC ca. 85–95 % effektiv genutzt werden, wie es in Abbildung 6.25 in Abschnitt 6.4.3 anhand von Simulationen nachgewiesen wurde. Die MATMUNI-Anwendung benötigt pro Datenpfad allerdings nur eine Bandbreite, die dem zu erwartenden Datenaufkommen eines GbE-Kanals entspricht. Der effektive Datendurchsatz Θ_{Eth} ist bei Ethernet am größten, wenn maximal große Ethernet-Frames versendet werden. Nach Tabelle C.1 und C.2 im Anhang entspricht dies bei Gigabit Ethernet einer Framerate von 81275 Frames/s und einem effektiven Durchsatz von $\Theta_{Eth} = 984{,}4$ Mbit/s. Innerhalb des NoC kommen durch die Paketierung der zu übertragenden Informationen noch weitere 8 Byte für NoC-Header und Tail-Flit pro Ethernet-Frame hinzu. Damit steigt die Datenlast auf den Kanälen $k_{11,21}$, $k_{21,31}$, $k_{31,21}$ und $k_{21,11}$ auf jeweils maximal 989,6 Mbit/s an. Die Kanäle sind damit jedoch nur in etwa zu einem Fünftel ausgelastet. Für den Fall minimaler Ethernet-Frames liegt die Last mit 809,5 Mbit/s nur bei ca. einem Sechstel der Kanalkapazität des NoC[1].

Die Kommunikationsprozesse in und aus Richtung des jeweiligen Speichermoduls sind in Abbildung 7.4 grün markiert und konkurrieren jeweils mit den Hauptdatenpfaden um die Schnittstellen zu den entsprechenden IP-Cores der Framebuffer und Frameverarbeitungsmodule. Diese Kontrollpfade von und zum Speichermodul benötigen jedoch nur einen geringen Anteil der Kanalbandbreite $BW_{Channel}$, wie der APCG in Abbildung 7.2b zeigt. Dieser Anteil nimmt aufgrund der Paketierung im NoC noch geringfügig zu. Deshalb ist zu erwarten, dass die inter-Modul-Kommunikation innerhalb des NoC-basierten MATMUNI-Systems für das vorge-

[1] Abhängig von den endgültigen Ergebnissen bzgl. der maximalen Frequenz nach der Platzierung und Verdrahtung im FPGA kann dieser Anteil variieren.

stellte Mapping weitestgehend blockierungsfrei abläuft. Dadurch nähern sich die tatsächlichen Kommunikationsverzögerungen in diesem Anwendungsfall an t_0 an.

Anhand des Mappings in den Abbildungen 7.3 und 7.4 kann zudem ein wesentlicher Vorteil der entwickelten BEAM-Topologie gegenüber k-fachen 2-Würfeln verdeutlicht werden. Für die Funktionsmodule des APCG wird nur eine 3×1-BEAM-Topologie mit insgesamt 3 NoC-Routern benötigt. Durch die Teilung des zentralen Speichers in zwei separate Module werden 9 der möglichen 11 IP-Cores in diesem NoC genutzt. Bei Verwendung eines klassischen k-fachen 2-Würfels wäre jedoch mindestens ein 3×3-Gitter mit insgesamt 9 Routern erforderlich. Mithilfe des BEAM-Ansatzes werden somit 6 Router (66 %) und damit wertvolle Logikressourcen des FPGAs eingespart. In Tabelle 7.1 sind zusätzlich die Mindestgrößen von BEAM-Topologien und k-fachen 2-Würfeln für MATMUNI-Systeme mit zwei bzw. vier bidirektionalen GbE-Kanälen gegenübergestellt. N_{BEAM} und N_{Mesh} errechnen sich anhand der Formeln (6.16) und (6.18). Bei einer ähnlichen Vorgehensweise beim Mapping wie im bisherigen Verlauf dieses Kapitels resultieren vier bidirektionale GbE-Kanäle in einem APCG mit 27 Funktionsmodulen, welche mindestens einen 4×7-fachen 2-Würfel benötigen. Jedoch ist eine 3×5-BEAM-Topologie bereits ausreichend, wodurch 13 NoC-Router (ca. 46 %) eingespart werden. Die Tatsache, dass einige der Router-Ports ungenutzt bleiben, ist daher irrelevant. Gleichzeitig weisen Topologien mit geringeren Kantenlängen auch eine geringere durchschnittliche Pfadlänge d_{avrg} und damit eine geringere durchschnittliche Latenz t_{avrg} auf, was sich vorteilhaft auf die inter-IP-Core-Kommunikation auswirkt.

Tabelle 7.1.: Einsparung von Routern für verschiedene Konfigurationen des MATMUNI-Systems: Vergleich von BEAM-Topologien mit k-fachen 2-Würfeln

Anzahl der GbE-Kanäle	Anzahl der Funktionsmodule im APCG	minimale BEAM-Topologie (R_{BEAM}, N_{BEAM})	minimaler k-facher 2-Würfel (R_{Mesh}, N_{Mesh})	Ersparnis (Anzahl der Router)
1	9	3×1 (3, 11)	3×3 (9, 9)	6 (≙ 66 %)
2	15	3×2 (6, 16)	3×5 (15, 15)	9 (≙ 60 %)
4	27	3×5 (15, 31)	4×7 (28, 28)	13 (≙ 46 %)

7.2.3. Wandel des Kommunikationsparadigmas

Die Abbildung des konventionellen MATMUNI-Systems auf eine NoC-Infrastruktur resultiert in einer umfassenden Veränderung des Kommunikationsschemas. Dieser Wechsel ist in Abbildung 7.5 skizziert. Prinzipiell basierte die Kommunikation in der konventionellen Architekturvariante

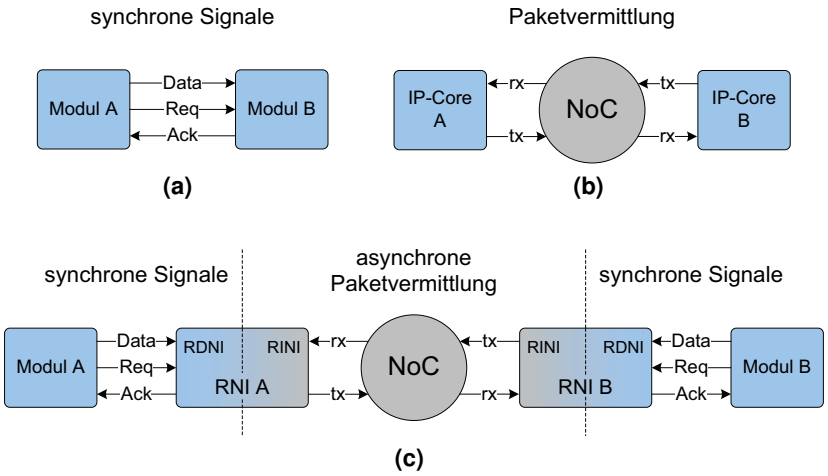


Abbildung 7.5.: Wandel des Kommunikationsparadigmas

auf direkten synchronen Signalen, wie es Abbildung 7.5a zeigt. In einem NoC erfolgt die Kommunikation zwischen zwei IP-Cores jedoch indirekt über ein paketvermitteltes Netzwerk aus Routern, so wie es in Abbildung 7.5b symbolisiert ist. Eine direkte Anbindung der Funktionsmodule an die standardisierten Router-Ports (vgl. Abbildung 6.9c) ist jedoch nicht möglich, da jedes Funktionsmodul eine individuelle Schnittstelle besitzt. Zur Umsetzung dieser spezifischen Schnittstellen auf die NoC-Schnittstelle werden die in Abschnitt 6.1.2 bereits erwähnten RNIs als Brücke zur Anbindung an das NoC genutzt. Dies ist in Abbildung 7.5c skizziert. Das RINI realisiert die NoC-Schnittstelle und das in den NoC-Routern verwendete Flusskontrollverfahren, in diesem Fall HSM (siehe Abschnitt 6.3). Das RDNI ist an das Funktionsmodul angepasst.

Neben der Schnittstellenumsetzung zwischen Funktionsmodulen und NoC erfüllen die RNIs noch weitere zentrale Aufgaben:

Physikalische Trennung Die RNIs sorgen auf Taktebene für eine eindeutige Trennung zwischen dem Kommunikationssubsystem und dem Funktionsmodul. Dazu werden FIFO-Primitive des FPGAs genutzt, welche mit unabhängigen Lese- und Schreibtakten betrieben werden können. Dies erlaubt den Einsatz von Takten mit unterschiedlichen Frequenzen über den mesochronen Anwendungsfall hinaus.

Funktionale Trennung Durch die Abstraktion des Kommunikationssubsystems werden Kommunikation und Verarbeitung voneinander entkoppelt. Das NoC erscheint somit völlig transparent für die eigentliche Anwendung.

Paketierung Für den Ende-zu-Ende-Datenaustausch zwischen den IP-Cores ist es notwendig,

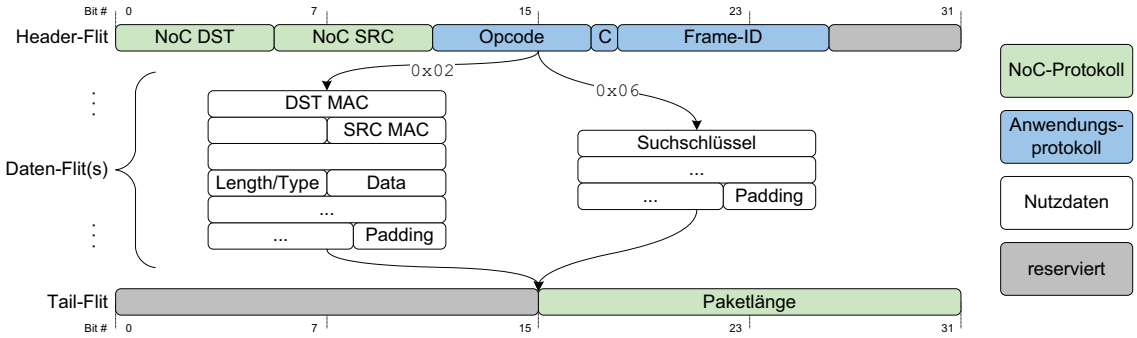


Abbildung 7.6.: Struktur des Kommunikationsprotokolls des NoC-basierten MATMUNI-Systems

die an den Schnittstellen der Funktionsmodule angelegten Informationen in ein Datenformat umzuwandeln, welches über das NoC übertragen werden kann. Dieser Vorgang wird als Paketierung bezeichnet und kapselt die relevanten Nutzinformationen der synchronen Signale in asynchron übermittelten Paketen.

Während die physikalische und funktionale Trennung durch die FIFO-Primitive des FPGAs realisiert wird, erfordert die Paketierung zusätzlichen Aufwand. Der Grund dafür ist, dass Informationen nun nicht mehr an dedizierte Leitungen gebunden sind, welche implizit sowohl Datenquelle und -senke als auch die Art der Informationen definieren. In einem paketvermittelten Netz wie einem NoC können Informationen prinzipiell auf unterschiedlichen Wegen die Datensenke erreichen. Zudem beanspruchen Pakete verschiedener logischer Kommunikationsverbindungen den gleichen physikalischen Kanal. Der Zusammenhang zwischen Datenquelle, Datensenke und Art der Informationen muss deshalb bewahrt werden, was durch die Nutzung definierter Protokolle realisiert wird, welche die Informationen mit einem Kontext versehen. Dadurch sind die Informationen eindeutig zuordenbar und aus Anwendungssicht identifizierbar. Treffende Beispiele für diesen Sachverhalt sind u. a. auch die in den Kapiteln 3 und 5 bereits diskutierten Fragestellungen bzgl. des Übergangs von leitungsvermittelten Sprachnetzen zu IP-basierten paketvermittelten Datennetzen. Auch wenn die Größenordnung an dieser Stelle eine andere ist, gilt trotzdem der gleiche o. g. Zusammenhang.

Für das NoC-basierte MATMUNI-System wird deshalb eine zweistufige Protokollhierarchie verwendet. Diese ist in Abbildung 7.6 dargestellt. Der grundlegende Aufbau eines Pakets entspricht der Einteilung in Header-, Daten- und Tail-Flits aus Abbildung 6.4. Auf *NoC-Ebene* wird ein einfaches NoC-Protokoll zur Identifikation der topologischen Kommunikationsendpunkte, den IP-Cores, genutzt. NoC SRC und NoC DST werden für die Wegewahl benötigt. Im Tail-Flit ist zudem die Gesamtlänge des Pakets enthalten. Das auf der *Anwendungsebene* genutzte Protokoll

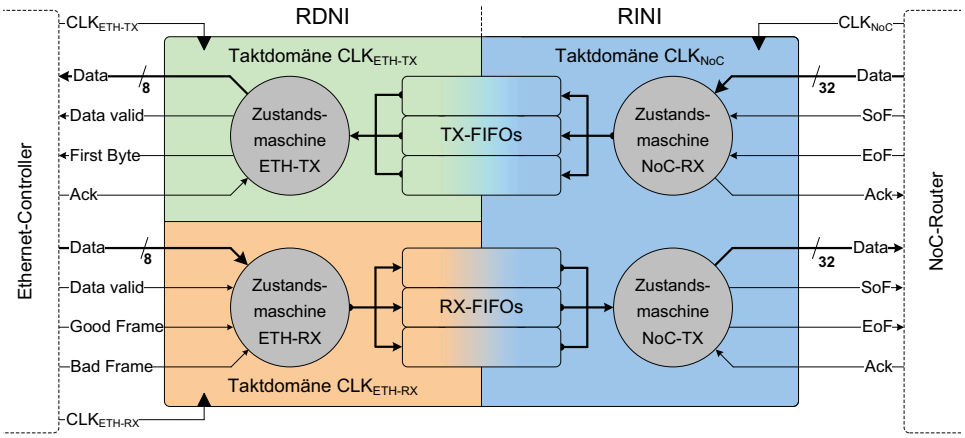


Abbildung 7.7.: RNI einer Ethernet-Schnittstelle im NoC-basierten MATMUNI-System

ist ebenfalls im Header-Flit integriert. Der Opcode definiert die Art der Informationen, welche als Nutzdaten in den Daten-Flits enthalten sind. Zwei Beispiele sind in Abbildung 7.6 dargestellt. Ein Opcode mit dem Wert `0x02` bedeutet, dass ein Ethernet-Frame gekapselt ist. Ist der Opcode gleich `0x06`, ist ein Schlüssel enthalten, mit dem im Speicher nach einer Regel gesucht werden soll. Das C-Flag unterscheidet zwischen Paketen des Upstream- (`0`) und Downstream-Kanals (`1`). In verschiedenen Fällen wird darüber hinaus noch eine zusätzliche Frame-ID genutzt, um z. B. mehrere NoC-Pakete dem gleichen Ethernet-Frame zuordnen zu können. Weitere Informationen zu diesem Anwendungsprotokoll bietet [Her07].

Als Beispiel eines RNI im NoC-basierten MATMUNI-System ist in Abbildung 7.7 das Blockschaltbild des RNI für eine der Ethernet-Schnittstellen gezeigt. Die RNIs der übrigen Funktionsmodule sind in ähnlicher Weise aufgebaut. Der anwendungsunabhängige Teil, das RINI, wird mit dem Takt des jeweiligen NoC-Routers angesteuert. In den Zustandsmaschinen des RINI (NoC-RX, NoC-TX) ist die Kontrolllogik der Empfangs- und Sende-FIFOs sowie das Flusskontrollverfahren des NoC implementiert. Das RINI kann aufgrund der gleichen Funktionalität für die übrigen IP-Cores weitergenutzt werden. Der anwendungsabhängige Teil des RNI, das RDNI, ist in diesem Beispiel zusätzlich in zwei separate Taktdomänen unterteilt, welche jeweils mit dem Sende- und Empfangstakt der Ethernet-Schnittstelle betrieben werden. In den Zustandsmaschinen des RDNI (ETH-RX, ETH-TX) findet einerseits die Kapselung der eintreffenden Ethernet-Frames in NoC-Pakete statt, d. h., NoC-Header und Tail-Flit werden dem Ethernet-Frame voran- bzw. nachgestellt. Andererseits wird der 8 Bit breite Ethernet-Datenbus auf die NoC-Datenkanäle ($W_{bit} = 32$ Bit) umgesetzt bzw. umgekehrt. Für den Übergang von einer synchronen Kommunikation zu einer paketbasierten asynchronen Kommunikation ist für jeden

im APCG der Applikation dargestellten Kommunikationsprozess (vgl. Abbildung 7.2b), an welchem das jeweilige Funktionsmodul als Quelle oder Senke beteiligt ist, eine separate FIFO notwendig.

7.3. Systemevaluation und Vergleich

In diesem Abschnitt erfolgt die Bewertung des NoC-basierten MATMUNI-Systems und ein Vergleich mit der in Kapitel 4.3 dokumentierten Variante als Referenzsystem, welches auf einer synchronen Datenpfadarchitektur basiert. Der Systemvergleich erfolgt anhand des für einen bidirektionalen GbE-Kanal konfigurierten MATMUNI-Systems, dessen Abbildung auf eine NoC-Topologie im vorhergehenden Abschnitt beschrieben wurde. In diesem Zusammenhang sind verschiedene Systemeigenschaften und -parameter von besonderem Interesse:

- die Syntheseergebnisse bzgl. des Ressourcenbedarfs und der Taktfrequenzen
- das Verhalten der Leistungsparameter Latenz und Durchsatz
- Auswirkungen des Wechsels des Kommunikationsparadigmas und der Nutzung der in Kapitel 6 vorgestellten NoC-Architektur

7.3.1. Synthese

Die Synthese des NoC-basierten MATMUNI-Systems erfolgte für das gleiche Ziel-FPGA (Xilinx Virtex-4 FX100 (XC4VFX100-10)) wie für das Referenzsystem in Abschnitt 4.3.2. Tabelle 7.2 vergleicht den Ressourcenverbrauch beider Systeme.

Tabelle 7.2.: Ressourcenbedarf der beiden Architekturvarianten des MATMUNI-Systems

Architekturvariante	Slices	Blockrams	Systemtakt
synchrone Datenpfadarchitektur	3151	24	$f = 126$ MHz
NoC-basierte Architektur	10141	64	siehe Tabelle 7.3

In Abschnitt 6.2.1 wurde bereits erwähnt, dass die Verwendung eines NoC in jedem Fall in einem erhöhten Hardwarebedarf resultiert, was an den Werten in Tabelle 7.2 deutlich zu erkennen ist. Das NoC-basierte System erfordert mit ca. 10141 Slices und 64 Blockrams jedoch ungefähr dreimal so viele Ressourcen wie die konventionelle Architekturvariante mit ca. 3150 Slices und 24 Blockram-Modulen. Die Gründe dafür sind aber nicht in den zusätzlichen NoC-Routern

Tabelle 7.3.: Ressourcenbedarf und erzielbare Taktfrequenzen der RNIs im NoC-basierten MATMUNI-System nach Platzierung und Verdrahtung. Die IP-Cores besitzen einen 8 Bit breiten Datenpfad. Das NoC besitzt 32 Bit breite Übertragungskanäle.

RNI für ...	Slices	Blockrams	$f_{IP-Core}$ / f_{NoC} [MHz]
CPU-Modul	305	1	133 / 133
Ethernet-Schnittstelle 0	720	5	165 / 133
Ethernet-Schnittstelle 1	624	5	170 / 133
Framebuffer Upstream	656	8	131 / 133
Framebuffer Downstream	675	8	131 / 133
Frameverarbeitung Upstream	857	5	152 / 133
Frameverarbeitung Downstream	784	5	175 / 133
Speichermodul Upstream	826	5	191 / 133
Speichermodul Downstream	754	5	193 / 133
Summe	6201	47	-

zu suchen. Einerseits sind die Synthesealgorithmen des Entwurfswerkzeugs und die FPGA-Strukturen auf synchrone Systeme ausgelegt und optimiert. Andererseits spielt die Komplexität der RNIs zur Anbindung der Funktionsmodule an das NoC eine entscheidende Rolle. Der Logikbedarf der einzelnen RNIs ist in Tabelle 7.3 zusammengefasst. Pro RNI sind mehrere Blockram-Module und im Durchschnitt ca. 700 Slices erforderlich. Verglichen mit der Größe der eigentlichen Funktionsmodule (vgl. Tabelle 4.1) sind die RNIs zum Teil deutlich größer. Insgesamt benötigen die RNIs des NoC-basierten MATMUNI-Systems somit ca. 6201 Slices und 47 Blockram-Module. Der Grund für diesen erheblichen Hardware-Overhead ist, dass die Funktionsmodule eigentlich für die synchrone Datenpfadarchitektur des Referenzsystems entworfen wurden. An dieser Stelle werden sie jedoch in einem asynchronen NoC-basierten System nachgenutzt. In den RNIs ist deshalb eine komplette Schnittstellenumsetzung zwischen den Funktionsmodulen des konventionellen MATMUNI-Systems und dem NoC notwendig. Um die korrekte Funktion des Gesamtsystems zu gewährleisten, enthalten die RNIs aufwendige Zustandsautomaten, welche die synchronen Signale auf die paketbasierte Kommunikation im NoC abbilden. Weitere Details zu den verschiedenen Aufgaben eines RNIs wurden bereits in Abschnitt 7.2.3 dargestellt. Wird allerdings bereits im Entwurfsprozess die Nutzung einer NoC-Infrastruktur mit eingeplant, können einzelne Funktionsmodule und damit auch die RNIs in vielen Bereichen effizienter entworfen werden. Dies würde u. a. eine Neugestaltung des gesamten Referenzsystems nach sich ziehen, was jedoch nicht Absicht dieses Kapitels ist.

Gegenüber den RNIs erfordern die drei in diesem System genutzten NoC-Router mit ca. 500 Slices pro Router (siehe z. B. Tabelle 6.3) nur relativ wenig Ressourcen. Darüber hinaus sind

die im konventionellen System genutzten Sync-FIFOs und Control-Module (siehe Abbildung 4.12) in der NoC-basierten Architekturvariante nicht mehr erforderlich und können eingespart werden, da deren Aufgaben – die Taktsynchronisierung mit den Ethernet-Schnittstellen und das Zwischenspeichern von Frames – bereits indirekt durch die NoC-Komponenten erfüllt werden. Pro bidirektionalen GbE-Kanal entfallen damit zwei Sync-FIFOs und zwei Control-Module mit einer Größe von jeweils ca. 187 bzw. 67 Slices (vgl. Tabelle 4.1). Der Mehrbedarf durch die NoC-Router und die Einsparungen durch nicht mehr benötigte Module egalisieren sich somit.

Tabelle 7.3 enthält darüber hinaus die maximal erzielbaren Frequenzen der unabhängigen Taktsignale der NoC-basierten Architektur nach der Platzierung und Verdrahtung auf dem Ziel-FPGA[2]. Diese Frequenzen fließen im weiteren Verlauf mit in die simulative Leistungsbewertung ein. $f_{IP-Core}$ beschreibt die Frequenz des Arbeitstakts der einzelnen Funktionsmodule und des jeweils dazugehörenden RDNI (vgl. Abbildung 7.7). Durch den asynchronen Charakter des NoC-basierten MATMUNI-Systems wird jedes Funktionsmodul mit einem individuellen Takt angesteuert, ohne dass es durch langsamere Funktionsmodule ausgebremst wird oder selbst andere Module ausbremsen kann, wie es im Referenzsystem der Fall ist. Somit sind die einzelnen Module auf Taktebene entkoppelt und werden z. T. mit Taktsignalen deutlich höherer Frequenz angesteuert, als es das synchrone Referenzsystem erlaubt (vgl. Tabelle 4.1). f_{NoC} beschreibt hingegen die Frequenz der zueinander mesochronen Taktsignale der NoC-Router und der RINIs. f_{NoC} liegt mit 133 MHz nach der Platzierung und Verdrahtung ca. 25 MHz unter den Werten der reinen Synthese ohne Platzierung und Verdrahtung, welche bereits in Abbildung 6.26a für unterschiedlich große NoC-Dimensionen dargestellt wurden. Einerseits sind nun konkrete Leitungsverzögerungen des FPGAs in die Analyse des Zeitverhaltens der Schaltung mit einbezogen. Andererseits müssen mit dem jeweiligen Takt eines NoC-Routers auch die an diesen Router angeschlossenen RINIs getrieben werden, welche durch die aufwendigen Brückenfunktionen der RNIs eine hohe Komplexität besitzen. Aufgrund seiner geringen Komplexität ist das CPU-Modul direkt in das RNI integriert und synchron mit f_{NoC} getaktet.

7.3.2. Simulation

Mithilfe des Synthese-Tools konnte nach dem Schritt der Platzierung und Verdrahtung jeweils ein backannotiertes VHDL-Modell des konventionellen und des NoC-basierten MATMUNI-Systems erstellt werden. Diese Modelle sind funktional identisch mit dem originalen VHDL-Code, jedoch wurden sie um die tatsächlichen Leitungsverzögerungen der auf dem FPGA platzierten und verdrahteten Schaltung erweitert. Sogenannte Timing-Simulationen unter nahezu realistischen

[2]Da die Synthese unter Standardeinstellungen durchgeführt wurde, können mithilfe verschiedener Optimierungen ggf. bessere Ergebnisse erzielt werden, z. B. durch die Definition von Synthese-Constraints [XCG].

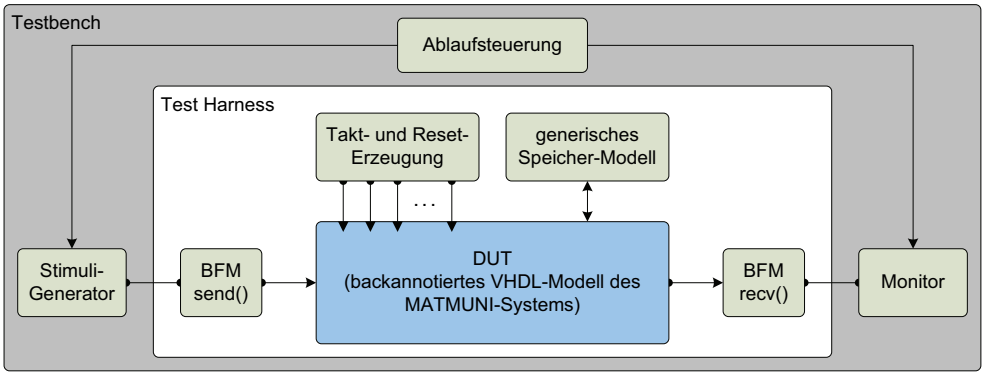


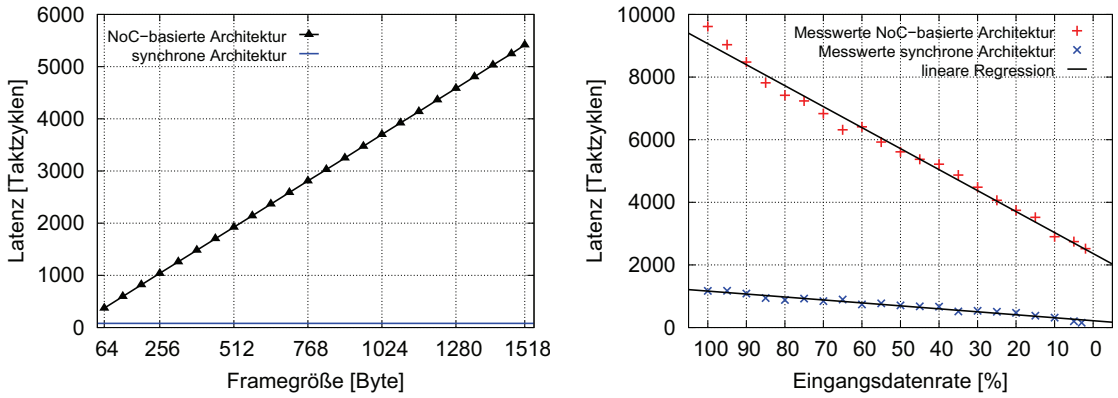
Abbildung 7.8.: Simulationsaufbau

Bedingungen sind damit durchführbar. Die verwendete Simulationsanordnung ist in Abbildung 7.8 dargestellt. Das backannotierte Modell ist als Device-under-Test (DUT) in einem Test Harness eingebunden. Bus Functional Models (BFMs) realisieren die Kommunikation mit dem DUT über dessen Schnittstellen. Im Test Harness werden zudem die entsprechenden Takt- und Reset-Signale zur Ansteuerung des DUT erzeugt. Ein generisches Speichermodell wird zur Emulation des externen Speichers des MATMUNI-Systems genutzt, da die Art des Speichers und der Speicherschnittstelle nicht weiter definiert ist und vom realen Einsatzumfeld der Anwendung abhängt. Die Testbench ist für die Ablaufsteuerung der Simulation verantwortlich, generiert die Stimuli und wertet die Systemantworten aus. Weitere Informationen zum Simulationsaufbau können z. B. [Ber03] entnommen werden.

Eine funktionale Verifikation des MATMUNI-Systems wurde mit den backannotierten VHDL-Modellen durchgeführt. Beide Architekturvarianten erzeugen für die gleichen Stimuli identische Systemantworten. Die äußeren Randbedingungen der Simulation entsprachen dabei den Anforderungen und Spezifikationen für den Betrieb mit Gigabit-Ethernet (vgl. Abschnitt 4.3.2). Während der Simulation wurde das NoC-basierte System mit den durch die Synthese ermittelten Maximalfrequenzen der Taktsignale aus Tabelle 7.3 angesteuert.

Latenz Ein wesentlicher Unterschied besteht in der Latenz der beiden Systemarchitekturen. Die Latenz t ist für das NoC-basierte MATMUNI-System ebenfalls wie für das Referenzsystem laut RFC [RFC1242, RFC2544] definiert (siehe Abschnitt 4.3.2).

Abbildung 7.9a zeigt dazu abhängig von der Framegröße den Verlauf der minimalen Latenz t_0 im lastfreien Fall. Abbildung 7.9b stellt den Verlauf der durchschnittlichen Latenz t_{avrg} über der Eingangsdatenrate für eine reale Verteilung der Framegrößen [SPH05] dar. Zum Vergleich sind zusätzlich die Werte des Referenzsystems aus Abbildung 4.15 in beiden Diagrammen



(a) Minimale lastfreie Latenz t_0 über der Framegröße

(b) durchschnittliche Latenz t_{avrg} über der Eingangsdatenrate

Abbildung 7.9.: Simulation der Latenz des NoC-basierten MATMUNI-Systems

eingezeichnet. Im Gegensatz zur minimalen Latenz des Referenzsystems mit ca. 79 Taktzyklen ist t_0 in der NoC-basierten Architektur einerseits um ein Vielfaches höher und andererseits von der Framegröße abhängig. Auch t_{avrg} ist deutlich höher und beträgt bei 100 % Eingangsdatenrate ca. 10000 Taktzyklen, sinkt jedoch bei geringen Eingangsdatenraten auf bis zu 2000 Taktzyklen ab. Zur näheren Erläuterung dieser Effekte werden die einzelnen Komponenten der Latenz genauer betrachtet. Sie sind in Formel (7.1) zusammengefasst. Die einzelnen Anteile der Gesamtlatenz sind für beide Systemarchitekturen ähnlich (vgl. Formel (4.1)), besitzen jedoch unterschiedlich starken Einfluss auf die Gesamtlatenz und sind in der NoC-basierten Architekturvariante zudem von verschiedenen Taktsignalen abhängig (siehe Tabelle 7.3).

$$t = \overbrace{T_{clk,NoC} \cdot n_{NoC}}^{\text{inter-IP-Core-Komm.}} + \overbrace{t_{CDC}}^{\text{CDC-Verzögerung}} + \overbrace{T_{clk,buf} \cdot n_{buf}}^{\text{Arbitrierungslatenz}} + \underbrace{T_{clk,mem} \cdot n_{mem}}_{\text{Speichersuche}} + \underbrace{T_{clk,func} \cdot n_{func}}_{\text{Frameverarbeitung}} \tag{7.1}$$

- n_{NoC} ist die Anzahl der Taktzyklen, die benötigt werden, um Informationen zwischen den einzelnen IP-Cores über das NoC-Subsystem zu übertragen. Zu diesen Informationen gehören z. B. die Suchschlüssel, die Ergebnisse der Speicher-Lookups und die Ethernet-Frames selbst. n_{NoC} ist dabei abhängig von der Breite der Übertragungskanäle des NoC (W_{bit}), dem Datenvolumen, z. B. der Framelänge L_{Frame}, und von Wartezyklen aufgrund von Blockierungen innerhalb des NoC. Wie in Abschnitt 6.2.1 gefordert wurde, wird

das NoC jedoch nicht an seiner Auslastungsgrenze betrieben. Dadurch und durch ein geschicktes Mapping werden Blockierungen im NoC weitestgehend vermieden. Somit sind vor allem W_{bit} und L_{Frame} ausschlaggebend. Multipliziert mit der Periodendauer des Taktes der NoC-Infrastruktur $T_{clk,NoC}$ ergibt dies die Verzögerung durch die inter-IP-Core-Kommunikation.

- t_{CDC} beschreibt die Verzögerung, die beim Übergang zwischen den verschiedenen Taktdomänen in den RNIs entsteht. Da die Verhältnisse der Frequenzen zueinander unterschiedlich sein können, müssen NoC-Pakete erst komplett in die FIFO-Puffer der RNIs geschrieben werden, bevor sie in der lesenden Taktdomäne aus dem Puffer ausgelesen werden dürfen. Dadurch werden Buffer-Underruns vermieden. t_{CDC} ist dabei selbst eine Summe aus mehreren separaten Verzögerungszeiten, die durch die unterschiedlichen Taktsignale der einzelnen Funktionsmodule bestimmt sind. Der wesentliche Anteil an t_{CDC} wird im Hauptdatenpfad durch die RNIs der Ethernet-Schnittstellen, der Framebuffer und der Frameverarbeitungsmodule hervorgerufen, da dort komplette Ethernet-Frames übertragen werden. t_{CDC} ist somit primär abhängig von L_{Frame}.

- Ähnlich wie in der synchronen Architektur beschreibt n_{buf} die Anzahl an Wartezyklen, die ein Frame im Framebuffermodul warten muss, bis der entsprechenden Suchanfrage durch den Speicherarbeiter Zugriff auf den Speicher gewährt wurde. n_{buf} ist abhängig von der Auslastung der Speichermodule, d. h., von der Anzahl der konkurrierenden Anfragen an den Speicher. Multipliziert mit der Periodendauer des Takts der Framebuffermodule $T_{clk,buf}$ ergibt dies die Arbitrierungslatenz.

- n_{mem} und n_{func} stellen die Anzahl der Taktzyklen dar, welche für die reine Speichersuche bzw. die Frameverarbeitung erforderlich sind. n_{mem} ist abhängig vom Suchschlüssel und der Speichertiefe. n_{func} ist für jeden Frame konstant und abhängig von Art und Zweck der Frameverarbeitungsmodule. $T_{clk,mem}$ und $T_{clk,func}$ sind die Perioden der Takte der Speicher- und Frameverarbeitungsmodule.

Einerseits dominieren die durch die Asynchronität des Gesamtsystems begründeten CDC-Verzögerungen sowie die paketbasierte inter-IP-Core-Kommunikation. Sie sind ausschlaggebend für die Höhe der Gesamtlatenz gegenüber dem MATMUNI-Referenzsystem. Andererseits haben die Verzögerungen durch die eigentliche Informationsverarbeitung innerhalb der einzelnen Funktionsmodule nur einen geringen Anteil an der Gesamtlatenz. Jedoch stellen die Funktionsmodule die Engpässe im NoC-basierten System dar, da sie gegenüber der NoC-Infrastruktur nur einen 8 Bit breiten Datenpfad besitzen und damit eine geringere Bandbreite aufweisen. Es ist

somit anzunehmen, dass die Leistungsfähigkeit des NoC-basierten MATMUNI-Systems nicht durch das NoC-Subsystem, sondern durch die Funktionsmodule in den IP-Cores limitiert ist.

Weiterhin stehen die Gesamtlatenz und die Puffergrößen in direktem Zusammenhang. Die gesamte Zeitdauer, welche Ethernet-Frames innerhalb des MATMUNI-Systems zwischengespeichert werden müssen, bis sie abgearbeitet und weitergeleitet werden können, kann und muss durch eine Anpassung der Pufferkapazitäten sowohl in den Framebuffer-Modulen als auch in den RNIs kompensiert werden. In diesem Zusammenhang wird in der Literatur [BT04] neben der Latenz der Begriff *Backlog* bzw. *Vorhaltezeit* gebraucht, welcher den Arbeitsrückstand des informationsverarbeitenden Systems bezeichnet, d. h. die Menge noch unverarbeiteter Informationen bzw. Frames innerhalb des Systems. Entsprechend des maximalen Backlogs sind die Pufferspeicher zu dimensionieren. Abhängig von der Eingangsdatenrate befinden sich unterschiedlich viele Ethernet-Frames gleichzeitig im MATMUNI-System, deren Anzahl bei maximaler Eingangsdatenrate ebenfalls ihr Maximum erreicht. Die Simulation der durchschnittlichen Latenz t_{avrg} in Abbildung 7.9b zeigt, dass ein Frame bei 100 % Eingangsdatenrate eine Verzögerung von ca. 10000 Taktzyklen erfährt. Für einen verlustfreien Betrieb müssen jedoch auch die zwischenzeitlich empfangenen Daten abgelegt werden können. Da die externen Schnittstellen des MATMUNI-Systems einen 8 Bit breiten Datenpfad besitzen, ergibt sich ein Backlog von ungefähr 10 Kilobyte. Aus diesem Grund wurde der zentrale FIFO-Puffer im Framebuffer-Modul des NoC-basierten MATMUNI-Systems mit einer Kapazität von 2^{14} Byte dimensioniert. Im Referenzsystem ist hingegen eine Puffergröße von 2^{12} Byte ausreichend.

Durchsatz Aufgrund der Tatsache, dass die einzelnen Funktionsmodule im NoC-basierten MATMUNI-System schneller getaktet werden können als im Referenzsystem (vgl. Tabelle 7.2 und 7.3), ist zu erwarten, dass die NoC-basierte Architekturvariante trotz einer höheren Latenz einen höheren Durchsatz erzielt als das Referenzsystem auf Basis einer synchronen Datenpfadarchitektur. In diesem Abschnitt wird deshalb anhand des maximalen Durchsatzes die Leistungsfähigkeit beider Systeme untersucht. Der Durchsatz ist wiederum nach [RFC1242, RFC2544] definiert und in Prozent der maximalen Eingangsdatenrate bzw. absolut in Frames pro Sekunde angegeben.

Für einen fairen Vergleich wird der Durchsatz jeweils nur für einen einzelnen unidirektionalen GbE-Kanal bestimmt, da der zentrale Speicher und Speicherarbiter in der NoC-basierten Variante während der Abbildung auf das NoC in zwei unabhängige Speicher-Module für Up- und Downstream unterteilt wurde (siehe Abschnitt 7.2.2). Wird nur ein einzelner unidirektionaler GbE-Kanal betrachtet, treten in keinem der beiden Testsysteme konkurrierende Zugriffe auf den Speicher auf und es entstehen keine Arbitrierungsverzögerungen. Bei einer Simulation eines oder mehrerer bidirektionaler GbE-Kanäle wäre die NoC-basierte Architektur im Vorteil.

Zur Ermittlung des Spitzendurchsatzes beider Architekturen werden die DUTs in der Test-

bench (siehe Abbildung 7.8) mit einer pro Simulationsdurchlauf zunehmenden Daten- bzw. Framerate stimuliert. Der Durchsatz entspricht der höchsten Eingangsdatenrate, bei der keine Testframes mehr verworfen werden. 100 % Eingangsdatenrate sind dabei als die maximale physikalische Bandbreite BW_{max} definiert, welche die Ein- bzw. Ausgangsports der DUTs zulassen. BW_{max} ergibt sich nach Formel (7.2) aus der Datenbusbreite W_{bit} und der durch die Synthese ermittelten maximalen Taktfrequenzen. Die Schnittstellen des MATMUNI-Systems haben eine Datenbusbreite von $W_{bit} = 8$ Bit (siehe Abbildung 7.7). Die Framerate (FR_{max}) ergibt sich nach Formel (7.3) aus BW_{max} und der Framelänge L_{Frame}. Die Berechnung von FR_{max} unterscheidet sich an dieser Stelle vom Ethernet-Standard dadurch, dass weder Inter Frame Gap und Präambel noch Start Frame Delimiter in Betracht gezogen werden (vgl. Formel (C.1) im Anhang). Zudem werden Ethernet-Frames ohne Frame Check Sequence zum MATMUNI-System gesendet, weswegen in Formel (7.3) zusätzlich 4 Byte von der Framegröße abgezogen werden. Somit kann das Verhalten der DUTs unter größtmöglicher Last simuliert werden. Tabelle 7.4 enthält für beide Systeme die jeweils maximalen Eingangsdatenraten. Das Referenzsystem kann maximal mit einer Taktfrequenz von 126 MHz angesteuert werden, die Eingangsports des NoC-basierten Systems mit maximal 165 MHz (siehe Tabelle 7.2 und 7.3).

$$BW_{max} = f_{max} \cdot W_{bit} \tag{7.2}$$

$$FR_{max} = \frac{BW_{max}}{8 \cdot (L_{Frame} - 4)} \tag{7.3}$$

Tabelle 7.4.: Maximale Eingangsdatenraten der beiden Architekturvarianten des MATMUNI-Systems

Architekturvariante	f_{max} [MHz]	BW_{max} [Gbit/s]	FR_{max} [Frames/s] ($L_{Frame} = 64$ Byte)
Referenzsystem	126	1,008	$2{,}1 \cdot 10^6$
NoC-basierte Architektur	165	1,320	$2{,}75 \cdot 10^6$

Jeder Simulationsdurchlauf besteht aus zwei Phasen. In der Einschwingperiode, in der jedoch noch keine Messungen erfolgen, werden 1000 Frames an das DUT gesendet, um das Gesamtsystem aus der transienten Startphase in einen stationären Betriebszustand zu versetzen. Anschließend folgt die eigentliche Messphase, in der weitere 5000 Frames entsprechend der jeweiligen Eingangsdatenrate erzeugt und zum DUT gesendet werden. Unabhängig von der Erzeugung der Stimuli basierend auf den in Tabelle 7.4 dargestellten Eingangsdatenraten erfolgt

die Taktung der DUTs in allen Simulationen mit den durch die Synthese ermittelten maximalen Taktfrequenzen.

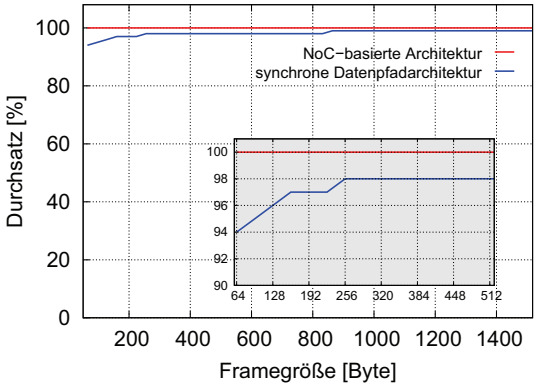
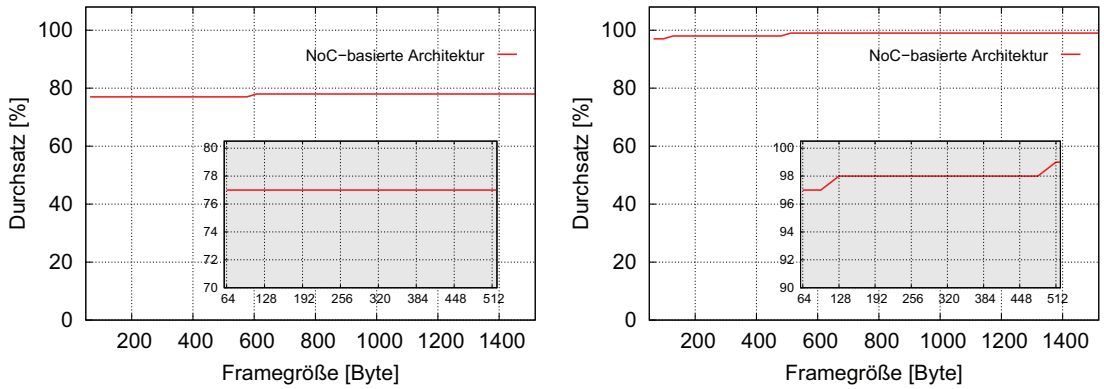


Abbildung 7.10.: Durchsatz über der Framegröße beider Architekturvarianten bei $BW_{max} = 1{,}008$ Gbit/s

Für die Simulationen in Abbildung 7.10 entspricht 100 % Eingangsdatenrate einer Bandbreite von $BW_{max} = 1{,}008$ Gbit/s. Diese Bandbreite ergibt sich mithilfe des maximalen Systemtakts $f = 126$ MHz des Referenzsystems nach Tabelle 7.4. Das Referenzsystem wird somit an seiner physikalischen Grenze betrieben. Für z. B. 64 Byte große Frames ist $FR_{max} = 2{,}1 \cdot 10^6$ Frames/s. Das Referenzsystem erreicht mit steigender Framegröße einen Durchsatz von maximal 99 %. Für den Worst Case, d. h. bei minimaler Framegröße, beträgt der Durchsatz des Referenzsystems 94 %. Dies entspricht einer absoluten Framerate von $1{,}974 \cdot 10^6$ Frames/s und repräsentiert den Spitzendurchsatz des Referenzsystems. Das NoC-basierte System erreicht für alle Framegrößen einen Durchsatz von 100 %. Dieser Unterschied ist durch die insgesamt höheren Frequenzen der Taktsignale in der NoC-basierten Variante gegenüber dem Referenzsystem begründet.

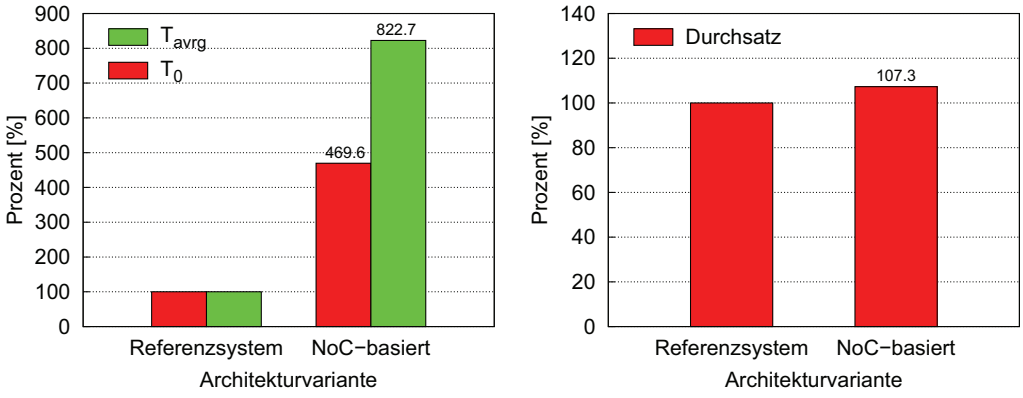
Um auch den Spitzendurchsatz des NoC-basierten Systems zu ermitteln, entspricht in Abbildung 7.11a eine Eingangsdatenrate von 100 % einer Bandbreite von $BW_{max} = 1{,}320$ Gbit/s. Diese ergibt sich aus der maximalen Taktfrequenz der Eingangsports des Systems (siehe Tabelle 7.4). Weitestgehend unabhängig von der Framegröße beträgt der erzielbare Durchsatz ca. 77 % der Eingangsdatenrate. Dies entspricht für den Fall minimal großer Frames einer absoluten Framerate von ca. $2{,}1175 \cdot 10^6$ Frames/s und stellt den Spitzendurchsatz des NoC-basierten MATMUNI-Systems dar. Der Grund für den Verwurf von ca. 23 % der Frames ist jedoch nicht im Kommunikationssubsystem zu suchen, da das NoC mit $W_{bit} = 32$ Bit und $f_{NoC} = 133$ MHz eine Kanalkapazität von $BW_{Channel} = 4{,}256$ Gbit/s besitzt. Hingegen stellt das Framebuffermodul den Engpass im logischen Datenpfad dieser Architektur dar, da dieses Modul nur eine maximale Frequenz von 131 MHz erreicht (siehe Tabelle 7.3) und intern auf einem 8 Bit breiten Datenpfad



(a) Simulation mit $BW_{max} = 1{,}320$ Gbit/s **(b)** Simulation mit $BW_{max} = 1{,}048$ Gbit/s

Abbildung 7.11.: Maximaler Durchsatz der NoC-basierten Architektur über der Framegröße

basiert. Um dies zu verifizieren, wurde in Abbildung 7.11b der Durchsatz des NoC-basierten MATMUNI-Systems mit einer maximalen Eingangsdatenrate von 1,048 Gbit/s simuliert, die sich aus der maximalen Frequenz des Framebuffermoduls nach Formel (7.2) errechnet. Der Durchsatz nähert sich dabei mit steigender Framegröße 100 % an. Bei 64-Byte großen Frames beträgt der Durchsatz ca. 97 %. Mithilfe von Formel (7.3) lässt sich daraus ein absoluter Durchsatz von ca. $2{,}1178 \cdot 10^6$ Frames/s ermitteln, welcher mit dem bereits ermittelten Spitzendurchsatz nahezu übereinstimmt und die Annahme bestätigt, dass das Framebuffermodul der limitierende Faktor des MATMUNI-Systems ist.



(a) Prozentualer Vergleich der Latenz **(b)** Prozentualer Vergleich des Durchsatzes

Abbildung 7.12.: Gegenüberstellung verschiedener Leistungsparameter des Referenzsystems (= 100 %) und der NoC-basierten Architektur

Abbildung 7.12 veranschaulicht die ermittelte minimale und durchschnittliche Latenz sowie den Durchsatz beider Systeme relativ zum Referenzsystem. Zusammenfassend lässt sich somit feststellen, dass das NoC-basierte MATMUNI-System trotz einer bis über 800 % höheren Latenz einen ca. 7 % höheren Durchsatz erzielt als das Referenzsystem auf Basis einer synchronen Datenpfadarchitektur. Allein durch den Wechsel der Systemarchitektur und des Kommunikationsparadigmas konnte die Leistungsfähigkeit gesteigert werden. Jedoch muss damit der Kompromiss eines erhöhten Ressourcenbedarfs eingegangen werden. Auf verschiedenen Ebenen verbleiben allerdings noch weitere Möglichkeiten der Optimierung des NoC-basierten MATMUNI-Systems, die in dieser Arbeit jedoch nicht weiter betrachtet werden:

- Das größte Optimierungspotential bzgl. der Leistungsfähigkeit weisen die einzelnen Module der MATMUNI-Anwendung auf, insbesondere das Framebuffermodul, da sie für das synchrone Referenzsystem entworfen aber für die Fallstudie in diesem Kapitel in einem asynchronen NoC-basierten System nachgenutzt wurden. Zu möglichen Modifikationen zählen u. a. die Anpassung der Datenpfadbreite der Funktionsmodule (8 Bit) an die Kanalbreite der NoC-Infrastruktur (32 Bit) und die Nachnutzung der in den RNIs ohnehin vorhandenen FIFO-Puffer für die Anwendung selbst. Somit kann auch der Hardware-Overhead durch die RNIs gesenkt werden.

- Durch eine Verbreiterung der Übertragungskanäle im NoC kann die Serialisierungslatenz zusätzlich verringert und die inter-IP-Core-Kommunikation beschleunigt werden. Einerseits ist dies völlig transparent für die Anwendung. Andererseits besteht ein FPGA zu 80 bis 90 % aus Verdrahtungsressourcen [DeH99], so dass diese Modifikation sich nicht kritisch auf den Ressourcenverbrauch auswirkt. Darüber hinaus resultiert eine geringere Latenz in einem kleineren Backlog und erlaubt die Verwendung kleinerer Puffer.

- Während die bisherigen Synthese-Ergebnisse auf Standardeinstellungen der genutzten Tools beruhen, bietet die Verwendung zusätzlicher Synthese-Constraints [XCG] weitere Formen der Optimierung, insbesondere bzgl. der topologischen Partitionierung und Platzierung der einzelnen Systemkomponenten im FPGA.

7.4. Zusammenfassung des Kapitels

Um mit der steigenden SoC-Komplexität umzugehen, ist der Einsatz alternativer Designkonzepte nötig, die sich insbesondere auf die mit der zunehmenden Design-Produktivitätslücke einhergehenden Problemstellungen auf physikalischer, funktionaler und ökonomischer Ebene fokussieren. Der in diesem Kapitel durchgeführte Machbarkeitsnachweis belegt die Anwendbarkeit der in Kapitel 6 entwickelten NoC-Architektur anhand des Übergangs von einer auf Punkt-zu-Punkt-Verbindungen basierenden synchronen Systemarchitektur zu einer asynchronen NoC-basierten Systemarchitektur für eine Beispielanwendung aus dem Bereich der Paketverarbeitung. Die während der Anwendungsabbildung und Evaluation gewonnenen Erfahrungen unterstreichen die in Abbildung 6.27 bereits dargestellte Entwicklung im Bereich komplexer SoCs. Abbildung 7.13 vergleicht in diesem Zusammenhang verschiedene Qualitätsmerkmale der konventionellen Ansätze mit Networks-on-Chip.

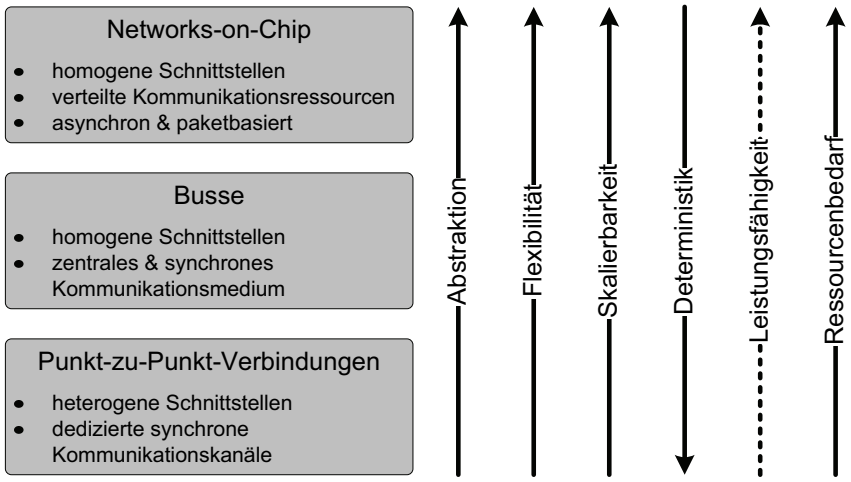


Abbildung 7.13.: Vergleich allgemeiner Charakteristika verschiedener Systemarchitekturen

Durch die strikte Trennung von Kommunikation und Verarbeitung in NoC-basierten Systemen wird einerseits das NoC mit seinen einheitlichen Schnittstellen als Kommunikationssubsystem vor der Anwendung abstrahiert. Ist das NoC zudem unterspezifiziert und anwendungsunabhängig entworfen, so wie es in dieser Arbeit in Abschnitt 6.2.1 beschrieben wurde, ist andererseits auch die eigentliche Applikation für das NoC vollkommen transparent. Dies resultiert wiederum in einem hohen Grad an Modularität und Wiederverwendbarkeit sowohl einzelner Komponenten der Anwendung als auch der Bausteine des NoCs. Durch diese Modularität sind NoC-basierte Systeme höchst flexibel und erweiterbar. Änderungen am System können schnell und leicht

durchgeführt werden, z. B. in Bezug auf die Integration und Platzierung zusätzlicher Funktionsmodule, wie es anhand der Teilung des zentralen Speicherarbiters des Referenzsystems in Abschnitt 7.2.2 demonstriert wurde.

Weiterhin profitieren NoC-basierte Systeme von einer hohen und im Vergleich zu den klassischen Ansätzen skalierbaren Kommunikationsleistung. Auf der einen Seite ist die Bandbreite einzelner Kommunikationskanäle konstant, da die Taktfrequenz der vorgestellten mesochronen NoC-Infrastruktur unabhängig von ihrer Größe ist. Zudem erlaubt das NoC die parallele Nutzung der verteilten Kommunikationsressourcen. Auf der anderen Seite sind die einzelnen IP-Cores durch den GALS-Ansatz auf Taktebene voneinander entkoppelt und werden durch angepasste individuelle Taktsignale getrieben.

In Abbildung 7.13 ist der Pfeil für die Leistungsfähigkeit jedoch nur gestrichelt dargestellt, da eine allgemeine Verbesserung der Leistungsfähigkeit nicht das primäre Ziel bei der Verwendung einer NoC-Infrastruktur ist, sondern der Umgang mit Komplexität. Somit kann eine Leistungsverbesserung nicht automatisch daraus geschlussfolgert werden. Einerseits ist ein direkter Vergleich verschiedener Realisierungsformen derselben Applikation bzgl. der Leistungsfähigkeit zumeist nicht möglich, da von vornherein die Entscheidung für eine konkrete Architektur feststeht. Am Beispiel des MATMUNI-Systems aus Abschnitt 4.3 konnte jedoch gezeigt werden, dass sich die Leistungsfähigkeit verbessern *kann*. Andererseits ist nicht jede Anwendung für ein NoC geeignet. Dies ist von den Eigenschaften der Anwendung abhängig und inwiefern die einzelnen Prozesse und Aufgaben räumlich und zeitlich verschoben werden können, um von der Parallelität einer NoC-Infrastruktur zu profitieren. Zum Beispiel ist das Systemverhalten NoC-basierter SoCs aufgrund ihrer Asynchronität auf Takt- und Kommunikationsebene nicht deterministisch und damit nicht mehr eindeutig vorhersagbar, wodurch speziell für Anwendungen aus dem Echtzeitbereich zusätzliche Vorkehrungen getroffen werden müssen.

Der Preis der genannten Vorzüge von NoC-Architekturen ist allerdings ein erhöhter Ressourcenbedarf, insbesondere gegenüber Punkt-zu-Punkt-Verbindungen. Jedoch erfüllt das NoC abhängig von der Anwendung indirekt noch weitere Funktionen, die über die reine Weiterleitung der Informationen hinausgehen, z. B. die Möglichkeit des Clock-Domain-Crossings oder die Nachnutzung der in den RNIs ohnehin vorhandenen FIFO-Puffer für die Anwendung. Dies verleiht dem Ausspruch „The network is the computer.“ zusätzliches Gewicht.

Teil IV.

Zusammenfassung

Indes sie forschten, röntgten, filmten, funkten,
entstand von selbst die köstlichste Erfindung:
der Umweg als die kürzeste Verbindung zwischen zwei Punkten.
(Erich Kästner, dt. Schriftsteller)

Kapitel 8.

Ergebnisse der Arbeit

Netzwerke im Großen und im Kleinen waren Thema dieser Dissertationsschrift, wie Abbildung 6.2 auf Seite 97 gezeigt hat. Weitflächige Kommunikationsnetze, insbesondere in Verbindung mit dem Internet, werden zunehmend komplexer. Gleiches gilt für den Bereich mikroskopischer Chip-interner Verbindungs- und Kommunikationsstrukturen und den Entwurf von Systems-on-Chip. In Letztgenanntem lassen sich durch die Entwicklung und Manifestierung des Network-on-Chip-Paradigmas prinzipielle Mechanismen und Zusammenhänge klassischer Datennetze nachnutzen. Die Beiträge der Arbeit stellen Lösungsansätze dar, welche sich mit einigen der Probleme und Randerscheinungen auseinandersetzen, die mit dieser zunehmenden Komplexität einhergehen [Kub08]. Im Folgenden sind die wesentlichen Ergebnisse der Arbeit zusammengefasst.

Der Ist-Zustand sowie Trends und Entwicklungen im Telekommunikationssektor wurden in Kapitel 3 aufgearbeitet und Anforderungen an aktuelle und zukünftige Telekommunikationsnetzwerke abgeleitet. Der Fokus lag dabei vor allem auf Ethernet-basierten Teilnehmerzugangsnetzen und dem Internet. Auf diese Bereiche entfallen zwei wesentliche Bestandteile der Arbeit.

- Mit einer Architektur zur Schicht-2-Adressumsetzung (MAC Address Translation) wurde in Kapitel 4 ein Ansatz sowohl zur Erhöhung der Sicherheit in Ethernet-basierten Teilnehmerzugangsnetzen als auch zur Skalierung der Anzahl der Schicht-2-Adressen in Richtung der Kernnetze vorgestellt [KWD+06, WKTB06, KWT+07]. Der diskutierte Ansatz ist zudem flexibel konfigurierbar und erlaubt eine transparente Integration in die existierende Infrastruktur Ethernet-basierter Teilnehmerzugangsnetzwerke.

- Auf Ebene des Internet Protokolls wurde in Kapitel 5 mit IPclip eine neuartige netzwerkübergreifende Architektur zur Realisierung von Trust-by-Wire im Internet präsentiert [DKW+08c, WKD+08]. Die IPclip-Architektur gewährleistet im paketvermittelten Inter-

net ein ähnliches Maß an Vertrauenswürdigkeit, wie es z. B. in klassischen leitungsvermittelten Sprachnetzen inhärent durch Rufnummer und Leitungsidentifikation gegeben ist. Durch IPclip wird sowohl die Migration klassischer Dienste ins Internet unterstützt als auch die Sicherheit im Internet erhöht. Die praktische Relevanz von IPclip wurde dazu anhand verschiedener zeitgemäßer und brisanter Anwendungsfälle demonstriert [KWD+08b, KWD+08c, KWD+08a].

- Darüber hinaus konnten diese beiden industrienahen Projekte – MAC Address Translation und IPclip – bis zur Prototypreife entwickelt und umgesetzt sowie als voll funktionsfähige Demonstratoren auf Basis von FPGA-Entwicklungsboards, welche dem Industriepartner u. a. für Vorfelduntersuchungen dienten, öffentlich vorgestellt werden [KWD+07, DKW+08a, DKW+09]. Ausgewählte Features des Ansatzes zur Schicht-2-Adressumsetzung, z. B. die Ausnahmebehandlung bestimmter Protokolle, wurden in Produkte des Industriepartners integriert. Das Grundprinzip und der Mechanismus von IPclip wurden vom Industriepartner als Erfindungsmeldung beim Europäischen Patentamt eingereicht, welches sich dort zurzeit in Bearbeitung befindet.

In der Telekommunikation kommen für verschiedenste Aufgaben der Paketverarbeitung in Hardware verbreitet komplexe integrierte Schaltkreise und Systems-on-Chip zum Einsatz, insbesondere auch in Form feldprogrammierbarer Bausteine. Um auch zukünftig den steigenden Datenraten genügen zu können und Dienste in einer ansprechenden bzw. mit der erforderlichen Güte bereitstellen zu können, müssen die genutzten Bausteine hochgradig leistungsfähig, flexibel und skalierbar bzgl. elektrischer Parameter und des Ressourcenverbrauchs sein. Deshalb war die Untersuchung eines neuartigen Architekturkonzepts für Systems-on-Chip Schwerpunkt des zweiten Teils dieser Arbeit. Konkret handelte es sich um die Entwicklung einer Network-on-Chip-Architektur zur Nutzung als Overlay-Netzwerk in FPGAs [KWHT07]. Die während der Entwicklung entstandenen funktionalen und topologischen Optimierungsansätze stellen zwei weitere zentrale Beiträge der Arbeit dar.

- Es wurde eine schlanke Network-on-Chip-Architektur mit hoher Kommunikationsbandbreite vorgestellt, welche als generisches und erweiterbares Kommunikationssubsystem in Systems-on-Chip Verwendung findet. Während die Entwicklung der einzelnen Komponenten des Network-on-Chip in dieser Arbeit noch einen Mehraufwand darstellte, kann bei der Umsetzung zukünftiger Anwendungen und Projekte von der Wiederverwendbarkeit und hohen Modularität des NoC profitiert werden. Die Network-on-Chip-Komponenten repräsentieren eine fertige und funktionsfähige Kommunikationsplattform, welche sowohl auf andere FPGA-Typen portiert als auch in ASICs (nicht Fokus dieser Arbeit) verwendet

werden kann. Diese Wiederverwendbarkeit wirkt sich verkürzend auf die Entwicklungszeit zukünftiger Applikationen als Systems-on-Chip aus.

- In Abschnitt 6.3 wurde mit HSM (Hybrid Switching Mechanism) ein optimiertes Verfahren zur Flusskontrolle in einem mesochron angesteuerten Network-on-Chip vorgestellt [KHT07]. Die wesentlichen Eigenschaften von HSM sind dabei einerseits ein geringer Hardware-Aufwand und andererseits eine hohe Effizienz. Zudem ist die maximale Taktfrequenz des HSM-basierten NoC konstant und unabhängig von den Ausmaßen der Network-on-Chip-Infrastruktur, so dass von einer hohen und skalierenden Kommunikationsleistung profitiert werden kann.

- Mit dem Border-Enhanced Mesh (BEAM) wurde in Abschnitt 6.4 eine Modifikation der Topologie klassischer k-facher 2-Würfel präsentiert. Durch einen hohen Konzentrationsfaktor können NoC-Router eingespart und damit der Hardware-Overhead der NoC-Infrastruktur gegenüber der eigentlichen Anwendung deutlich gesenkt werden. Die Reduzierung der Anzahl der Router resultiert zudem in einer geringeren durchschnittlichen Pfadlänge und damit in einer geringeren durchschnittlichen Latenz im NoC, was sich positiv auf die inter-IP-Core-Kommunikation auswirkt. Darüber hinaus können Networks-on-Chip auf BEAM-Basis eine höhere Effizienz aufweisen. Simulationen haben gezeigt, dass die Kommunikationsressourcen einerseits gleichmäßiger und andererseits insgesamt höher ausgelastet sind als in vergleichbaren k-fachen 2-Würfeln.

- In Kapitel 7 wurde durch einen Proof-of-Concept die Anwendbarkeit der entwickelten NoC-Infrastruktur anhand der Abbildung einer Beispielanwendung aus dem Bereich der Paketverarbeitung auf das NoC [KCHT07] und des Vergleichs mit einem Referenzsystem nachgewiesen. Die Eigenschaften beider Systeme wurden mittels Synthese- und Simulationsergebnissen veranschaulicht und erörtert. Demzufolge erreicht die NoC-basierte Architekturvariante einen ca. 7 % höheren Durchsatz. Darüber hinaus bieten sich jedoch noch weitere Optimierungsmöglichkeiten bzgl. der Leistungsfähigkeit und insbesondere im Bereich des Ressourcenverbrauchs, welcher sich aufgrund der ineffizienten Integration der ursprünglichen Funktionsblöcke des Referenzsystems mehr als verdoppelte.

Weitere Informationen zu den Ergebnissen und den einzelnen Themengebieten sowie die Publikationen des Autors sind auf den folgenden Projekt-Webseiten einsehbar:

- `http://www.networks-on-chip.com`
- `http://www.imd.uni-rostock.de/networking`

As of now, computer networks are still in their infancy. But as they grow up and become more sophisticated, we will probably see the spread of 'computer utilities' which, like present electric and telephone utilities, will service individual homes and offices across the country.

(Leonard Kleinrock, „Vater" des Internets, 1969)

Kapitel 9.

Ausblick & Fazit

Neben den in der Arbeit adressierten Problemstellungen in der Telekommunikation und im Entwurf digitaler integrierter Systeme existieren noch zahlreiche angrenzende Forschungsfelder mit hohem Nutzwert und Innovationspotential bzw. ergeben sich aus den aktuellen Entwicklungen.

Mehrfach wurde in der Arbeit die Telekommunikation als hochdynamischer Bereich beschrieben, der zurzeit durch die Zusammenführung verschiedener separater Teilnetze in ein gemeinsames Gesamtnetz geprägt ist. Zugangsnetze auf Basis unterschiedlicher Technologien werden durch ein paketvermitteltes Kernnetz verbunden und ermöglichen die globale und trotzdem transparente Nutzung und Integration von Diensten, wie es in Kapitel 1 mit den Begriffen Ubiquität und Hyperkonnektivität illustriert wurde. Die erste Migrationsphase, die technologische Integration und Konsolidierung, ist bereits abgeschlossen. Aktuelle Entwicklungen befassen sich primär mit der Verbesserung der Qualität und Verfügbarkeit der Dienste. Die dritte Migrationsphase hingegen wird noch mehr als bisher das Thema *Sicherheit* betreffen, da das Internet per se ein offen zugängliches Medium ist. Während diese Offenheit einer der Grundsätze und eine Ursache für den bisherigen Erfolg und das Wachstum des Internets ist [Kle03, Kle04], ist sie gleichzeitig auch ein Grund für die zunehmenden Bedrohungen und Kriminalität im Internet. Die Brisanz dieses Themas ist z. B. an aktuellen Meldungen und Debatten bzgl. der Verbreitung und des Missbrauchs vertraulicher Kundendaten und auch an der Verabschiedung und Umsetzung des „Nationalen Plans zum Schutz der Informationsinfrastrukturen" des Bundesministeriums des Inneren [Bun05, Bun07a] zu erkennen. Sicherheit in der Telekommunikationsbranche wird deshalb in den kommenden Jahren eine übergeordnete Rolle spielen, einerseits zum Schutz der einzelnen Nutzer und andererseits zum Schutz der Netzwerkinfrastruktur. Da das Internet keine geographischen Grenzen hat, ist dies eine globale Problemstellung.

Dass die Nutzerzahlen und damit die Datenraten auch weiterhin ansteigen, wird gemeinhin

als Tatsache betrachtet und ist durch Langzeitstatistiken belegbar [DCX, JPX]. Neben der Netzwerksicherheit wird deshalb zukünftig der *Energieverbrauch* als zweites primäres Problem in der Telekommunikation angesehen, da der Energiebedarf der einzelnen Netzwerkkomponenten mit den zu verarbeitenden Datenmengen ansteigt [Aoy08]. Während der Zunahme der Nutzer und Datenraten aus funktionaler Sicht durch neue Protokolle und Technologien begegnet wird, z. B. durch IPv6 und optische Medien, sind Maßnahmen zur Senkung des Energieverbrauchs auf allen Ebenen und in allen angrenzenden Bereichen der Telekommunikation nötig.

Zu diesen Bereichen zählt insbesondere auch der Entwurf integrierter Schaltkreise und Systeme als Rückgrat der Paket- und Datenverarbeitung in der Telekommunikation. Networks-on-Chip stellen in diesem Gebiet ein neuartiges Architekturkonzept für komplexe SoCs dar. Skalierbarkeit, u. a. auch im Bereich des Leistungsverbrauchs, gepaart mit hoher Performanz sind dabei wesentliche Charakteristika NoC-basierter Systeme. Während NoCs in dieser Arbeit als Overlay-Strukturen in FPGAs genutzt werden, schlagen erste Forschungsarbeiten bereits die Nutzung von NoCs als *fest verdrahtete* Kommunikationsstrukturen in FPGAs vor [HKHT05, GCK07, GBHW08, SPA+08], um die bisher auf einer Hierarchie von Signalleitungen basierenden Verbindungsressourcen, u. a. auch globale Leitungen, mit einer NoC-Infrastruktur zu ersetzen. Das Leistungspotential und das Spektrum der möglichen Nutzungsformen von FPGAs kann dadurch zusätzlich erweitert werden, wie es in [KHT05, KHST06] z. B. mit Laufzeit-adaptiven Systemen zur autonomen Anpassung paketverarbeitender Systeme an veränderte Lastsituationen in Netzwerken vorgeschlagen wird. Dieser interessante Forschungsbereich erfordert jedoch die intensive Mitarbeit und Offenheit der führenden Hersteller feldprogrammierbarer Bausteine sowie eine Standardisierung von NoC-Schnittstellen.

Trotz der weltweit zunehmenden Forschungs- und Entwicklungsaktivitäten haben NoCs den kommerziellen Durchbruch jedoch noch *nicht* wirklich erzielt. Prototypische Realisierungen wie Intel's NoC-basierter 80-Kern-Chip [VHR+07] sind bisher nur als wissenschaftliche Fallstudien zu betrachten. Nach Arteris, Inc. [Art05] werden sich NoCs durchsetzen, wenn eins oder mehrere der folgenden Kriterien gegenüber konventionellen Ansätzen effektiv erfüllt werden:

- Reduktion der Herstellungskosten, der Time-to-Market oder Time-to-Volume von SoCs
- Erhöhung der Performanz integrierter Systeme
- Minimierung von Design-Risiken

Augenscheinlich geht der Trend im System- und Schaltungsentwurf immer mehr von Multi-Core- in Richtung Many-Core-Systeme (siehe Abbildung 6.27 auf Seite 145), was gleichzeitig ein treibender Faktor für Networks-on-Chip ist. Dass derartige Systeme aus fertigungstechnischer Sicht durchaus realisierbar sind, zeigt u. a. der bereits erwähnte 80-Kern-Chip von Intel. Die

größten Herausforderungen liegen somit nicht primär in der Schaffung der Hardware-Plattform, sondern vielmehr in der Programmierung und der Ausschöpfung des Leistungspotentials dieser hochgradig parallelen Architekturen, d. h., im Bereich der Entwicklungswerkzeuge und Software [Tur08]. Dies wirft verschiedene Fragestellungen bezüglich dafür geeigneter Applikationen, praktikabler Programmiermodelle und Beschreibungssprachen sowie einer ökonomisch vertretbaren Verifikation derartiger Systeme auf. Die *effiziente Abbildung* von Anwendungen auf parallele Systeme, welche nach wie vor durch das Amdahlsche Gesetz bestimmt ist [Amd67, Rod85], und das *Verständnis von Parallelität* sind deshalb ausschlaggebend.

Für den Umgang mit den genannten Fragestellungen und Problemen werden jedoch nicht nur Abstraktion, Standardisierung und technologische Fortschritte Schlüsselrollen spielen. Aufgrund der vielen Interessengruppen sowohl im Telekommunikationssektor als auch in der Halbleiterindustrie ist die weitere Entwicklung oft durch finanzielle und politische Entscheidungen gehemmt. Das Schlusswort dieser Dissertationsschrift ist deshalb ein Zitat von Kleinrock aus dem Jahr 1997 [LCC+97], welches trotz seines „Alters“ nichts an Aktualität eingebüßt hat und prinzipiell auf alle der genannten Bereiche bezogen werden kann:

> If the Internet stumbles, it will not be because we lack technology, vision, or motivation but because we cannot set a direction and march collectively into the future.

Literaturverzeichnis

[ACG+03] ANDRIAHANTENAINA, Adrijean ; CHARLERY, Hervé ; GREINER, Alain ; MORTIEZ, Laurent ; ZEFERINO, Cesar A.: SPIN: A Scalable, Packet Switched, on-Chip Micro-Network. In: *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'03)*. München, Deutschland, März 2003. – ISBN 0–7695–1870–2, S. 20070–20073 (zitiert auf S. 103)

[ACN+07] ASCIA, Giuseppe ; CATANIA, Vincenzo ; NUOVO, Alessandro G. D. ; PALESI, Maurizio ; PATTI, Davide: Efficient Design Space Exploration for Application Specific Systems-on-a-Chip. In: *Journal of Systems Architecture: the EUROMICRO Journal* 53 (2007), Nr. 10, S. 733–750. – ISSN 1383–7621 (zitiert auf S. 148)

[ACP05] ASCIA, Giuseppe ; CATANIA, Vincenzo ; PALESI, Maurizio: Mapping Cores on Network-on-Chip. In: *International Journal of Computational Intelligence Research (IJCIR)* 1 (2005), Dezember, Nr. 1-2, S. 109–126. – ISSN 0972–9836 (zitiert auf S. 150)

[AG03] ANDRIAHANTENAINA, Adrijean ; GREINER, Alain: Micro-Network for SoC: Implementation of a 32-Port SPIN Network. In: *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'03)*. München, Deutschland, März 2003. – ISBN 0–7695–1870–2, S. 11128–11129 (zitiert auf S. 103)

[Aga91] AGARWAL, Anant: Limits on interconnection network performance. In: *IEEE Transactions on Parallel and Distributed Systems* 2 (1991), Oktober, Nr. 4, S. 398–412. – ISSN 1045–9219 (zitiert auf S. 110)

[Agi06] AGILENT TECHNOLOGIES, INC.: *Understanding DSLAM and BRAS Access Devices*. White Paper, Juli 2006. – `http://cp.literature.agilent.com/litweb/pdf/5989-4766EN.pdf` (zitiert auf S. 32 und 43)

[AH06] AL-HASHIMI, Bashir M.: *System-on-Chip: Next Generation Electronics*. Institution of Engineering and Technology, 2006. – ISBN 0–86341–552–0 (zitiert auf S. 94)

[Alc05] ALCATEL: *Operational Excellence in Triple Play Service Delivery: The Role of Policy-Enabled Subscriber Service Management*. White Paper, August 2005. – `http://www1.alcatel-lucent.com/bnd/news/docs/3Play_ServDeliv_wp.pdf` (zitiert auf S. 26)

[Alf05] ALFKE, Peter: *Metastable Recovery in Virtex-II Pro FPGAs*. Xilinx, Inc., Application Note XAPP094. `http://www.xilinx.com/support/documentation/application_notes/xapp094.pdf`. Version: Februar 2005. – Version 3.0 (zitiert auf S. 101)

[ALT] *Wireline End Market*. Altera Corporation, . – `http://www.altera.com/end-markets/wireline` (zitiert auf S. 57)

[Alt99] ALTERA CORPORATION: *Metastability in Altera Devices*. Application Note 42. `http://www.altera.com/literature/an/an042.pdf`. Version: Mai 1999. – Version 4.0 (zitiert auf S. 101)

[Amd67] AMDAHL, Gene: Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities. In: *Spring Joint Computer Conference, AFIPS Conference Proceedings Volume 30*, 1967, S. 483–485 (zitiert auf S. 183)

[Aoy08] AOYAMA, Tomonori: A New Generation Network. In: *Proceedings of the 1st ITU-T Kaleidoscope Conference: Innovations in Next Generation Networks - Future Network and Services (K-INGN'08)*. Genf, Schweiz, Mai 2008. – ISBN 92–61–12441–0, S. 3–10. – `http://www.itu.int/dms_pub/itu-t/oth/29/01/T29010010010001PDFE.pdf` (zitiert auf S. 182)

[Arb07] ARBOR NETWORKS, INC.: *Worldwide Infrastructure Security Report*. Volume III, September 2007. – `http://www.arbornetworks.com/report` (zitiert auf S. 28)

[Art05] ARTERIS: *A Comparison of Network-on-Chip and Busses*. White Paper, 2005. – `http://www.arteris.com` (zitiert auf S. 95, 96 und 182)

[Asc07] ASCHENBRENNER, Norbert: Neues Soziales Netz. In: *Pictures of the Future – Die Zeitschrift für Forschung und Innovation* Herbst (2007), 80–83. `http://www.siemens.de/pof`. – ISSN 1618–548X (zitiert auf S. 22)

[Axe03] AXELSON, Jan: *Embedded Ethernet and Internet Complete: Designing and Programming Small Devices for Networking*. Lakeview Research LLC, 2003. – ISBN 1–931448–01–9 (zitiert auf S. 14)

[BB03] BARABÁSI, Albert-László ; BONABEAU, Eric: Scale-Free Networks. In: *Scientific American* 288 (2003), Mai, Nr. 5, S. 50–59. – ISSN 0036–8733 (zitiert auf S. 20)

[BB04] BERTOZZI, Davide ; BENINI, Luca: Xpipes: A Network-on-Chip Architecture for Gigascale Systems-on-Chip. In: *IEEE Circuits and Systems Magazine* 4 (2004), Nr. 2, S. 18–31. – ISSN 1540–7977 (zitiert auf S. 103)

[BCGK04] BOLOTIN, Evgeny ; CIDON, Israel ; GINOSAR, Ran ; KOLODNY, Avinoam: Cost considerations in Network on chip. In: *INTEGRATION, the VLSI Journal, Special Issue: Networks on chip and reconfigurable fabrics* 38 (2004), Oktober, Nr. 1, S. 19–42. – ISSN 0167–9260 (zitiert auf S. 95)

[BD06] BALFOUR, James ; DALLY, William J.: Design tradeoffs for tiled CMP on-chip networks. In: *Proceedings of the 20th Annual International Conference on Supercomputing (ICS'06)*. Cairns, Queensland, Australien, 2006. – ISBN 1–59593–282–8, S. 187–198 (zitiert auf S. 142)

[Bec05] BECK, Michael: *Ethernet in the First Mile: The IEEE 802.3ah EFM Standard*. McGraw-Hill, 2005. – ISBN 0–07–146991–5 (zitiert auf S. 25)

[Bei06] BEIJNUM, Iljitsch van: *Running IPv6*. Apress, 2006. – ISBN 1–59059–527–0 (zitiert auf S. 51 und 105)

[Ber03] BERGERON, Janick: *Writing Testbenches: Functional Verification of HDL Models*. Zweite Auflage. Kluwer Academic Publishers, 2003. – ISBN 1–4020–7401–8 (zitiert auf S. 164)

[Beu07] BEUTHNER, Andreas: *Vom LAN-Protokoll zum WAN-Champion – Das Ethernet als Konkurrenz zu Sonet, SDH, Frame Relay und ATM*. Online-Artikel. `http://www.searchnetworking.de`. Version: Mai 2007 (zitiert auf S. 33 und 41)

[Bha05] BHAT, Shubha: *Energy Models for Network-on-Chip Components*, Technische Universität Eindhoven, Diplomarbeit, Dezember 2005. – `http://alexandria.tue.nl/extra1/afstversl/wsk-i/bhat2006.pdf` (zitiert auf S. 96)

[BI07] BUNDESVERBAND INFORMATIONSWIRTSCHAFT, Telekommunikation und neue Medien (.: *Zukunft digitale Wirtschaft*. Gemeinsame Studie des BITKOM e.V. und der Roland Berger Strategy Consultants. `http://www.bitkom.org`. Version: 2007 (zitiert auf S. 22 und 23)

[BKC08] BOGUÑÁ, Marián ; KRIOUKOV, Dmitri ; CLAFFY, K. C.: Navigability of complex networks. In: *Nature Physics* 4 (2008), November, Nr. 11, S. 7. – ISSN 1745–2473 (zitiert auf S. 20)

[BKP] BERTOZZI, Davide (Hrsg.) ; KUMAR, Shashi (Hrsg.) ; PALESI, Maurizio (Hrsg.): *VLSI Design, Volume 2007, Special Issue: Networks-on-Chip*. Hindawi Publishing Corporation. ISSN 1065–514X (zitiert auf S. 100)

[BL03] BREBNER, Gordon ; LEVI, Delon: Networking on Chip with Platform FPGAs. In: *Proceedings of the 2003 IEEE International Conference on Field-Programmable Technology (FPT'03)*. Tokio, Japan, Dezember 2003. – ISBN 0–7803–8320–6, S. 13–20 (zitiert auf S. 103)

[BM02] BENINI, Luca ; MICHELI, Giovanni D.: Networks on Chips: A New SoC Paradigm. In: *IEEE Computer* 35 (2002), Januar, Nr. 1, S. 70–78. – ISSN 0018–9162 (zitiert auf S. 96)

[BM05] BLUSCHKE, Andreas ; MATTHEWS, Michael: FTTx. In: *dsl-review* B18 (2005), Nr. 03/05, 2. `http://xdsl.teleconnect.de/xDSL_germ/HTML/know_frame.html`. – ISSN 1612–2402 (zitiert auf S. 41)

[BM06a] BENINI, Luca ; MICHELI, Giovanni D.: *Networks on Chips: Technology and Tools*. Elsevier, 2006. – ISBN 0–12–370521–5 (zitiert auf S. 94, 100 und 103)

[BM06b] BJERREGAARD, Tobias ; MAHADEVAN, Shankar: A survey of research and practices of Network-on-chip. In: *ACM Computing Surveys* 38 (2006), Nr. 1, S. 1. – ISSN 0360–0300 (zitiert auf S. 100)

[BM06c] BLUSCHKE, Andreas ; MATTHEWS, Michael: Breitband – Stand und Prognosen. In: *dsl-review* A22, A32, A33, A35, A37 (2004–2006), Nr. 29/04, 40/05, 45/05, 16/06, 26/06, 2. `http://xdsl.teleconnect.de/xDSL_germ/HTML/know_frame.html`. – ISSN 1612–2402 (zitiert auf S. 24 und 25)

[BMB01] BLUSCHKE, Andreas ; MATTHEWS, Michael ; BADACH, Anatol: *xDSL-Fibel: Ein Leitfaden von A wie ADSL bis Z wie ZipDSL*. VDE-Verlag, 2001. – ISBN 3–8007–2557–6 (zitiert auf S. 25)

[BMF08] BERTOZZI, Davide ; MEDARDONI, Simone ; FERRANTE, Alberto: Network Interface Sharing Techniques for Area Optimized NoC Architectures. In: *Proceedings of the 11th Euromicro Conference on Digital System Design (DSD'08), Special Session on Prospective Aspects of Networks-on-Chip*. Parma, Italien, September 2008. – ISBN 978–0–7695–3277–6 (zitiert auf S. 142)

[BMS04] BLUSCHKE, Andreas ; MATTHEWS, Michael ; SCHIFFEL, Reinhard: *Zugangsnetze für die Telekommunikation*. Carl Hanser Verlag München Wien, 2004. – ISBN 3–446–22675–3 (zitiert auf S. 40)

[Bon04] BONEH, Dan: The Difficulties of Tracing Spam Email / Department of Computer Science, Stanford University. Version: September 2004. `www.ftc.gov/reports/rewardsys/expertrpt_boneh.pdf`. 2004. – Forschungsbericht (zitiert auf S. 84 und 86)

[BPS99] BENNETT, Jon C. R. ; PARTRIDGE, Craig ; SHECTMAN, Nicholas: Packet reordering is not pathological network behavior. In: *IEEE/ACM Transactions on Networking* 7 (1999), Nr. 6, S. 789–798. – ISSN 1063–6692 (zitiert auf S. 111)

[Bre02] BREBNER, Gordon: *Computers in Communication*. McGraw-Hill London, 2002. – ISBN 0–07–709198–1 (zitiert auf S. 7)

[BT04] BOUDEC, Jean-Yves L. ; THIRAN, Patrick: *Network Calculus – A Theory of Deterministic Queuing Systems for the Internet*. Springer, 2004 (LNCS 2050). – ISBN 3–540–42184–X (zitiert auf S. 12 und 167)

[BT07] BAUER, Oliver ; TENZ, Beate: *Entwicklung der Informationsgesellschaft – IKT in Deutschland*. Statistisches Bundesamt, Wiesbaden, 2007 http://www.destatis.de. – ISBN 978–3–8246–0817–1. – Ausgabe 2007 (zitiert auf S. 22)

[Bun04] BUNDESMINISTERIUM DER JUSTIZ: *Telekommunikationsgesetz (TKG)*. Bundesgesetz, Verwaltungsrecht. http://bundesrecht.juris.de/tkg_2004/index.html. Version: Juli 2004. – (letzte Gesetzestextänderung 18. Februar 2007) (zitiert auf S. 23)

[Bun05] BUNDESMINISTERIUM DES INNEREN: *Nationaler Plan zum Schutz der Informationsinfrastrukturen*. http://www.bmi.bund.de. Version: Juli 2005 (zitiert auf S. 181)

[Bun07a] BUNDESMINISTERIUM DES INNEREN: *Umsetzungsplan KRITIS des Nationalen Plans zum Schutz der Informationsinfrastrukturen*. http://www.bmi.bund.de. Version: September 2007 (zitiert auf S. 181)

[Bun07b] BUNDESNETZAGENTUR FÜR ELEKTRIZITÄT, GAS, TELEKOMMUNIKATION, POST UND EISENBAHNEN: *Jahresbericht 2006*. http://www.bundesnetzagentur.de. Version: Februar 2007 (zitiert auf S. 22)

[Bun07c] BUNDESNETZAGENTUR FÜR ELEKTRIZITÄT, GAS, TELEKOMMUNIKATION, POST UND EISENBAHNEN: *Tätigkeitsbericht 2006/2007 für den Bereich Telekommunikation*. http://www.bundesnetzagentur.de. Version: Dezember 2007 (zitiert auf S. 22)

[Cad04] CADENCE DESIGN SYSTEMS, INC.: *Clock Domain Crossing – Closing the Loop on Clock Domain Functional Implementation Problems*. White Paper, Juli 2004. – http://www.cadence.com/rl (zitiert auf S. 101)

[CBP+06] CIOFFI, John M. ; BRADY, Mark ; POURAHMAD, Vahbod ; JAGANNATHAN, Sumanth ; LEE, Wooyul ; KIM, Youngjae ; CHEN, Chiang-Yu ; SEONG, Kibeom ; YU, David ; OUZZIF, Meryem ; MARIOTTE, Hubert ; TARAFI, Rabah ; GINIS, George ; LEE, Bin ; CHUNG, Seong T. ; SILVERMAN, Peter J.: Vectored DSLs with DSM: The Road to ubiquitous Gigabit DSLs. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April - Mai 2006 (zitiert auf S. 25 und 106)

[CDK05] COULOURIS, George ; DOLLIMORE, Jean ; KINDBERG, Tim: *Distributed Systems: Concepts and Design*. Fünfte Auflage. Addison Wesley, 2005. – ISBN 0–321–26354–5 (zitiert auf S. 20 und 96)

[CGEDC+04] CHIRUVOLU, Girish ; GE, An ; ELIE-DIT-COSAQUE, David ; ALI, Maher ; ROUYER, Jessy: Issues and Approaches on Extending Ethernet Beyond LANs. In: *IEEE Communications Magazine* 42 (2004), März, Nr. 3, S. 80–86. – ISSN 0163–6804 (zitiert auf S. 44 und 54)

[CGS05] CHLAMTAC, Imrich (Hrsg.) ; GUMASTE, Ashwin (Hrsg.) ; SZABÓ, Csaba A. (Hrsg.): *Broadband Services – Business Models and Technologies for Community Services*. John Wiley & Sons, Ldt, 2005. – ISBN 0–470–02248–5 (zitiert auf S. 26)

[Cha03] CHAKRABORTY, Samarjit: *System-Level Timing Analysis and Scheduling for Embedded Packet Processors*. Institut für Technische Informatik und Kommunikationsnetze, Eidgenössische Technische Hochschule Zürich, Dissertation, April 2003 (zitiert auf S. 13)

[Chi98] CHIEN, Andrew A.: A Cost and Speed Model for k-ary n-Cube Wormhole Routers. In: *IEEE Transactions on Parallel and Distributed Systems* 9 (1998), Nr. 2, S. 150–162. – ISSN 1045–9219 (zitiert auf S. 108)

[Cis01] CISCO SYSTEMS INC.: *Dictionary of Internetworking Terms and Acronyms: Cisco Systems' Offical Internetworking Dictionary*. Cisco Press, 2001. – ISBN 1–58720–045–7. – http://www.cisco.com/univercd (zitiert auf S. 10)

[Cis06] CISCO SYSTEMS INC.: *Resolve IP Fragmentation, MTU, MSS, and PMTUD Issues with GRE and IPSEC*. White Paper. http://www.cisco.com/warp/public/105/pmtud_ipfrag.pdf. Version: Oktober 2006 (zitiert auf S. 77)

[CJMG07] CIOFFI, Joseph M. ; JAGANNATHAN, Sumanth ; MEHSENI, Mehdi ; GINIS, George: CuPON: The Copper Alternative to PON 100 Gb/s DSL Networks. In: *IEEE Communications Magazine* 45 (2007), Juni, Nr. 6, S. 132–139. – ISSN 0163–6804 (zitiert auf S. 25)

[CK05] CIDON, Israel ; KEIDAR, Idit: Zooming in on Network-on-Chip Architectures / Technion Department of Electrical Engineering. 2005 (CCIT 565). – Forschungsbericht (zitiert auf S. 103)

[CLM+04] CIOFFI, John M. ; LEE, Bin ; MOHSENI, Mehdi ; LESHEM, Amir ; YOUMING, Li: *GDSL (Gigabit DSL)*. Committee T1 – Telecommunications, Working Group T1E1.4 (DSL Access), Contribution 2003/487R1. http://www.eng.biu.ac.il/~leshema/standard%20contributions/3E144871-GDSL_final.pdf. Version: August 2004 (zitiert auf S. 25)

[CO01] COFFMAN, Kerry G. ; ODLYZKO, Andrew M.: Internet Growth: Is there a "Moore's Law" for Data Traffic? In: ABELLO, J. (Hrsg.) ; PARDALOS, P. M. (Hrsg.) ; RESENDE, M. G. C. (Hrsg.): *Handbbook of Massive Data Sets*. Kluwer Academic Publishers, 2001, S. 47–93 (zitiert auf S. 3 und 26)

[COM] *Google Earth COM API*. Google Inc. http://earth.google.com/comapi/ (zitiert auf S. 229)

[CTD08] CTD – CONNECTING THE DOTS: *Rise of the Participation Culture*. Marketing Directions, Inc. http://www.wsjb.com/RPC. Version: Februar 2008. – Report (zitiert auf S. 27)

[Dal91] DALLY, William J.: Express Cubes: Improving the Performance of k-ary n-cube Interconnection Networks. In: *IEEE Transactions on Computers* 40 (1991), Nr. 9, S. 1016–1023. – ISSN 0018–9340 (zitiert auf S. 141)

[Dal92] DALLY, William J.: Virtual-Channel Flow Control. In: *IEEE Transactions on Parallel and Distributed Systems* 3 (1992), März, Nr. 2, S. 194–205. – ISSN 1045–9219 (zitiert auf S. 111)

[Dan06] DANIELIS, Peter: *Realisierung und Implementierung eines Algorithmus zur Echtzeit-Mustererkennung in einem Ethernet-Datenstrom*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Diplomarbeit, Oktober 2006 (zitiert auf S. 30 und 83)

[Dau08] DAUM, Enrico: *Entwicklung und Analyse von Optimierungsansätzen für ein NoC-basiertes System*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Masterarbeit, September/Oktober 2008 (zitiert auf S. 137)

[DBT+06a] DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; KUBISCH, Stephan ; WIDIGER, Harald: Efficient Port-based Network Access Control for IP DSLAMs in Ethernet-based Fixed Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006 (zitiert auf S. 31)

[DBT+06b] DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; WIDIGER, Harald ; KUBISCH, Stephan: ACIP: An Access Control and Information Protocol for Ethernet-based Broadband Access Networks. In: *Proceedings of the 12th International Telecommunications Network Strategy and Planning Symposium (Networks'06) on CD-Rom*. Neu Delhi, Indien, November 2006. – ISBN 978–3–8007–2999–9 (zitiert auf S. 31)

[DCX] *Deutscher Commercial Internet Exchange*. – http://www.de-cix.de/content/network/Traffic-Statistics.html (zitiert auf S. 3, 26, 105 und 182)

[DeH99] DEHON, André: Balancing Interconnect and Computation in a Reconfigurable Computing Array (or, why you don't really want 100% LUT utilization). In: *Proceedings of the 1999 ACM/SIGDA 7th International Symposium on Field Programmable Gate Arrays (FPGA'99)*. Monterey, Kalifornien, USA, Februar 1999. – ISBN 1–58113–088–0, S. 69–78 (zitiert auf S. 171)

[Dem07] DEMUTH, Tino: *Analyse der Entwicklungen, Trends und des Stand der Technik im Forschungsgebiet Networks-on-Chip*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Kleiner Beleg, Dezember 2007 (zitiert auf S. 100)

[DEN] DENIC EG: *ENUM - Eine Nummer für alle Dienste*. – `http://www.denic.de/de/enum` (zitiert auf S. 69)

[DH04] DEWENTER, Ralf ; HAUCAP, Justus: *Die Liberalisierung der Telekommunikationsbranche in Deutschland*. Universität der Bundeswehr Hamburg, Fachgruppe Volkswirtschaftslehre, Diskussionspapier Nr. 27, März 2004 (zitiert auf S. 23)

[DKT07] DANIELIS, Peter ; KUBISCH, Stephan ; TIMMERMANN, Dirk: Realisierung und Implementierung eines Algorithmus zur Echtzeit-Mustererkennung in einem Ethernet-Datenstrom. In: *Tagungsband des 12. Symposium Maritime Elektrotechnik, Elektronik und Informationstechnik*. Rostock, Deutschland, Oktober 2007. – ISBN Universitätsdruckerei Rostock 685–07, S. 191–196 (zitiert auf S. 30 und 83)

[DKW+08a] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; LANGE, Christian: Trust-by-Wire in Packet-switched IP Networks: Calling Line Identification Presentation for IP. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE'08), University Booth*. München, Deutschland, März 2008. – `http://www.edacentrum.de/eda-netzwerke/universitybooth/ubooth08-full-program.pdf` (zitiert auf S. 70, 81, 178, 227 und 228)

[DKW+08b] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; TIMMERMANN, Dirk: A Conceptual Framework for Increasing Physical Proximity in Unstructured Peer-To-Peer Networks. In: *Proceedings of the 2008 IEEE Sarnoff Symposium (auf CD-Rom)*. Princeton, New Jersey, USA, April 2008. – ISBN 978–1–4244–1843–5 (zitiert auf S. 226)

[DKW+08c] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: IPclip – An Innovative Mechanism to Reestablish Trust-by-Wire in Packet-switched IP Networks. In: *3. Essener Workshop „Neue Herausforderungen in der Netzsicherheit" (EWNS'08)*. Essen, Deutschland, April 2008. – `http://wiki.uni-due.de/TdR/index.php/Essener_Workshop_zur_Netzsicherheit_2008_%28EWNS08%29` (zitiert auf S. 70, 81 und 177)

[DKW+09] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; ROHRBECK, Jens ; ALTMAN, Vladyslav ; SKODZIK, Jan ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk: Trust-by-Wire in Packet-Switched IPv6 Networks: Tools and FPGA Prototype for the IPclip System. In: *Proceedings of the 6th Annual IEEE Consumer Communications and Networking Conference (CCNC'09)*. Las Vegas, Nevada, USA, Januar 2009. – ISBN 978–1–4244–2309–5, S. 1–2 (zitiert auf S. 81, 178 und 227)

[DP98] DALLY, William J. ; POULTON, John W.: *Digital Systems Engineering*. New York, New York, USA : Cambridge University Press, 1998. – ISBN 0–521–59292–5 (zitiert auf S. 100)

[DS86] DALLY, William J. ; SEITZ, Charles L.: The Torus Routing Chip. In: *Distributed Computing* 1 (1986), Nr. 4, S. 187–196. – ISSN 0178–2770 (zitiert auf S. 100 und 143)

[DS87] DALLY, William J. ; SEITZ, Charles L.: Deadlock-Free Message Routing in Multiprocessor Interconnection Networks. In: *IEEE Transactions on Computers* 36 (1987), Nr. 5, S. 547–553. – ISSN 0018–9340 (zitiert auf S. 111)

[DSL03] DSL FORUM: *DSL Evolution - Architecture Requirements for the Support of QoS-Enabled IP Services*. Technischer Report TR-059. `http://www.dslforum.org`. Version: September 2003 (zitiert auf S. 43)

[DSL04a] DSL FORUM: *Broadband Remote Access Server (BRAS) Requirements Document*. Technischer Report TR-092. `http://www.dslforum.org`. Version: August 2004 (zitiert auf S. 43)

[DSL04b] DSL FORUM: *CPE WAN Mangement Protocol*. Technischer Report TR-069. `http://www.heise.de/netze/artikel/99963`. Version: Mai 2004 (zitiert auf S. 31)

[DSL06] DSL FORUM: *Migration to Ethernet-Based DSL Aggregation*. Technischer Report TR-101. `http://www.dslforum.org`. Version: April 2006 (zitiert auf S. 41 und 44)

[DSL07] DSL FORUM: *Broadband Multi-Service Architecture & Framework Requirements*. Technischer Report TR-144. `http://www.dslforum.org`. Version: August 2007 (zitiert auf S. 26)

[DT01] DALLY, William J. ; TOWLES, Brian: Route packets, not wires: on-chip interconnection networks. In: *Proceedings of the 38th Design Automation Conference (DAC'01)*. Las Vegas, Nevada, USA, Juni 2001. – ISBN 1–58113–297–2, S. 684–689 (zitiert auf S. 96)

[DT03] DALLY, William J. ; TOWLES, Brian: *Principles and Practices of Interconnection Networks*. San Francisco, Kalifornien, USA : Morgan Kaufmann Publishers Inc., 2003. – ISBN 0–12–200751–4 (zitiert auf S. 94, 107, 110, 120, 124, 129, 131, 138 und 143)

[Dua93] DUATO, José: A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks. In: *IEEE Transactions on Parallel and Distributed Systems* 4 (1993), Nr. 12, S. 1320–1331. – ISSN 1045–9219 (zitiert auf S. 111)

[DYN03] DUATO, José ; YALAMANCHILI, Sudhakar ; NI, Lionel: *Interconnection Networks: An Engineering Approach*. überarbeitete Auflage. Morgan Kaufmann Publishers Inc., 2003. – ISBN 1–55860–852–4 (zitiert auf S. 94, 107, 129, 131 und 143)

[EEL07] EHLIAR, Andreas ; EILERT, Johan ; LIU, Dake: A comparison of three FPGA optimized NoC architectures. In: *Proceedings of the Swedish System-on-Chip Conference (SSoCC'07)*. Gullmarsstrand, Fiskebäckski, Schweden, Mai 2007 (zitiert auf S. 103)

[EL07] EHLIAR, Andreas ; LIU, Dake: An FPGA Based Open Source Network-on-Chip Architecture. In: *Proceedings of the International Conference on Field Programmable Logic and Applications (FPL'07)*. Amsterdam, Niederlande, August 2007. – ISBN 978–1–4244–1060–6, S. 800–803 (zitiert auf S. 122)

[EMSW02] EKLUND, Carl ; MARKS, Roger B. ; STANWOOD, Kenneth L. ; WANG, Stanley: IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access. In: *IEEE Communications Magazine* 40 (2002), Juni, Nr. 6, S. 98–107. – ISSN 0163–6804 (zitiert auf S. 41)

[Erb05] ERBSCHLOE, Michael: *Trojans, Worms, and Spyware – A Computer Security Professional's Guide to Malicious Code*. Elsevier, 2005. – ISBN 0–7506–7848–8 (zitiert auf S. 28 und 84)

[Ern06] ERNO SALMINEN AND ARI KULMALA AND TIMO D. HÄMÄLÄINEN: *Survey of Network-on-chip Proposals*. White Paper, März 2006. – `http://www.ocpip.org/socket/whitepapers/OCP-IP_Survey_of_NoC_Proposals_White_Paper_April_2008.pdf` (zitiert auf S. 100)

[Fal03] FALLOWS, Deborah: Spam – How It Is Hurting Email and Degrading Life on the Internet / Ferris Research, Inc. Version: Oktober 2003. `http://www.pewinternet.org`. 2003. – Forschungsbericht. – PEW Internet & American Life Project (zitiert auf S. 28 und 84)

[FJ04a] FRANSSON, Pierre ; JONSSON, Andreas: End-to-End Measurements on Performance Penalties of IPv4 Options. In: *Proceedings of the 47th annual IEEE Global Telecommunications Conference (Globecom'04)*. Dallas, Texas, USA, November/Dezember 2004. – ISBN 0–7803–8794–5, S. 1441–1447 (zitiert auf S. 80)

[FJ04b] FRANSSON, Pierre ; JONSSON, Andreas: End-to-End Measurements on Performance Penalties of IPv4 Options / Luleå University of Technology. 2004 (2004:03). – Forschungsbericht. – ISSN 1402–1536 (zitiert auf S. 80)

[FKK96] FREIER, Alan O. ; KARLTON, Philip ; KOCHER, Paul C.: *The SSL Protocol Version 3.0*. Internet Draft, November 1996 (zitiert auf S. 50)

[Foa04] FOAG, Jürgen: *Speculative Protocol-Processing for High-Speed Packet Forwarding*. Lehrstuhl für Integrierte Schaltungen, Technische Universität München, Dissertation, Februar 2004 (zitiert auf S. 13)

[GBHW08] GOOSSENS, Kees ; BENNEBROEK, Martijn ; HUR, Jae Y. ; WAHLAH, Muhammad A.: Hardwired Networks on Chip in FPGAs to unify Data and Configuration Interconnects. In: *Proceedings of the 2nd ACM/IEEE International Symposium on Networks on Chip (NOCS'08)*. Newcastle, England, April 2008. – ISBN 978–0–7695–3098–7, S. 45–54 (zitiert auf S. 182)

[GCK07] GINDIN, Roman ; CIDON, Israel ; KEIDAR, Idit: NoC-Based FPGA: Architecture and Routing. In: *Proceedings of the First International Symposium on Networks-on-Chip (NOCS'07)*. Washington, DC, USA : IEEE Computer Society, 2007. – ISBN 0–7695–2773–6, S. 253–264 (zitiert auf S. 182)

[GCNS08] GUNARATNE, Chamara ; CHRISTENSEN, Kenneth ; NORDMAN, Bruce ; SUEN, Stephen: Reducing the Energy Consumption of Ethernet with Adaptive Link Rate (ALR). In: *IEEE Transactions on Computers* 57 (2008), April, Nr. 4, S. 448–461. – ISSN 0018–9340 (zitiert auf S. 16)

[GDR05] GOOSSENS, Kees ; DIELISSEN, John ; RĂDULESCU, Andrei: The Æthereal Network on Chip: Concepts, Architectures, and Implementations. In: *IEEE Design and Test of Computers* 22 (2005), September-Oktober, Nr. 5, S. 414–421. – ISSN 0740–7475 (zitiert auf S. 103 und 106)

[GG00] GUERRIER, Pierre ; GREINER, Alain: A Generic Architecture for On-chip Packet-switched Interconnections. In: *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'00)*. Paris, Frankreich, März 2000. – ISBN 1–58113–244–1, S. 250–256 (zitiert auf S. 96)

[GHR05] GOODMAN, Joshua ; HECKERMAN, David ; ROUNTHWAITE, Robert: Stopping Spam. In: *Scientific American* 292 (2005), April, Nr. 4, S. 42–49. – ISSN 0036–8733 (zitiert auf S. 84)

[Gil92] GILDER, George: Into The Fibersphere. In: *Forbes ASAP Magazine* (1992), Dezember 7. – `http://www.seas.upenn.edu/~gaj1/ggindex.html` (zitiert auf S. 104)

[Gil06] GILLESPIE, Tarleton: Engineering a Principle: 'End-to-End' in the Design of the Internet. In: *Social Studies of Science* 36 (2006), Juni, Nr. 3, S. 427–457. – ISSN 0306–3127 (zitiert auf S. 20)

[Gin03] GINOSAR, Ran: Fourteen Ways to Fool Your Synchronizer. In: *Proceedings of the 9th International Symposium on Asynchronous Circuits and Systems (ASYNC'03)*. Vancouver, British Columbia, Kanada, Mai 2003. – ISBN 0–7695–1898–2, S. 89–97 (zitiert auf S. 101)

[GM99] GUPTA, Pankaj ; MCKEOWN, Nick: Packet Classification on Multiple Fields. In: *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM)*. New York, New York, USA : ACM Press, August - September 1999. – ISBN 1–58113–135–6, S. 147–160 (zitiert auf S. 11)

[GM01] GUPTA, Pankaj ; MCKEOWN, Nick: Algorithms for Packet Classification. In: *IEEE Network Special Issue on Fast IP Packet Forwarding and Classification for Next Generation Internet Services* 15 (2001), März - April, Nr. 2, S. 24–32. – ISSN 0890–8044 (zitiert auf S. 11)

[GMB+08] GILABERT, Francisco ; MEDARDONI, Simone ; BERTOZZI, Davide ; BENINI, Luca ; GÓMEZ, María E. ; LÓPEZ, Pedro ; DUATO, José: Exploring High-Dimensional Topologies for NoC Design Through an Integrated Analysis and Synthesis Framework. In: *Proceedings of the 2nd ACM/IEEE International Symposium on Networks on Chip (NOCS'08)*. Newcastle, England, April 2008. – ISBN 978–0–7695–3098–7, S. 107–116 (zitiert auf S. 142)

[GMPW02] GOOSSENS, Kees ; MEERBERGEN, J. van ; PEETERS, A. ; WIELAGE, Paul: Networks on Silicon: Combining Best-Effort and Guaranteed Services. In: *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE'02)*. Paris, Frankreich, März 2002. – ISBN 0–7695–2288–2, S. 423–425 (zitiert auf S. 106)

[GOO] *Google Earth*. Google Inc. `http://earth.google.de`. – Version 4.1.7, Mai 2007 (zitiert auf S. 229)

[Gra05] GRASSERT, Frank: *Verlustleistungsoptimierte Schaltungstechniken für höchste Geschwindigkeiten*. Institut für Angewandte Mikroelektronik und Datentechnik, Universität Rostock, Dissertation, August 2005 (zitiert auf S. 145)

[Gri01] GRIES, Matthias: *Algorithm-Architecture Trade-offs in Network Processor Design*. Institut für Technische Informatik und Kommunikationsnetze, Eidgenössische Technische Hochschule Zürich, Dissertation, Mai 2001 (zitiert auf S. 13)

[GSS07] GOLDHAMMER, Klaus ; SCHMID, Michael ; STOCKBRÜGGER, Christoph: *Zukunft der TV-Übertragung*. Goldmedia GmbH – Media Consulting & Research, Studie. `http://www.goldmedia.com/publikationen`. Version: August 2007 (zitiert auf S. 26)

[Hag06] HAGEN, Silvia: *IPv6 Essentials*. Zweite Auflage. O'Reilly, 2006. – ISBN 0–596–10058–2 (zitiert auf S. 18 und 49)

[Hal05] HALSALL, Fred: *Computer Networking and the Internet*. Fünfte Auflage. Addison Wesley, 2005. – ISBN 0–321–26358–8 (zitiert auf S. 9, 21, 40 und 84)

[Har] HARRISON, Chris: *Internet Map*. `http://chrisharrison.net/projects/InternetMap` (zitiert auf S. 20)

[Has97] HASELOFF, Eilhard: Metastable Response in 5-V Logic Circuits / Texas Instruments, Inc. 1997 (SDYA006). – Forschungsbericht. – `http://focus.ti.com/lit/an/sdya006/sdya006.pdf` (zitiert auf S. 101)

[HB05] HOFMANN, Markus ; BEAUMONT, Leland: *Content Networking: Architecture, Protocols, and Practice*. Morgan Kaufmann Publishers Inc., 2005. – ISBN 1–55860–834–6 (zitiert auf S. 9)

[HC03] HODJAT, Babak ; CHEYER, Adam: Evolution of the Laws that Deal with the Utilization of Information Networks. In: *Proceedings of the 2003 BISC FLINT-CIBI International Joint Workshop on Soft Computing for Internet and BioInformatics*. Berkeley, Kalifornien, USA, Dezember 2003 (zitiert auf S. 3)

[Hea05] HEAVY READING: *Beyond High Speed Internet (HSI): Reinventing the B-RAS*. White Paper, Dezember 2005. – `http://www.alcatel.com/bnd/news/ip/heavy_reading/HeavyReading_BRAS_wp.pdf` (zitiert auf S. 41)

[Hei06] HEINRICH, Enrico: *Erweiterung einer bestehenden Implementierung eines NoC-Switches um das Konzept virtueller Kanäle*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Studienarbeit, Dezember 2006 (zitiert auf S. 111)

[Hei07] HEINRICH, Enrico: *Entwicklung und Implementierung eines GALS-Konzeptes für ein Network-on-Chip*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Diplomarbeit, September 2007 (zitiert auf S. 112)

[Hel03] HELD, Gilbert: *Ethernet Networks: Design, Implementation, Operation, Management*. John Wiley & Sons, Ltd., 2003. – ISBN 0–470–84476–0 (zitiert auf S. 14)

[Her07] HERNÁNDEZ BURGOS, SARA: *Realization of a Communication Protocol for Packet Processing Systems based on a Network-on-Chip*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Diplomarbeit, September 2007 (zitiert auf S. 160)

[Hig07] HIGH LEVEL POLICY TASK FORCE ON VOIP: *Common Position on VoIP (Draft)*. European Regulators Group (ERG), Nummer ERG (07) 56 Rev1. `http://erg.eu.int`. Version: 2007 (zitiert auf S. 82)

[Hil04] HILDEBRANDT, Jens: *Hardware-basiertes Task-Scheduling für Echtzeit-Systeme*. Institut für Angewandte Mikroelektronik und Datentechnik, Universität Rostock, Dissertation, Mai 2004 (zitiert auf S. 59)

[HKHT05] HECHT, Ronald ; KUBISCH, Stephan ; HERRHOLTZ, Andreas ; TIMMERMANN, Dirk: Dynamic Reconfiguration with hardwired Networks-on-Chip on future FPGAs. In: *Proceedings of the 15th International Conference on Field Programmable Logic and Applications (FPL'05)*. Tampere, Finnland, August 2005. – ISBN 0–7803–9362–7, S. 527–530 (zitiert auf S. 96 und 182)

[HKM+06] HECHT, Ronald ; KUBISCH, Stephan ; MICHELSEN, Harald ; ZEEB, Elmar ; TIMMERMANN, Dirk: A Distributed Object System Approach for Dynamic Reconfiguration. In: *Proceedings of the 20th IEEE International Parallel & Distributed Processing Symposium (IPDPS), 13th Reconfigurable Architectures Workshop (RAW'06)*. Rhodos, Griechenland, April 2006. – ISBN 1–4244–0054–6 (zitiert auf S. 96)

[HM05a] HU, Jingcao ; MARCULESCU, R.: Application-Specific Network-on-Chip Architecture Customization via Long-Range Link Insertion. In: *Proceedings of the 2005 IEEE/ACM International Conference on Computer-aided Design (ICCAD'05)*. San Jose, Kalifornien, USA, November 2005. – ISBN 0–7803–9254–X, S. 246–253 (zitiert auf S. 141)

[HM05b] HU, Jingcao ; MARCULESCU, Radu: Communication and Task Scheduling of Application-Specific Networks-on-Chip. In: *IEE Proceedings - Computers and Digital Techniques* 152 (2005), September, Nr. 5, S. 643–651. – ISSN 1350–2387 (zitiert auf S. 148)

[HM05c] HU, Jingcao ; MARCULESCU, Radu: Energy- and Performance-Aware Mapping for Regular NoC Architectures. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 24 (2005), April, Nr. 4, S. 551–562. – ISSN 0278–0070 (zitiert auf S. 148)

[HMH01] HO, Ron ; MAI, Kenneth W. ; HOROWITZ, Mark A.: The Future of Wires. In: *Proceedings of the IEEE* 89 (2001), April, Nr. 4, S. 490–504. – ISSN 0018–9219 (zitiert auf S. 95 und 96)

[HN06] HILTON, C. ; NELSON, B.: PNoC: a flexible circuit-switched NoC for FPGA-based systems. In: *IEE Proceedings – Computers and Digital Techniques* 153 (2006), Nr. 3, S. 181–188. – ISSN 1350–2387 (zitiert auf S. 121 und 122)

[HP97] HSU, Windsor W. ; PEIR, Jih-Kwon: Buses. In: ALLEN B. TUCKER, Jr. (Hrsg.): *The Computer Science and Engineering Handbook*. CRC Press, 1997. – ISBN 0–8493–2909–4 (zitiert auf S. 94)

[IEEa] *IEEE 802.3 Energy Efficient Ethernet Study Group*. – `http://grouper.ieee.org/groups/802/3/eee_study` (zitiert auf S. 16)

[IEEb] *IEEE 802.3 Higher Speed Study Group*. – `http://grouper.ieee.org/groups/802/3/hssg` (zitiert auf S. 15)

[IEEc] *IEEE P802.3ah Ethernet in the First Mile Task Force*. – `http://grouper.ieee.org/groups/802/3/efm` (zitiert auf S. 25)

[IEE98] IEEE COMPUTER SOCIETY: *IEEE Standard for Information Technology—Telecommunications and Informations Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 2: Logical Link Control. IEEE Std 802.2*. New York, New York, USA : Institute of Electrical and Electronics Engineers, 1998 `http://standards.ieee.org/getieee802/802.2.html`. – ISBN 1–55937–959–6 (zitiert auf S. 13)

[IEE02] IEEE COMPUTER SOCIETY: *IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. IEEE Std 802*. New York, New York, USA : Institute of Electrical and Electronics Engineers, 2002 `http://standards.ieee.org/getieee802/802.html`. – ISBN 0–7381–2941–0 (zitiert auf S. 13)

[IEE05] IEEE COMPUTER SOCIETY: *IEEE Standard for Information Technology—Telecommunications and Informations Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications. IEEE Std 802.3*. New York, New York, USA : Institute of Electrical and Electronics Engineers, 2005 `http://standards.ieee.org/getieee802/802.3.html`. – ISBN 0–7381–4741–9 (zitiert auf S. 13, 14, 15 und 238)

[IEE06a] IEEE COMPUTER SOCIETY: *IEEE Standards for Local and metropolitan area networks—Virtual Bridged Local Area Networks. IEEE Std 802.1Q*. New York, New York, USA : Institute of Electrical and Electronics Engineers, 2006 `http://standards.ieee.org/getieee802/802.1.html`. – ISBN 0–7381–4877–6 (zitiert auf S. 14)

[IEE06b] IEEE COMPUTER SOCIETY: *IEEE Standards for Local and metropolitan area networks—VirtualBridged Local Area Networks—Amendment 4: Prvider Bridges. IEEE Std 802.1ad*. New York, New York, USA : Institute of Electrical and Electronics Engineers, 2006 `http://standards.ieee.org/getieee802/802.1.html`. – ISBN 0–7381–4874–1 (zitiert auf S. 14)

[IEE08] *802.1ah - Provider Backbone Bridges*. Januar 2008. – `http://www.ieee802.org/1/pages/802.1ah.html` (zitiert auf S. 54)

[Inm] INMARSAT: *BGAN – Global voice and broadband data*. – `http://www.inmarsat.com/Services/Land/BGAN` (zitiert auf S. 10)

[Int] INTERNET ASSIGNED NUMBERS AUTHORITY (IANA): *IP Option Numbers*. http://www.iana.org/assignments/ip-parameters. – letzte Änderung Februar 2007 (zitiert auf S. 72)

[Int07a] INTERNATIONAL TELECOMMUNICATION UNION (ITU): *ITU Global Cybersecurity Agenda (GCA) – A Framework for International Cooperation in Cybersecurity*. Report. http://www.itu.int/cybersecurity/gca. Version: September 2007 (zitiert auf S. 28)

[Int07b] INTERNATIONAL TELECOMMUNICATION UNION (ITU): *The Future of Voice*. Workshop on the Future of Voice, Februar 2007. – http://www.itu.int/osg/spu/ni/voice/papers/Chairmans_Report.pdf (zitiert auf S. 24)

[Iro08] IRONPORT SYSTEMS, INC. & CISCO, INC.: *Internet Security Trends – A Report on Emerging Platforms for Spam, Viruses and Malware*. Report. http://www.ironport.com/securitytrends. Version: Januar 2008 (zitiert auf S. 28)

[Ise98] ISENBERG, David S.: The Dawn of the Stupid Network. In: *netWorker* 2 (1998), März, Nr. 1, S. 24–31. – ISSN 1091–3556 (zitiert auf S. 20 und 104)

[ISO94] ISO ; ISO/IEC (Hrsg.): *Information Technology—Open Systems Intercconnection—Basic Reference Model: The Basic Model*. 7498-1. Genf, Schweiz: ISO/IEC, 1994 (zitiert auf S. 8)

[ITRS] *International Technology Roadmap for Semiconductors*. Report Editions 1999-2007. http://www.itrs.net (zitiert auf S. 4)

[Jav07] JAVVIN TECHNOLOGIES, INC.: *Network Protocols Handbook*. Vierte Auflage. Javvin Technologies, Inc., 2007. – ISBN 978–1–60267–002–0 (zitiert auf S. 9, 10 und 16)

[Jos05] JOSHA BRONSON: *Protecting Your Network from ARP Spoofing-Based Attacks*. Foundstone, Inc. http://www.foundstone.com. Version: 2005. – White Paper (zitiert auf S. 46)

[JPX] *Japan Internet Exchange*. – http://www.jpix.ad.jp/en/technical/traffic.html (zitiert auf S. 182)

[JT03] JANTSCH, Axel (Hrsg.) ; TENHUNEN, Hannu (Hrsg.): *Networks on Chip*. Kluwer Academic Publishers, 2003. – ISBN 1–4020–7392–5 (zitiert auf S. 96 und 100)

[Jun08] JUNIPER NETWORKS, INC.: *Using PPPoE and IPoE in Ethernet Broadband Networks*. Whiter Paper Nr. 200187-002, Januar 2008. – http://www.juniper.net/solutions/literature/white_papers/200187.pdf (zitiert auf S. 77)

[JW02] JUNG, Volker (Hrsg.) ; WARNECKE, Hans-Jürgen (Hrsg.): *Handbuch für die Telekommunikation*. Zweite überarbeitete Auflage. Springer, 2002. – ISBN 3–540–42795–3 (zitiert auf S. 40)

[JW05] JERRAYA, Ahmed A. ; WOLF, Wayne: *Multiprocessor Systems-on-Chips*. Elsevier, 2005. – ISBN 0–12385–251–X (zitiert auf S. 94 und 97)

[KBD07] KIM, John ; BALFOUR, James ; DALLY, William J.: Flattened Butterfly Topology for On-Chip Networks. In: *Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO'07)*. Chicago, Illinois, USA, Dezember 2007. – ISBN 0–7695–3047–8, S. 172–182 (zitiert auf S. 143)

[KCHT07] KUBISCH, Stephan ; CORNELIUS, Claas ; HECHT, Ronald ; TIMMERMANN, Dirk: Mapping a Pipelined Data Path onto a Network-on-Chip. In: *Proceedings of the 2nd IEEE International Symposium on Industrial Embedded Systems (SIES'07)*. Lissabon, Portugal, Juli 2007. – ISBN 1–4244–0840–7, S. 178–185 (zitiert auf S. 104, 146 und 179)

[KDA07] KIM, John ; DALLY, William J. ; ABTS, Dennis: Flattened butterfly: a cost-efficient topology for high-radix networks. In: *ACM SIGARCH Computer Architecture News* 35 (2007), Nr. 2, S. 126–137. – ISSN 0163–5964 (zitiert auf S. 143)

[KGGV07] KRSTIĆ, Miloš ; GRASS, Eckhard ; GÜRKAYNAK, Frank K. ; VIVET, Pascal: Globally Asynchronous, Locally Synchronous Circuits: Overview and Outlook. In: *IEEE Design & Test of Computers* 24 (2007), Nr. 5, S. 430–441. – ISSN 0740–7475 (zitiert auf S. 101)

[KHST06] KUBISCH, Stephan ; HECHT, Ronald ; SALOMON, Ralf ; TIMMERMANN, Dirk: Intrinsic Flexibility and Robustness in Adaptive Systems: A Conceptual Framework. In: *Proceedings of the 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals/06)*. Logan, Utah, USA, Juli 2006. – ISBN 1–4244–0166–6, S. 98–103 (zitiert auf S. 182)

[KHT05] KUBISCH, Stephan ; HECHT, Ronald ; TIMMERMANN, Dirk: Design Flow on a Chip - An Evolvable HW/SW Platform. In: *Proceedings of the 2nd IEEE International Conference on Autonomic Computing (ICAC'05)*. Seattle, Washington, USA, Juni 2005. – ISBN 0–7695–2276–9, S. 393–394 (zitiert auf S. 182)

[KHT07] KUBISCH, Stephan ; HEINRICH, Enrico ; TIMMERMANN, Dirk: A Mesochronous Network-on-Chip for an FPGA. In: *Proceedings of the Annual Doctoral Workshop on Mathematical and Engineering Methods in Computer Science (MEMICS)*. Znojmo, Tschechische Republik, Oktober 2007. – ISBN 978–80–7355–077–6, S. 113–120 (zitiert auf S. 112, 146 und 179)

[KJM+02] KUMAR, Shashi ; JANTSCH, Axel ; MILLBERG, Mikael ; OBERG, Johny ; SOININEN, Juha-Pekka ; FORSELL, Martti ; TIENSYRJA, Kari ; HEMANI, Ahmed: A Network on Chip Architecture and Design Methodology. In: *Proceedings of the IEEE Computer Society Annual Symposium on VLSI (ISVLSI'02)*. Pittsburgh, Pennsylvania, USA, April 2002. – ISBN 0–7695–1486–3, S. 117–124 (zitiert auf S. 96)

[KK79] KERMANI, Parviz ; KLEINROCK, Leonard: Virtual Cut-Through: A New Computer Communication Switching Technique. In: *Computer Networks* 3 (1979), S. 267–286. – ISSN 1389–1286 (zitiert auf S. 100)

[Kle03] KLEINROCK, Leonard: An Internet Vision: The Invisible Global Infrastructure. In: *Ad Hoc Networks* 1 (2003), Nr. 1, S. 3–11. – ISSN 1570–8705 (zitiert auf S. 22, 40 und 181)

[Kle04] KLEINROCK, Leonard: The Internet Rules of Engagement: Then and Now. In: *Technology in Society – Technology and Science Entering the 21st Century* 26 (2004), Nr. 2-3, S. 193–207. – ISSN 0160–791X (zitiert auf S. 22, 28, 84 und 181)

[Kle08] KLEINROCK, Leonard: History of the Internet and Its Flexible Future. In: *IEEE Wireless Communications* 15 (2008), Nr. 1, S. 8–18. – ISSN 1536–1284 (zitiert auf S. 3 und 28)

[KNP07] KAUMANNS, Ralf ; NEUS, Andreas ; PÖRSCHMANN, Frank C.: *Konvergenz oder Divergenz? – Erwartungen und Präferenzen der Konsumenten an die Telekommunikations- und Medienangebote von morgen*. IBM Corporation, IBM Global Business Services, Studie, März 2007. – `http://www.ibm.com/services/de/bcs/html/konvergenz_divergenz.html` (zitiert auf S. 26 und 27)

[KP07] KUNDU, Partha (Hrsg.) ; PEH, Li-Shiuan (Hrsg.): *IEEE Micro, Vol. 27, Nr. 5, Special Issue: On-Chip Interconnects for Multicores*. IEEE Computer Society, 2007. ISSN 0272–1732 (zitiert auf S. 100)

[KPKJ08] KUMAR, Amit ; PEH, Li-Shiuan ; KUNDU, Partha ; JHA, Niraj K.: Toward Ideal On-Chip Communication Using Express Virtual Channels. In: *IEEE Micro* 28 (2008), Nr. 1, S. 80–90. – ISSN 0272–1732 (zitiert auf S. 142)

[KR07] KUON, Ian ; ROSE, Jonathan: Measuring the Gap Between FPGAs and ASICs. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 26 (2007), Februar, Nr. 2, S. 203–215. – ISSN 0278–0070 (zitiert auf S. 104)

[Kru06] KRUKOWSKI, Lukasz: *Providing additional Information on IP Level by using the IP Header's Option Field to enhance (Inter)Network Security*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Praktikumsarbeit, September 2006 (zitiert auf S. 227)

[Kub04] KUBISCH, Stephan: *Entwurf und Implementierung eines Network-on-Chip Prototypen*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Diplomarbeit, April 2004 (zitiert auf S. 99)

[Kub08] KUBISCH, Stephan: Networks: Complexity and Scalability. In: *TCPP PhD-Forum of the 22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08)*. Miami, Florida, USA, April 2008. – `http://www.ipdps.org/ipdps2008/2008_phdforum.html` (zitiert auf S. 146 und 177)

[KWD+06] KUBISCH, Stephan ; WIDIGER, Harald ; DUCHOW, Daniel ; TIMMERMANN, Dirk ; BAHLS, Thomas: Wirespeed MAC Address Translation and Traffic Management in Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006 (zitiert auf S. 45, 52, 57, 59 und 177)

[KWD+07] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; LANGE, Christian ; RÖWER, Oliver: Configuration Tool and FPGA-Prototype of a Hardware Packet Processing System. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE'07), University Booth Proceedings on CD-Rom*. Nizza, Frankreich, April 2007 (zitiert auf S. 52, 60, 178 und 223)

[KWD+08a] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: Countering Phishing Threats with Trust-by-Wire in Packet-switched IP Networks – A Conceptual Framework. In: *Proceedings of the 22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08), 4th International Workshop on Security in Systems and Networks (SSN'08) (auf CD-Rom)*. Miami, Florida, USA, April 2008. – ISBN 978–1–4244–1694–3 (zitiert auf S. 87, 88, 90 und 178)

[KWD+08b] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: Trust-by-Wire in Packet-switched IP Networks: Calling Line Identification Presentation for IP. In: *Proceedings of the 1st ITU-T Kaleidoscope Conference: Innovations in Next Generation Networks - Future Network and Services (K-INGN'08)*. Genf, Schweiz, Mai 2008. – ISBN 92–61–12441–0, S. 375–382 (zitiert auf S. 70, 77, 81, 82, 90 und 178)

[KWD+08c] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; TIMMERMANN, Jens Schulzand D. ; BAHLS, Thomas ; DUCHOW, Daniel: Complementing E-Mails with Distinct, Geographic Location Information in Packet-switched IP Networks. In: *Proceedings of the 2008 MIT Spam Conference (auf CD-Rom)*. Cambridge, Massachusetts, USA, März 2008. – `http://www.spamconference.org` (zitiert auf S. 84, 86, 90 und 178)

[KWHT07] KUBISCH, Stephan ; WIDIGER, Harald ; HECHT, Ronald ; TIMMERMANN, Dirk: Network-on-Chip Communication Grids for High Performance Packet Processing. In: *Proceedings of the 2007 ACM/SIGDA 15th International Symposium on Field Programmable Gate Arrays (FPGA'07)*. Monterey, Kalifornien, USA, Februar 2007. – ISBN 978–1–59593–600–4, S. 228 (zitiert auf S. 146 und 178)

[KWT+07] KUBISCH, Stephan ; WIDIGER, Harald ; TIMMERMANN, Dirk ; DUCHOW, Daniel ; BAHLS, Thomas: sMAT – A Simplified MAC Address Translation Scheme. In: *Proceedings of the 2007 15th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN'07) (auf CD-Rom)*. Princeton, New Jersey, USA, Juni 2007. – ISBN 1–4244–1100–9. – `http://www.ieee-lanman.org` (zitiert auf S. 52, 57 und 177)

[LCC+97] LEINER, Barry M. ; CERF, Vinton G. ; CLARK, David D. ; KAHN, Robert E. ; KLEINROCK, Leonard ; LYNCH, Daniel C. ; POSTEL, Jon ; ROBERTS, Lawrence G. ; WOLFF, Stephen S.: The Past and Future History of the Internet. In: *Communications of the ACM* 40 (1997), Nr. 2, S. 102–108. – ISSN 0001–0782 (zitiert auf S. 22, 28 und 183)

[LCOM07] LEE, Hyung G. ; CHANG, Naehyuck ; OGRAS, Umit Y. ; MARCULESCU, Radu: On-chip Communication Architecture Exploration: A Quantitative Evaluation of Point-to-Point, Bus, and Network-on-Chip Approaches. In: *ACM Transactions on Design Automation of Electronic Systems* 12 (2007), August, Nr. 3, S. 21–40. – ISSN 1084–4309 (zitiert auf S. 95)

[Lea07] LEAVITT, Neal: Vendors Fight Spam's Sudden Rise. In: *IEEE Computer* 40 (2007), März, Nr. 3, S. 16–19. – ISSN 0018–9162 (zitiert auf S. 84)

[Lin06] LIN, Chinlon (Hrsg.): *Broadband Optical Access Networks and Fiber-to-the-Home – Systems Technologies and Deployment Strategies*. John Wiley & Sons, Ltd., 2006. – ISBN 0–470–09478–8 (zitiert auf S. 25 und 41)

[LJS06] LI, Qing ; JINMEI, Tatuya ; SHIMA, Keiichi: *IPv6 Core Protocols Implementation*. Morgan Kaufmann Publishers Inc., 2006. – ISBN 0–12–447751–8 (zitiert auf S. 18 und 49)

[LO08] LEVINSON, David ; ODLYZKO, Andrew M.: Too expensive to meter: The Influence of Transaction Costs in Transportation and Communication. In: *Philosophical Transactions of the Royal Society, Special Issue on Networks: Modeling and Control* 336 (2008), Juni, Nr. 1872, S. 2033–2046. – ISSN 1471–2962. – `http://www.dtc.umn.edu/~odlyzko/doc/metering-expensive.pdf` (zitiert auf S. 27)

[Loc86] LOCKE, Carey D.: *Best-effort decision-making for real-time scheduling*. Pittsburgh, Pennsylvania, USA, Computer Science Department, Carnegie Mellon University, Dissertation CMU-CS-86-134, 1986 (zitiert auf S. 59)

[Mat97] MATZKE, Doug: Will Physical Scalability Sabotage Performance Gains? In: *IEEE Computer* 30 (1997), September, Nr. 9, S. 37–39. – ISSN 0018–9162 (zitiert auf S. 95)

[Max04] MAXFIELD, Clive: *The Design Warrior's Guide to FPGAs – Devices, Tools and Flows*. Elsevier, 2004. – ISBN 0–7506–7604–3 (zitiert auf S. 103)

[MB00] MCKNIGHT, Lee W. ; BOROUMAND, Jahangir: Pricing Internet Services: Approaches and Challenges. In: *IEEE Computer* 33 (2000), Nr. 2, S. 128–129. – ISSN 0018–9162 (zitiert auf S. 27)

[Met73] METCALFE, Robert M.: *Packet Communication*. Cambridge, Massachusetts, USA : Technischer Report MIT/LCS/TR-114, 1973. – Massachusetts Institute of Technology, Cambridge, Massachusetts, USA (zitiert auf S. 8)

[Met07] METRO ETHERNET FORUM: *Carrier Ethernet – The technology of choice for Access networks*. White Paper. http://www.metroethernetforum.org. Version: März 2007 (zitiert auf S. 41)

[MHRSW05] MINTZ-HABIB, Matthew ; RAWAT, Anshuman ; SCHULZRINNE, Henning ; WU, Xiaotao: A VoIP Emergency Services Architecture and Prototype. In: *14th International Conference on Computer Communications and Networks*. San Diego, Kalifornien, USA, Oktober 2005. – ISBN 0–7803–9428–3, S. 523–528 (zitiert auf S. 82)

[Mil07] MILLER, Jim: Stumbling Forward into the Connected Future. In: *IEEE Internet Computing* 11 (2007), Nr. 5, S. 82–85. – ISSN 1089–7801 (zitiert auf S. 3)

[MIRO] *Miro - Free, Open Source Internet TV and Video Player*. http://www.getmiro.com (zitiert auf S. 27)

[ML405] XILINX, INC.: *Virtex-4 FX ML405 Evaluation Platform*. – http://www.xilinx.com/products/devkits/HW-V4-ML405-UNI-G.htm (zitiert auf S. 60 und 223)

[MM04] MURALI, Srinivasan ; MICHELI, Giovanni D.: Bandwidth-Constrained Mapping of Cores onto NoC Architectures. In: *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'04)*. Paris, Frankreich, Februar 2004. – ISBN 0–7695–2085–5–2, S. 20896 (zitiert auf S. 148)

[MNT+04] MILLBERG, Mikael ; NILSSON, Erland ; THID, Rikard ; KUMAR, Shashi ; JANTSCH, Axel: The Nostrum Backbone – a Communication Protocol Stack for Networks on Chip. In: *Proceedings of the 17th International Conference on VLSI Design (VLSI Design'04)*. Mumbai, Indien, Januar 2004. – ISBN 0–7695–2072–3, S. 693–696 (zitiert auf S. 103)

[Mog] *Mogulus – Broadcast Live*. – http://www.mogulus.com (zitiert auf S. 27)

[Moo65] MOORE, Gordon E.: Cramming more components onto integrated circuits. In: *Electronics* 38 (1965), April, Nr. 8, S. 114–117. – http://download.intel.com/research/silicon/moorespaper.pdf (zitiert auf S. 3)

[Mor05] MORGAN STANLEY: *Global Telecom Outlook Day*. Marktanalyse und -report, Oktober 2005. – http://www.morganstanley.com/institutional/techresearch/pdfs/MS_Global_Telecom_Outlook101705.pdf (zitiert auf S. 22)

[Mor07] MORGAN STANLEY: *Technology / Internet Trends*. Marktanalyse und -report, November 2007. – http://www.morganstanley.com/institutional/techresearch/pdfs/Future-of-Media-110807.pdf (zitiert auf S. 22)

[MOZ06] MARCULESCU, Radu ; OGRAS, Umit Y. ; ZAMORA, Nicholas H.: Computation and Communication Refinement for Multiprocessor SoC Design: A System-Level Perspective. In: *ACM Transactions on Design Automation of Electronic Systems (TODAES)* 11 (2006), Juli, Nr. 3, S. 564–592. – ISSN 1084–4309 (zitiert auf S. 148)

[MR07] MEDHI, Deepankar ; RAMASAMY, Karthikeyan: *Network Routing – Algorithms, Protocols, and Architectures*. Morgan Kaufmann, 2007. – ISBN 0–12–088588–3 (zitiert auf S. 109)

[MS06] MITIĆ, Milica ; STOJČEV, Mile: An Overview of On-Chip Buses. In: *Facta Universitates, Series: Electronics and Energetics* 19 (2006), Dezember, Nr. 3, S. 405–428. – ISSN 0353–3670. – http://factaee.elfak.ni.ac.yu (zitiert auf S. 94)

[MSCL06] MAK, Terence S. T. ; SEDCOLE, Pete ; CHEUNG, Peter Y. K. ; LUK, Wayne: On-FPGA Communication Architectures and Design Factors. In: *Proceedings of the 16th International Conference on Field Programmable Logic and Applications (FPL'06)*. Madrid, Spanien, August 2006. – ISBN 1–59593–057–4, S. 452–457 (zitiert auf S. 104)

[MTCM06] MELLO, Aline ; TEDESCO, Leonel ; CALAZANS, Ney ; MORAES, Fernando: Evaluation of current QoS Mechanisms in Networks on Chip. In: *Proceedings of the International Symposium on System-on-Chip (SoC'06)*. Tampere, Finnland, November 2006. – ISBN 1–4244–0622–6, S. 1–4 (zitiert auf S. 112)

[Nat02] NATIONAL MARINE ELECTRONICS ASSOCIATION (NMEA): *NMEA 0183 Standard*. Januar 2002. – `http://www.kowoma.de/gps/zusatzerklaerungen/NMEA.htm` (zitiert auf S. 73)

[Nel03] NELSON, Mårten: *Anti-Spam for Businesses and ISPs: Market Size 2003-2008*. `http://www.ferris.com`. Version: April 2003 (zitiert auf S. 84)

[NIM00] NATIONAL IMAGERY AND AGENCY (NIMA): Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems. Version: Januar 2000. `http://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf`. 2000 (TR 8350.2). – Forschungsbericht. – letzte Änderung Juni 2004 (zitiert auf S. 73 und 226)

[NM93] NI, Lionel M. ; MCKINLEY, Philip K.: A survey of wormhole routing techniques in direct networks. In: *IEEE Computer* 26 (1993), Februar, Nr. 2, S. 62–76. – ISSN 0018–9162 (zitiert auf S. 100)

[NN05] NAKHJIRI, Mahjid ; NAKHJIRI, Mahsa: *AAA and Network Security for Mobile Access*. John Wiley & Sons Ltd, 2005. – ISBN 0–470–01194–7 (zitiert auf S. 31)

[NOC08] *Proceedings of the 2nd, ACM/IEEE International Symposium on Networks-on-Chips (NOCS'08), 5-6 April 2008, Newcastle University, England*. IEEE Computer Society, 2008. – ISBN 978–0–7695–3098–7 (zitiert auf S. 100)

[Nok07] NOKIA SIEMENS NETWORKS: *Bradband Access for All – A Brief Technology Guide*. White Paper, 2007. – `http://www.nokiasiemensnetworks.com` (zitiert auf S. 40)

[Nor07a] NORTEL NETWORKS: *Hyperconnectivity: An Unstoppable Force of Change*. White Paper, 2007. – `http://www.hyperconnectivity.com` (zitiert auf S. 3 und 22)

[Nor07b] NORTEL NETWORKS: *Provider Backbone Bridges bring massive Service Scalability to Ethernet*. White Paper, 2007. – `http://www.nortel.com/solutions/collateral/nn120620.pdf` (zitiert auf S. 54)

[Nor07c] NORTEL NETWORKS: *Service Delivery Technologies for Metro Ethernet Networks*. White Paper, 2007. – `http://www.nortel.com/solutions/optical/collateral/nn105600.pdf` (zitiert auf S. 54)

[NPI+08] NEDEVSCHI, Sergiu ; POPA, Lucian ; IANNACCONE, Gianluca ; RATNASAMY, Sylvia ; WETHERALL, David: Reducing Network Energy Consumption via Sleeping and Rate-Adaptation. In: *Proceedings of the 5th USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI'08)*. San Francisco, Kalifornien, USA, April 2008, S. 323–336. – `http://www.usenix.org/event/nsdi08/tech` (zitiert auf S. 16)

[Odl01a] ODLYZKO, Andrew M.: Content is Not king. In: *First Monday* 6 (2001), Februar, Nr. 2. – `http://www.firstmonday.org/ISSUES/issue6_2/odlyzko/` (zitiert auf S. 3)

[Odl01b] ODLYZKO, Andrew M.: Internet Pricing and the History of Communications. In: *Computer Networks* 36 (2001), Nr. 5–6, S. 493–517. – ISSN 1389–1286 (zitiert auf S. 27)

[Odl03] ODLYZKO, Andrew M.: Internet Traffic Growth: Sources and Implications, SPIE, 2003, S. 1–15. – `http://link.aip.org/link/?PSI/5247/1/1` (zitiert auf S. 24)

[OHM05] OGRAS, Umit Y. ; HU, JJingcao ; MARCULESCU, Radu: Key Research Problems in NoC Design: A Holistic Perspective. In: *Proceedins of the 3rd IEEE/ACM/IFIP International Conference on Hardware-Software Codesign and System Synthesis (CODES+ISSS'05)*. Jersey City, New Jersey, USA, September 2005. – ISBN 1–59593–161–9, S. 69–74 (zitiert auf S. 105, 139 und 148)

[Oll04] OLLMANN, Gunter: *The Phishing Guide – Understanding and Preventing Phishing Attacks*. NGS Next Generation Security Software Ltd. `http://www.ngsconsulting.com`. Version: September 2004. – White Paper (zitiert auf S. 87)

[OM06] OGRAS, Umit Y. ; MARCULESCU, Radu: 'It's a Small World After All': NoC Performance Optimization Via Long-Range Link Insertion. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 14 (2006), Juli, Nr. 7, S. 693–706. – ISSN 1063–8210 (zitiert auf S. 141)

[OMLC06] OGRAS, Umit Y. ; MARCULESCU, Radu ; LEE, Hyung G. ; CHANG, Naehyuck: Communication architecture optimization: making the shortest path shorter in regular networks-on-chip. In: *Proceedings of the Conference on Design, Automation and Test in Europe (DATE'06)*, 2006. – ISBN 3–9810801–0–6, S. 712–717 (zitiert auf S. 141)

[Par05] PARK, Kun I.: *QoS in Packet Networks*. Springer, 2005. – ISBN 0–387–23390–3 (zitiert auf S. 12)

[PD04] PETERSON, Larry L. ; DAVIE, Bruce S.: *Computernetze: eine systemorientierte Einführung*. Deutsche Ausgabe der 3. amerikanischen Auflage. dpunkt.verlag, 2004. – ISBN 3–89864–242–9 (zitiert auf S. 9, 10 und 15)

[PGIB99] PUENTE, Valentin ; GREGORIO, José A. ; IZU, Cruz ; BEIVIDE, Ramón: Impact of the Head-of-Line Blocking on Parallel Computer Networks: Hardware to Applications. In: *Proceedings of the 5th International Euro-Par Conference on Parallel Processing (Euro-Par'99)*. London, England : Springer-Verlag, 1999. – ISBN 3–540–66443–2, S. 1222–1230 (zitiert auf S. 111)

[PM04] PIÓRO, Michal ; MEDHI, Deepankar: *Routing, Flow, and Capacity Design in Communication and Computer Networks*. Morgan Kaufmann Publishers Inc., 2004. – ISBN 0–12–557189–5 (zitiert auf S. 13)

[PNK+06] PARK, Dongkook ; NICOPOULOS, Chrysostomos A. ; KIM, Jongman ; VIJAYKRISHNAN, N. ; DAS, Chita R.: A Distributed Multi-Point Network Interface for Low-Latency, Deadlock-Free On-Chip Interconnects. In: *Proceedings of the International Conference on Nano-Networks (Nano-Net'06)*. Lausanne, Schweiz, September 2006. – ISBN 1–4244–0391–X, S. 1–6 (zitiert auf S. 142)

[Por07] PORTA, Thomas F. L. (Hrsg.): *IEEE Applications and Practice Magazine, Vol. 45, Nr. 3, Special Issue: 100 Gigabit Ethernet*. IEEE Communications Society, 2007. ISSN 0163–6804 (zitiert auf S. 15)

[PP08] PATEL-PREDD, Prachi: Energy-Efficient Ethernet. In: *IEEE Spectrum* 45 (2008), Mai, Nr. 5 (INT), S. 9. – ISSN 0018–9235 (zitiert auf S. 16)

[Pri07] PRIBBERNOW, Frank: *Konzeption eines Routers für on-chip Netzwerke zur Reduzierung der Verlustleistung*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Studienarbeit, August 2007 (zitiert auf S. 124)

[PZWF07] PARAMESWARAN, Manoj ; ZHAO, Xia ; WHINSTON, Andrew B. ; FANG, Fang: Reengineering the Internet for Better Security. In: *IEEE Computer* 40 (2007), Januar, Nr. 1, S. 40–44. – ISSN 0018–9162 (zitiert auf S. 71)

[QT] *Qt by Trolltech*. – `http://www.trolltech.com` (zitiert auf S. 223 und 228)

[RCN03] RABAEY, Jan M. ; CHANDRAKASAN, Anantha ; NIKOLIC, Borivoje: *Digital Integrated Circuits – A Design Perspective*. Zweite Auflage. Prentice Hall, 2003. – ISBN 0–13–090996–3 (zitiert auf S. 145)

[RDP+05] RĂDULESCU, Andrei ; DIELISSEN, John ; PESTANA, Santiago G. ; GANGWAL, Om P. ; RIJPKEMA, Edwin ; WIELAGE, Paul ; GOOSSENS, Kees: An Efficient On-Chip Network Interface Offering Guaranteed Services, Shared-Memory Abstraction, and Flexible Network Programming. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 24 (2005), Januar, Nr. 1, S. 4–17. – ISSN 0278–0070 (zitiert auf S. 106)

[RE94] RAMANY, Swaminathan ; EAGER, Derek: The Interaction Between Virtual Channel Flow Control and Adaptive Routing in Wormhole Networks. In: *Proceedings of the 8th International Conference on Supercomputing (ICS'94)*. Manchester, England, Juli 1994. – ISBN 0–89791–665–4, S. 136–145 (zitiert auf S. 111)

[RFC0791] INFORMATION SCIENCES INSTITUTE, UNIVERSITY OF SOUTHERN CALIFORNIA: *Internet Protocol – DARPA Internet Program Protocol Specification*. RFC 791 (Standard), September 1981. `http://tools.ietf.org/html/rfc791` (zitiert auf S. 16 und 71)

[RFC0792] POSTEL, Jon: *Internet Control Message Protocol*. RFC 792, September 1981. `http://tools.ietf.org/html/rfc792` (zitiert auf S. 78)

[RFC0822] CROCKER, David H.: *Standard for the Format of ARPA Iinternet Text Messages*. RFC 822, August 1982. `http://tools.ietf.org/html/rfc822` (zitiert auf S. 85)

[RFC0826] PLUMMER, David C.: *An Ethernet Address Resolution Protocol – or – Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware*. RFC 826, November 1982. `http://tools.ietf.org/html/rfc826` (zitiert auf S. 44, 49 und 215)

[RFC0903] FINLAYSON, Ross ; MANN, Timothy ; MOGUL, Jeffrey ; THEIMER, Marvin: *A Reverse Address Resolution Protocol*. RFC 903, Juni 1984. `http://tools.ietf.org/html/rfc903` (zitiert auf S. 44, 49 und 215)

[RFC1157] CASE, Jeffrey D. ; FEDOR, Mark ; SCHOFFSTALL, Martin L. ; DAVIN, James R.: *A Simple Network Management Protocol (SNMP)*. RFC 1157, Mai 1990. `http://tools.ietf.org/html/rfc1157` (zitiert auf S. 47)

[RFC1191] MOGUL, Jeffrey C. ; DEERING, Steeve E.: *Path MTU Discovery*. RFC 1191, November 1990. `http://tools.ietf.org/html/rfc1191` (zitiert auf S. 77)

[RFC1242] BRADNER, Scott: *Benchmarking Terminology for Network Interconnection Devices*. RFC 1242, Juli 1991. `http://tools.ietf.org/html/rfc1242` (zitiert auf S. 60, 62, 164 und 167)

[RFC1752] BRADNER, Scott ; MANKIN, Allison: *The Recommendation for the IP Next Generation Protocol*. RFC 1752, Januar 1995. `http://tools.ietf.org/html/rfc1752` (zitiert auf S. 18)

[RFC1958] CARPENTER, Brian E.: *Architectural Principles of the Internet*. RFC 1958, Juni 1996. http://tools.ietf.org/html/rfc1958 (zitiert auf S. 16, 22 und 24)

[RFC2131] DROMS, Ralph: *Dynamic Host Configuration Protocol*. RFC 2131, März 1997. http://tools.ietf.org/html/rfc2131 (zitiert auf S. 49 und 215)

[RFC2311] DUSSE, Steve ; HOFFMAN, Paul ; RAMSDELL, Blake ; LUNDBLADE, Laurence ; REPKA, Lisa: *S/MIME Version 2 Message Specification*. RFC 2311, März 1998. http://tools.ietf.org/html/rfc2311 (zitiert auf S. 52)

[RFC2460] DEERING, Stephen E. ; HINDEN, Robert M.: *Internet Protocol, Version 6 (IPv6) Specification*. RFC 2460, Dezember 1998. http://tools.ietf.org/html/rfc2460 (zitiert auf S. 18)

[RFC2461] NARTEN, Thomas ; NORDMARK, Erik ; SIMPSON, William A.: *Neighbor Discovery for IP Version 6 (IPv6)*. RFC 2461, Dezember 1998. http://tools.ietf.org/html/rfc2461 (zitiert auf S. 49 und 218)

[RFC2463] CONTA, Alex ; DEERING, Stephen: *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*. RFC 2463, Dezember 1998. http://tools.ietf.org/html/rfc2463 (zitiert auf S. 49 und 215)

[RFC2464] CRAWFORD, Matt: *Transmission of IPv6 Packets over Ethernet Networks*. RFC 2464, Dezember 1998. http://tools.ietf.org/html/rfc2464 (zitiert auf S. 18)

[RFC2544] BRADNER, Scott ; MCQUAID, Jim: *Benchmarking Methodology for Network Interconnect Devices*. RFC 2544, März 1999. http://tools.ietf.org/html/rfc2544 (zitiert auf S. 60, 62, 164 und 167)

[RFC2697] HEINANEN, Juha ; GUERIN, Roch: *A Single Rate Three Color Marker*. RFC 2697, September 1999. http://tools.ietf.org/html/rfc2697 (zitiert auf S. 59 und 62)

[RFC2698] HEINANEN, Juha ; GUERIN, Roch: *A Two Rate Three Color Marker*. RFC 2698, September 1999. http://tools.ietf.org/html/rfc2698 (zitiert auf S. 59 und 62)

[RFC2775] CARPENTER, Brian E.: *Internet Transparency*. RFC 2775, Februar 2000, . – http://tools.ietf.org/html/rfc2775 (zitiert auf S. 22)

[RFC2821] KLENSIN, John C.: *Simple Mail Transfer Protocol*. RFC 2821, April 2001, . – http://tools.ietf.org/html/rfc2821 (zitiert auf S. 30, 84 und 85)

[RFC2889] MANDEVILLE, Robert ; PERSER, Jerry: *Benchmarking Methodology for LAN Switching Devices*. RFC 2889, August 2000. http://tools.ietf.org/html/rfc2889 (zitiert auf S. 60)

[RFC3022] SRISURESH, Pyda ; EGEVANG, Kjeld B.: *Traditional IP Network Address Translator (Traditional NAT)*. RFC 3022, Januar 2001. http://tools.ietf.org/html/rfc3022 (zitiert auf S. 45)

[RFC3031] ROSEN, Eric C. ; VISWANATHAN, Arun ; CALLON, Ross: *Multiprotocol Label Switching Architecture*. RFC 3031, Januar 2001. http://tools.ietf.org/html/rfc3031 (zitiert auf S. 54)

[RFC3261] ROSENBERG, Jonathan ; SCHULZRINNE, Henning ; CAMARILLO, Gonzalo ; JOHNSTON, Alan ; PETERSON, Jon ; SPARKS, Robert ; HANDLEY, Mark ; SCHOOLER, Eve: *SIP: Session Initiation Protocol*. RFC 3261, Juni 2002. http://tools.ietf.org/html/rfc3261 (zitiert auf S. 83)

[RFC3315] DROMS, Ralph ; BOUND, Jim ; VOLZ, Bernie ; LEMON, Ted ; PERKINS, Charles E. ; CARNEY, Mike: *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*. RFC 3315, Juli 2003. http://tools.ietf.org/html/rfc3315 (zitiert auf S. 49 und 215)

[RFC3410] CASE, Jeffrey ; MUNDY, Russ ; PARTAIN, David ; STEWART, Bob: *Introduction and Applicability Statements for Internet Standard Management Framework*. RFC 3410, Dezember 2002. http://tools.ietf.org/html/rfc3410 (zitiert auf S. 47)

[RFC3439] BUSH, Randy ; MEYER, David: *Some Internet Architectural Guidelines and Philosophy*. RFC 3439, Dezember 2002. http://tools.ietf.org/html/rfc3439 (zitiert auf S. 22)

[RFC3550] SCHULZRINNE, Henning ; CASNER, Stephen L. ; FREDERICK, Ron ; JACOBSON, Van: *RTP: A Transport Protocol for Real-Time Applications*. RFC 3550, Juli 2003. http://tools.ietf.org/html/rfc3550 (zitiert auf S. 83)

[RFC3761] FÄLTSTRÖM, Patrik ; MEALLING, Michael: *The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)*. RFC 3761, April 2004. http://tools.ietf.org/html/rfc3761 (zitiert auf S. 69)

[RFC3825] POLK, James M. ; SCHNIZLEIN, John ; LINSNER, Marc: *Dynamic Host Configuration Protocol Option for Coordiante-based Location Configuration Information*. RFC 3825, Juli 2004. http://tools.ietf.org/html/rfc3825 (zitiert auf S. 225)

[RFC3912] DAIGLE, Leslie: *WHOIS Protocol Specification*. RFC 3912, September 2004. http://tools.ietf.org/html/rfc3912 (zitiert auf S. 86)

[RFC4120] NEUMAN, Clifford ; YU, Tom ; HARTMAN, Sam ; RAEBURN, Kenneth: *The Kerberos Network Authentication Service (V5)*. RFC 4120, Juli 2005. http://tools.ietf.org/html/rfc4120 (zitiert auf S. 52)

[RFC4301] KENT, Stephen ; SEO, Karen: *Security Architecture for the Internet Protocol*. RFC 4301, Dezember 2005. http://tools.ietf.org/html/rfc4301 (zitiert auf S. 50 und 105)

[RFC4302] KENT, Stephen: *IP Authentication Header*. RFC 4302, Dezember 2005. http://tools.ietf.org/html/rfc4302 (zitiert auf S. 50)

[RFC4303] KENT, Stephen: *IP Encapsulating Security Payload (ESP)*. RFC 4303, Dezember 2005. http://tools.ietf.org/html/rfc4303 (zitiert auf S. 50)

[RFC4306] KAUFMAN, Charlie: *Internet Key Exchange (IKEv2) Protocol*. RFC 4306, Dezember 2006. http://tools.ietf.org/html/rfc4306 (zitiert auf S. 50)

[RFC4346] DIERKS, Tim ; RESCORLA, Eric: *The Transport Layer Security (TLS) Protocol Version 1.1*. RFC 4346, April 2006. http://tools.ietf.org/html/rfc4346 (zitiert auf S. 50)

[RFC4447] MARTINI, Luca ; ROSEN, Eric C. ; EL-AAWAR, Nasser ; SMITH, Toby ; HERON, Giles: *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*. RFC 4447, August 2006. http://tools.ietf.org/html/rfc4447 (zitiert auf S. 59)

[RFC4448] MARTINI, Luca ; ROSEN, Eric C. ; EL-AAWAR, Nasser ; HERON, Giles: *Encapsulation Methods for Transport of Ethernet over MPLS Networks*. RFC 4448, April 2006. http://tools.ietf.org/html/rfc4448 (zitiert auf S. 59)

[RFC4862] THOMSON, Susan ; NARTEN, Thomas ; JINMEI, Tatuya: *IPv6 Stateless Address Autoconfiguration*. RFC 4862, September 2007. `http://tools.ietf.org/html/rfc4862` (zitiert auf S. 49 und 218)

[Rhi08] RHINOW, Grit: *Trust-by-Wire in paketvermittelten IP-Netzen: Analyse und Konzept zur Migration des IPclip-Mechanismus von IPv4 nach IPv6*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Kleiner Beleg, März 2008 (zitiert auf S. 80)

[RHPG04] RODRÍGUEZ, José Manuel A. ; HERRAIZ, Antonio G. ; PELAYO, Juan Antonio C. ; GARCÍA, Alvaro P.: Layer 2 VPN architectures and operation. In: *Proceedings of the IASTED International Conference on Communication Systems and Networks*. Marbella, Spanien, September 2004. – ISBN 0–88986–456–X, S. 208–213 (zitiert auf S. 54)

[Rod85] RODGERS, David P.: Improvements in multiprocessor system design. In: *SIGARCH Computer Architecture News* 13 (1985), Nr. 3, S. 225–231. – ISSN 0163–5964 (zitiert auf S. 183)

[RP08] ROSEN, Brian ; POLK, James: *Best Current Practice for Communications Services in Support of Emergency Calling*. Internet Draft, Juli 2008 (zitiert auf S. 82)

[RSC98] REED, David P. ; SALTZER, Jerome H. ; CLARK, David D.: Active Networking and End-to-End Arguments. In: *IEEE Network* 12 (1998), Mai/Juni, Nr. 3, S. 67–71. – ISSN 0890–8044 (zitiert auf S. 20 und 104)

[RSPN08] ROSEN, Brian ; SCHULZRINNE, Henning ; POLK, James ; NEWTON, Andrew: *Framework for Emergency Calling in Internet Multimedia*. Internet Draft, Februar 2008. – `http://tools.ietf.org/html/draft-ietf-ecrit-framework-05` (zitiert auf S. 82)

[Rus95] RUSSEL, Peter: *The Global Brain Awakens: Our Next Evolutionary Leap*. Zweite Edition. Global Brain, Inc., 1995. – ISBN 978–086315–616–8 (zitiert auf S. 22)

[RW03] ROSSI, Mattia ; WELZL, Michael: *On the Impact of IP Option Processing*. Leopold-Franzens-Universität Innsbruck, Preprint-Reihe des Fachbereichs Mathematik & Informatik, Nr. 15. `http://www.welzl.at/research/publications/optionprocessing_techrep1.pdf`. Version: Oktober 2003 (zitiert auf S. 80)

[RW04] ROSSI, Mattia ; WELZL, Michael: *On the Impact of IP Option Processing – Part 2*. Leopold-Franzens-Universität Innsbruck, Preprint-Reihe des Fachbereichs Mathematik & Informatik, Nr. 26. `http://www.welzl.at/research/publications/optionprocessing_techrep2.pdf`. Version: Juli 2004 (zitiert auf S. 80)

[San07] SANTOS, Omar: *End-to-End Network Security: Defense-in-Depth*. Cisco Press, 2007. – ISBN 1–58705–332–2 (zitiert auf S. 85)

[SBKV05] SETHURAMAN, Balasubramanian ; BHATTACHARYA, Prasun ; KHAN, Jawad ; VEMURI, Ranga: LiPaR: A Light-Weight Parallel Router for FPGA-based Networks-on-Chip. In: *Proceedings of the 15th ACM Great Lakes Symposium on VLSI (GLSVLSI)*. Chicago, Illinois, USA, April 2005. – ISBN 1–59593–057–4, S. 452–457 (zitiert auf S. 121 und 122)

[SCKT06] SILL, Frank ; CORNELIUS, Claas ; KUBISCH, Stephan ; TIMMERMANN, Dirk: Mixed Gates: Leakage Reduction techniques applied to Switches for Networks-on-Chip. In: *Proceedings of the Reconfigurable Communication-centric Systems-on-Chip Workshop 2006 (ReCoSoC'06)*. Montpellier, Frankreich, Juli 2006. – ISBN 2–9517461–2–1, S. 76–82 (zitiert auf S. 96)

[SGS05] SAUER, Christian ; GRIES, Matthias ; SONNTAG, Sören: Modular Reference Implementation of an IP-DSLAM. In: *Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC'05)*. La Manga del Mar Menor, Cartagena, Spanien, Juni 2005. – ISBN 0–7695–2373–0, S. 191–198 (zitiert auf S. 43)

[Sie06] SIEMENS AG COMMUNICATIONS – FIXED NETWORK ACCESS: *SURPASS Carrier Ethernet – Best Practices in designing a Carrier Grade Metro Ethernet Network for Triple Play Services*. White Paper, 2006. – `http://optical.usa.siemens.com/carrierethernet/downloads/Siemens_CE_&_IPTV_bestpractices_final_080806.pdf` (zitiert auf S. 26)

[Sil07] SILL, Frank: *Untersuchung und Reduzierung des Leckstroms integrierter Schaltungen in Nanometer-Technologien bei konstanten Performanceanforderungen*. Institut für Angewandte Mikroelektronik und Datentechnik, Universität Rostock, Dissertation, Dezember 2007 (zitiert auf S. 145)

[SJ06] SINNREICH, Henry ; JOHNSTON, Alan B.: *Internet Communications Using SIP: Delivering VoIP and Multimedia Services with Session Initiation Protocol*. 2. Auflage. John Wiley & Sons, Ltd., 2006. – ISBN 978–0–471–77657–4 (zitiert auf S. 22 und 24)

[SKY] *Skype™ – Allgemeine Geschäftsbedingungen*. `http://www.skype.com/intl/de/legal/terms/voip` (zitiert auf S. 82)

[SLKH02] SALMINEN, Erno ; LAHTINEN, Vesa ; KUUSILINNA, Kimmo ; HÄMÄLÄINEN, Timo D.: Overview of bus-based system-on-chip interconnections. In: *In Proceedings of the IEEE International Symposium on Circuits and Systems 2002 (ISCAS'02)*. Scottsdale, Arizona, USA, Mai 2002. – ISBN 0–7803–7448–7, S. 372–375, Band II (zitiert auf S. 94)

[Sof07] SOFKE, Sören: *Implementierung und Vergleich verschiedener Router-Konzepte für on-chip Netzwerke*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Projektarbeit, November 2007 (zitiert auf S. 99, 124 und 137)

[SPA] *The Apache SpamAssassin Project*. `http://spamassassin.apache.org` (zitiert auf S. 86)

[SPA+08] SHELBURNE, Matthew ; PATTERSON, Cameron ; ATHANAS, Peter ; JONES, Mark ; MARTIN, Brian ; FONG, Ryan: MetaWire: Using FPGA Configuration Circuitry to Emulate a Network-on-Chip. In: *Proceedings of the International Conference on Field Programmable Logic and Applications (FPL'08)*. Heidelberg, Deutschland, September 2008. – ISBN 978–1–4244–1961–6, S. 257–262 (zitiert auf S. 182)

[SPH05] SINHA, Rishi ; PAPADOPOULOS, Christos ; HEIDEMANN, John: Fingerprinting Internet Paths using Packet Pair Dispersion / University of Southern California Computer Science Department. Version: Februar 2005. `http://www.isi.edu/~johnh/PAPERS/Sinha05a.html`. 2005 (06-876). – Forschungsbericht (zitiert auf S. 60, 62 und 164)

[Spi05] SPINDLER, Richard: *PathMTU Discovery mit IP-Optionen*. Leopold-Franzens-Universität Innsbruck, Institut für Informatik, Bakkalaureatsarbeit, Juli 2005 (zitiert auf S. 80)

[Spi06] SPIES, Stephan: *Entwicklung einer Systemspezifikation für ein Hardwaremodul zur Schicht-2-Adressumsetzung in zukünftigen Zugangsnetzwerken*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Kleiner Beleg, Oktober 2006 (zitiert auf S. 52 und 54)

[Spu00] SPURGEON, Charles E.: *Ethernet: The Definitive Guide*. O'Reilly & Associates, Inc., 2000. – ISBN 1–56592–660–9 (zitiert auf S. 14)

[SRC84] SALTZER, Jerome H. ; REED, David P. ; CLARK, David D.: End-To-End Arguments in System Design. In: *ACM Transactions on Computer Systems* 2 (1984), November, Nr. 4, S. 277–288. – ISSN 0734–2071. – `http://www.reed.com/dpr/` (zitiert auf S. 104)

[SS05] SIVALINGAM, Krishna M. (Hrsg.) ; SUBRAMANIAM, Suresh (Hrsg.): *Emerging Optical Network Technologies: Architectures, Protocols and Performance*. Springer, 2005. – ISBN 0–387–22584–6 (zitiert auf S. 25 und 41)

[SS06] SMITH, David ; SKELLEY, Chelsea A.: *Globalization Transformation*. Technology Futures, Inc. `http://www.tfi.com/pubs/white.html`. Version: September 2006. – White Paper (zitiert auf S. 3 und 22)

[SSM+01] SGROI, Marco ; SHEETS, Mike ; MIHAL, Andrew ; KEUTZER, Kurt ; MALIK, Sharad ; RABAEY, Jan ; SANGIOVANNI-VINCENTELLI, Alberto: Addressing the System-on-a-Chip Interconnect Woes Through Communication-Based Design. In: *Proceedings of the 38th Conference on Design Automation (DAC'01)*. Las Vegas, Nevada, USA, Juni 2001. – ISBN 1–58113–297–2, S. 667–672 (zitiert auf S. 96)

[Sta08] STATISTISCHES BUNDESAMT: *Preise – Preise und Preisindizes für Nachrichtenübermittlung, Dezember 2007*. Fachserie 17, Reihe 9.1. `http://www.destatis.de`. Version: Januar 2008 (zitiert auf S. 22)

[Ste03] STEIN, Mike: Crossing the Abyss: Asynchronous Signals in a Synchronous World. In: *EDN* (2003), July, S. 59–69. – ISSN 0012–7515. – `http://www.edn.com` (zitiert auf S. 101 und 113)

[Str03] STREITFELD, David: *Opening Pandora's In-Box*. Mai 2003. – `http://www.latimes.com/technology/la-fi-spam11may11001420,1,5168218,full.story?ctrack=1&cset=true` (zitiert auf S. 84)

[Str08] STRZELETZ, Andy: *Konzipierung und Umsetzung einer Evaluierungsplattform zur Entwicklung von Algorithmen für PON-Systeme*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Masterarbeit, Februar 2008 (zitiert auf S. 33)

[SWT06] SALOMON, Ralf ; WIDIGER, Harald ; TOCKHORN, Andreas: Rapid Evolution of Time-efficient Packet Classifiers. In: *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI'06)*. Vancouver, British Columbia, Kanada, Juli 2006. – ISBN 0–7803–9487–9, S. 2793–2799 (zitiert auf S. 33)

[Sym07a] SYMANTEC MESSAGING AND WEB SECURITY: *The State of Spam – A monthly Report*, Januar–Dezember 2007. `http://www.symantec.com/business/theme.jsp?themeid=state_of_spam` (zitiert auf S. 84)

[Sym07b] SYMANTEC SECURITY RESPONSE & BUSINESS INTELLIGENCE: *Symantec Internet Security Threat Report – Trends für January-June 07*. Volume XII, September 2007. – `http://www.symantec.com/enterprise/theme.jsp?themeid=threatreport` (zitiert auf S. 28)

[Tan03] TANENBAUM, Andrew S.: *Computer Networks*. 4. Auflage. Prentice Hall PTR, 2003. – ISBN 0–13–046002–8 (zitiert auf S. 10)

[TGL07] TEEHAN, Paul ; GREENSTREET, Mark ; LEMIEUX, Guy: A Survey and Taxonomy of GALS Design Styles. In: *IEEE Design & Test of Computers* 24 (2007), Nr. 05, S. 418–428. – ISSN 0740–7475 (zitiert auf S. 100)

[The06] THE HONEYNET PROJECT: *Know Your Enemy: Honeynets*. White Paper. `http://www.honeynet.org`. Version: Mai 2006 (zitiert auf S. 86)

[Til08] TILERA CORPORATION: *TILE64™Processor – Product Brief*. 2008. – http://www.tilera.com (zitiert auf S. 96)

[Tur08] TURLEY, Jim: Software sells Processors. In: *Embedded Technology Journal* XII (2008), September, Nr. 11. http://www.embeddedtechjournal.com/articles_2008/pdf/20080909_software.pdf (zitiert auf S. 183)

[Tut06] TUTSCH, Dietmar: *Performance Analysis of Network Architectures*. Springer, 2006. – ISBN 3–540–34308–3 (zitiert auf S. 143)

[TW07] TAYLOR, Steven ; WEXLER, Joanie: 2007 Metro Ethernet State-of-the-Market Report / Nortel. Kubernan, Mai 2007. – Forschungsbericht. – http://www.nortel.com (zitiert auf S. 41)

[Uni07] UNITED STATES GOVERNMENT ACCOUNTABILITY OFFICE (GAO): *CYBERCRIME – Public and Private Entities Face Challenges in Addressing Cyber Threats*. Report to Congressional Requesters, Nr. GAO-07-705, Juni 2007. – http://www.gao.gov (zitiert auf S. 87)

[Van02] VANSTON, Lawrence K.: *Residential Broadband Forecasts*. Technology Futures, Inc. http://www.tfi.com/pubs/w/pdf/ti_broadband.pdf. Version: Oktober 2002. – White Paper (zitiert auf S. 22 und 24)

[Var05] VARGHESE, George: *Network Algorithmics: An Interdisciplinary Approach to Designing Fast Networked Devices*. Morgan Kaufmann Publishers Inc., 2005. – ISBN 0–12–088477–1 (zitiert auf S. 13 und 32)

[VHR+07] VANGAL, Sriram ; HOWARD, Jason ; RUHL, Gregory ; DIGHE, Saurabh ; WILSON, Howard ; TSCHANZ, James ; FINAN, David ; IYER, Priya ; SINGH, Arvind ; JACOB, Tiju ; JAIN, Shailendra ; VENKATARAMAN, Sriram ; HOSKOTE, Yatin ; BORKAR, Nitin: An 80-Tile 1.28 TFLOPS Network-on-Chip in 65 nm CMOS. In: *Proceedings of the 2007 International Solid-State Circuits Conference (ISSCC'07)*. San Francisco, Kalifornien, USA, Februar 2007. – ISBN 1–4244–0853–9, S. 98–100 (zitiert auf S. 96, 103 und 182)

[VLLX02] VRIENDT, Johan D. ; LAINÉ, Philippe ; LEROUGE, Christophe ; XU, Xiaofeng: Mobile Network Evolution: A Revolution on the Move. In: *IEEE Communications Magazine* 40 (2002), April, Nr. 4, S. 104–111. – ISSN 0163–6804 (zitiert auf S. 25)

[VM04] VARATKAR, Girish V. ; MARCULESCU, Radu: On-Chip Traffic Modeling and Synthesis for MPEG-2 Video Applications. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 12 (2004), Januar, Nr. 1, S. 108–119. – ISSN 1063–8210 (zitiert auf S. 139)

[VP07] VYNCKE, Eric ; PAGGEN, Christopher: *LAN Switch Security: What Hackers Know About Your Switches*. Cisco Press, 2007. – ISBN 1–58705–256–3 (zitiert auf S. 46)

[VSD01] VAIDYA, Aniruddha S. ; SIVASUBRAMANIAM, Anand ; DAS, Chita R.: Impact of Virtual Channels and Adaptive Routing on Application Performance. In: *IEEE Transactions on Parallel and Distributed Systems* 12 (2001), Februar, Nr. 2, S. 223–237. – ISSN 1045–9219 (zitiert auf S. 112)

[Wak00] WAKERLY, John F.: *Digital Design – Principles & Practices*. Dritte Auflage. Prentice Hall, 2000. – ISBN 0–13–769191–2 (zitiert auf S. 101)

[Was08] WASCHKI, Ansgar: *Untersuchung eines Routing-Algorithmus für Networks-on-Chip auf Deadlock-Freiheit*. Universität Rostock, Fakultät für Informatik und Elektrotechnik, Institut für Angewandte Mikroelektronik und Datentechnik, Bachelorarbeit, Oktober 2008 (zitiert auf S. 129)

[WCL05] WU, Chia-Ming ; CHI, Hsin-Chou ; LEE, Ming-Chao: Mapping of IP Cores to Network-on-Chip Architectures Based on Communication Task Graphs. In: *Proceedings of the 6th International Conference On ASIC (ASICON'05)*. Hualien, Taiwan, Oktober 2005. – ISBN 0–7803–9210–8, S. 953–956 Vol.2 (zitiert auf S. 148)

[WCL07] WANG, Pi-Chung ; CHAN, Chia-Tai ; LIN, Po-Yen: MAC Address Translation for Enabling Scalable Virtual Private LAN Services. In: *Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)*. Niagara Falls, Kanada, Mai 2007. – ISBN 0–7695–2847–3, S. 870–875 (zitiert auf S. 48)

[WH05] WESTE, Neil H. E. ; HARRIS, David: *CMOS VLSI Design*. Dritte Auflage. Addison Wesley, 2005. – ISBN 0–321–26977–2 (zitiert auf S. 145)

[Wik05] WIKLUND, Daniel: *Development and Performance Evaluation of Networks on Chip*, Linköping University, Department of Electrical Engineering, Diss., 2005 (zitiert auf S. 103)

[Wil04] WILSON, Ron: *Cisco taps processor array architecture for NPU*. EETimes, September 2004. – `http://www.eetimes.com/showArticle.jhtml?articleID=26806315` (zitiert auf S. 96)

[Wim07] WIMMREUTER, Wilhelm: *ENUM Interconnect of VoIP Islands – Is there a Life after Phone Call Charges?* Vortrag auf dem 9. ENUM-Tag der DENIC eG, September 2007. – `http://www.denic.de/media/pdf/enum/veranstaltungen/Wimmreuter_20070903.pdf` (zitiert auf S. 69)

[WKD+06] WIDIGER, Harald ; KUBISCH, Stephan ; DUCHOW, Daniel ; TIMMERMANN, Dirk ; BAHLS, Thomas: A Simplified, Cost-Effective MPLS Labeling Architecture for Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006 (zitiert auf S. 52 und 59)

[WKD+08] WIDIGER, Harald ; KUBISCH, Stephan ; DANIELIS, Peter ; SCHULZ, Jens ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk: IPclip: An Architecture to restore Trust-by-Wire in Packet-switched Networks. In: *Proceedings of the 33rd IEEE Conference on Local Computer Networks (LCN'08)*. Montreal, Quebec, Kanada, Oktober 2008. – ISBN 978–1–4244–2413–9, S. 312–319 (zitiert auf S. 81, 177 und 227)

[WKT07] WIDIGER, Harald ; KUBISCH, Stephan ; TIMMERMANN, Dirk: A Structural Architecture for HW Packet Processing. In: *Proceedings of the IEEE 11th Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM'07)*. Victoria, British Columbia, Kanada, August 2007. – ISBN 1–4244–1190–4, S. 363–366 (zitiert auf S. 58)

[WKTB06] WIDIGER, Harald ; KUBISCH, Stephan ; TIMMERMANN, Dirk ; BAHLS, Thomas: An Integrated Hardware Solution for MAT, MPLS-UNI, and TM in Access Networks. In: *Proceedings of the 31st Annual IEEE Conference on Local Computer Networks (LCN'06)*. Tampa, Florida, USA, November 2006. – ISBN 1–4244–0419–3, S. 272–279 (zitiert auf S. 45, 52, 57, 58 und 177)

[WL03] WIKLUND, Daniel ; LIU, Dake: SoCBUS: Switched Network on Chip for Hard Real Time Embedded Systems. In: *CD-ROM/Abstracts Proceedings of the 17th International Parallel and Distributed Processing Symposium (IPDPS'03)*. Nizza, Frankreich, April 2003. – ISBN 0–7695–1926–1, S. 78 (zitiert auf S. 103)

[WPC] *WinPcap: The Windows Packet Capture Library*. – `http://www.winpcap.org` (zitiert auf S. 223 und 228)

[WST06] WIDIGER, Harald ; SALOMON, Ralf ; TIMMERMANN, Dirk: Packet Classification with Evolvable Hardware Hash Functions – An Intrinsic Approach. In: *Proceedings of the Second International Workshop on Biologically Inspired Approaches to Advanced Information Technology (Bio-ADIT'06)*. Osaka, Japan, Januar 2006. – ISBN 3–540–31253–6 (zitiert auf S. 33)

[WTST07] WIDIGER, Harald ; TOCKHORN, Andreas ; SALOMON, Ralf ; TIMMERMANN, Dirk: Accelerating the Evolution of Evolvable Hardware-based Packet Classifiers. In: *Proceedings of the IEEE SSCI Workshop on Evolvable and Adaptive Hardware (WEAH'07)*. Honolulu, Hawaii, USA, April 2007. – ISBN 1–4244–0699–4, S. 27–34 (zitiert auf S. 33)

[XCG] XILINX, INC.: *Constraint Guide 9.1i*. – `http://toolbox.xilinx.com/docsan/xilinx9/books/docs/cgd/cgd.pdf` (zitiert auf S. 163 und 171)

[XLX] XILINX, INC.: *Products & Services – Silicon Devices*. – `http://www.xilinx.com/products/silicon_solutions` (zitiert auf S. 112)

[XV5] XILINX, INC.: *Virtex-5 Family Overview*. – `http://www.xilinx.com/support/documentation/data_sheets/ds100.pdf` (zitiert auf S. 103)

[XWC] XILINX, INC.: *Wired Communications*. – `http://www.xilinx.com/esp/wired` (zitiert auf S. 57)

[YMB02] YE, Terry T. ; MICHELI, Giovanni D. ; BENINI, Luca: Analysis of power consumption on switch fabrics in network routers. In: *Proceedings of the 39th Conference on Design Automation (DAC'02)*. New Orleans, Louisiana, USA, Juni 2002. – ISBN 1–58113–461–4, S. 524–529 (zitiert auf S. 96)

[YYS+08] YAMAZAKI, Hiroshi ; YAMADA, Takashi ; SAKAMAKI, Yohei ; KANEKO, Akimasa ; SANO, Akihide ; MASUDA, Hiroji ; MIYAMOTO, Yutaka: Advanced Optical Modulators with Hybrid Configuration of Silica-Based PLC and $LiNbO_3$ Phase-Shifter Array for Ultra-High-Speed Transport Networks. In: *Proceedings of the 1st ITU-T Kaleidoscope Conference: Innovations in Next Generation Networks - Future Network and Services (K-INGN'08)*. Genf, Schweiz, Mai 2008. – ISBN 92–61–12441–0, S. 237–244 (zitiert auf S. 106)

[Zim80] ZIMMERMANN, Hubert: OSI Reference Model—The ISO Model of Architecture for Open System Interconnection. In: *IEEE Transactions on Communications* COM-28 (1980), April, Nr. 4, S. 425–432. – ISSN 0090–6778 (zitiert auf S. 8)

[ZKS04] ZEFERINO, Cesar A. ; KREUTZ, Márcio E. ; SUSIN, Altamiro A.: RASoC: A Router Soft-Core for Networks-on-Chip. In: *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition Designers' Forum (DATE'04)*. Paris, Frankreich, Februar 2004. – ISBN 0–7695–2085–5, S. 198–203 Vol.3 (zitiert auf S. 121 und 122)

Teil V.

Anhänge

Man kann ein Problem nicht mit den gleichen Denkstrukturen lösen, die zu seiner Entstehung beigetragen haben.
(Albert Einstein, dt.-schweiz.-amerik. Physiker)

Anhang A.

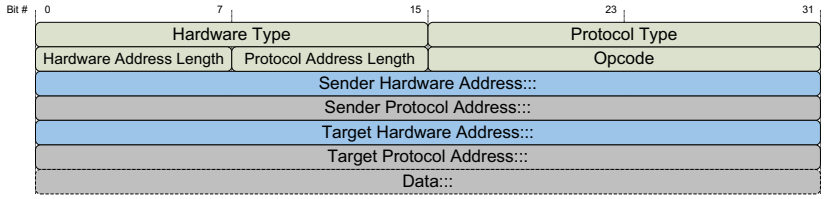
MAC Address Translation

A.1. Sonderbehandlungen verschiedener Protokolle

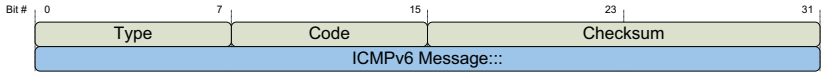
Abschnitt 4.2.1 berichtete über die Sonderbehandlung bestimmter Protokolle durch die MAT-Funktionalität. Im Folgenden sind Details zu (R)ARP [RFC0826, RFC0903] und DHCP [RFC2131] gegeben. Details zu ICMPv6 [RFC2463] und DHCPv6 [RFC3315], welche die gleichen Aufgaben in IPv6-Umgebungen realisieren, werden an dieser Stelle nicht illustriert, da die grundlegende Vorgehensweise ähnlich und an (R)ARP und DHCP angelehnt ist.

Sonderbehandlung von ARP, RARP und ICMPv6 Abbildung A.1a zeigt den Aufbau eines (R)ARP-Paketes. Beide Protokolle haben die gleiche Struktur und unterscheiden sich allein im Befehlscode und dem unbekannten Teil der Adressinformationen. ARP erfragt die zu einer bekannten IP-Adresse gehörende MAC-Adresse (Abbildungen A.2 und A.3). RARP erfragt die an eine bekannte MAC-Adresse gebundene IP-Adresse (Abbildungen A.4 und A.5). Für MAT entscheidend sind die im (R)ARP-Paket enthaltenen Informationen zur Sender und Target Hardware Address, welche in Abbildung A.1a blau hervorgehoben sind. (R)ARP-Anfragen werden zwar zumeist bereits durch den DSLAM beantwortet, können aber auch den BRAS erreichen. Diese Fälle müssen durch die MAT-Funktionalität behandelt werden. Die MAC-Adresse des BRAS (MAC_BRAS) wird nie getauscht, da der BRAS zur Netzwerkinfrastruktur gehört. Sie sollte durch ein gesetztes `W`-Flag (White List) gekennzeichnet sein. (R)ARP-Pakete können anhand der Werte `0x0806` bzw. `0x8035` im Typ-Feld des Ethernet-Headers erkannt werden. Opcode (`1` = ARP Request, `2` = ARP Reply, `3` = RARP Request, `4` = RARP Reply) und Richtung des Datenflusses (Up-/Downstream) sind Kriterien für den Tausch der MAC-Adressen.

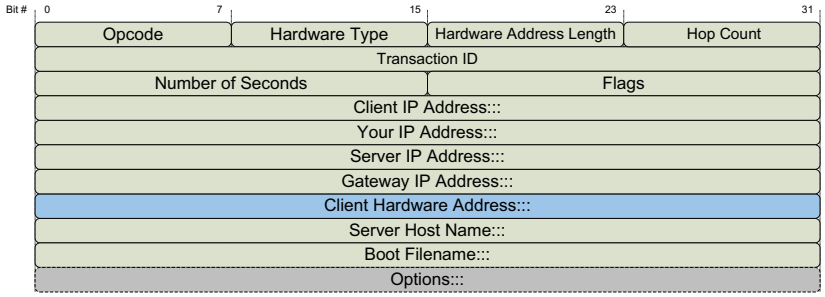
Abbildung A.2 zeigt ein vom Teilnehmer ausgehendes ARP-Request im Upstream. Das MAT-



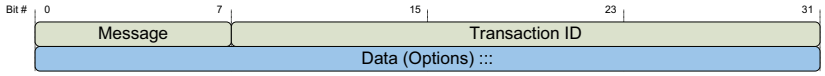
(a) (R)ARP-Header und -Nutzdaten



(b) ICMPv6-Header und -Nutzdaten



(c) DHCP-Header und -Nutzdaten



(d) DHCPv6-Header und -Nutzdaten

Abbildung A.1.: Struktur und Aufbau von (R)ARP-, DHCP(v6)- und ICMPv6-Nachrichten

Modul auf dem Zugangsknoten (DSLAM) tauscht die CMAC innerhalb des Ethernet-Headers *und* der ARP-Nutzdaten. Gleiches gilt für das ARP-Reply im Downstream. Abbildung A.3 zeigt ein vom BRAS ausgehendes ARP-Request im Downstream. Das MAT-Modul führt keinen Tausch der Adressen durch. Im ARP-Reply im Upstream wird jedoch die CMAC innerhalb des Ethernet-Headers *und* der ARP-Nutzdaten getauscht.

Abbildung A.4 zeigt ein vom Teilnehmer kommendes RARP-Request im Upstream. Die CMAC innerhalb des Ethernet-Headers *und* der RARP-Nutzdaten wird gegen eine PMAC getauscht. Gleiches gilt für das RARP-Reply im Downstream. Abbildung A.5 zeigt ein vom BRAS ausgehendes RARP-Request im Downstream und das RARP-Reply im Upstream. Sowohl beim Request als auch beim Reply wird ein Tausch der PMAC bzw. CMAC innerhalb des Ethernet-Headers *und* der RARP-Nutzdaten durchgeführt.

In IPv6-Umgebungen kommt der Neighbor Discovery-Mechanismus von ICMPv6 anstelle von

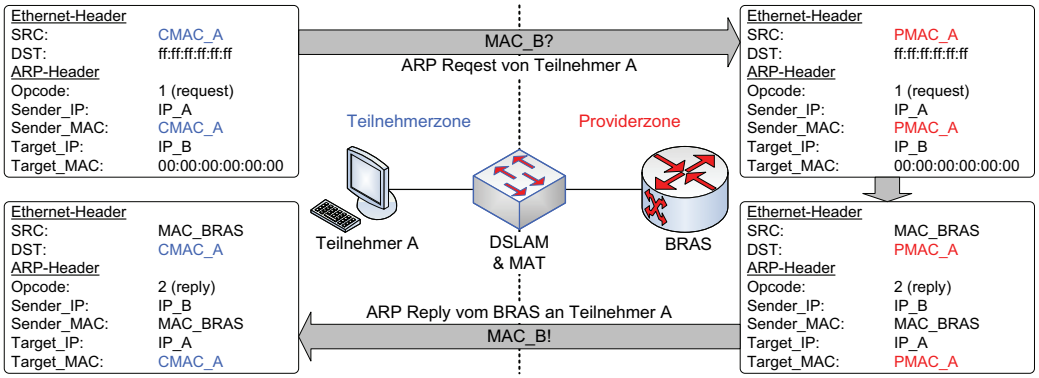


Abbildung A.2.: MAT bei einem ARP-Request vom Teilnehmer im Upstream

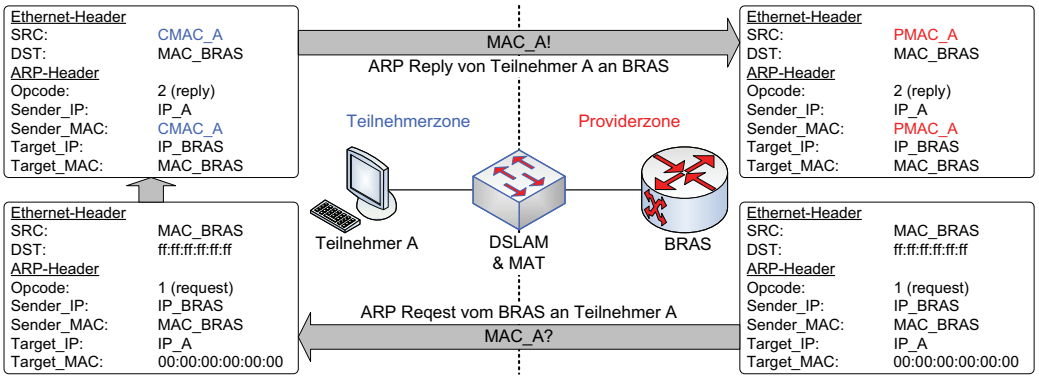


Abbildung A.3.: MAT bei einem ARP-Request vom BRAS im Downstream

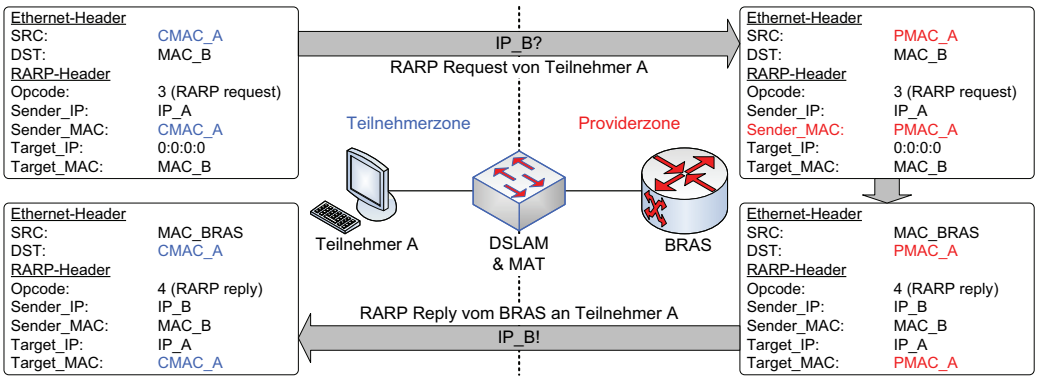


Abbildung A.4.: MAT bei einem RARP-Request vom Teilnehmer im Upstream

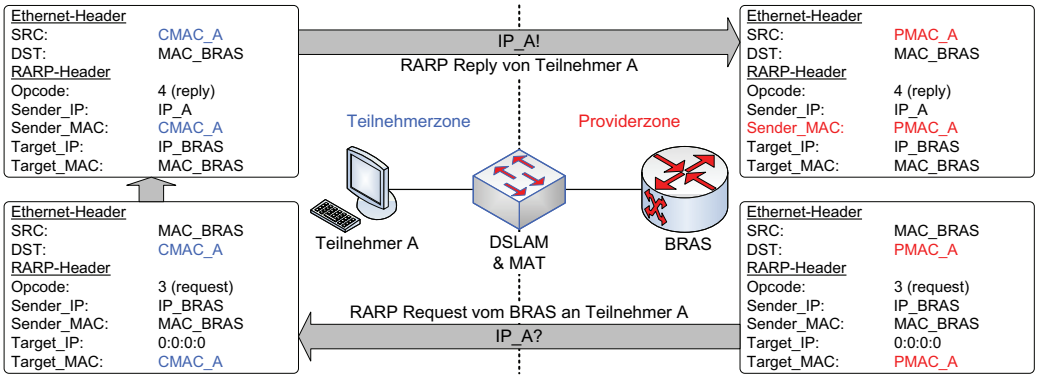


Abbildung A.5.: MAT bei einem RARP-Request vom BRAS im Downstream

(R)ARP zum Einsatz, welches sich jedoch an die Mechanismen von (R)ARP anlehnt [RFC2461, RFC4862]. Darüber hinaus wird Neighbor Discovery auch zur Autokonfiguration von IPv6-Adressen genutzt. Abbildung A.1b zeigt die Aufbau des ICMPv6-Headers. Request und Reply werden mit (Inverse) Neighbor Solicitation und Advertisement bezeichnet. Da ICMPv6 ein Protokoll der Transportschicht ist, muss zunächst das Next Header-Feld des IPv6-Headers auf den Wert `58` geprüft werden, welcher ICMPv6 signalisiert (siehe auch Abbildung 2.7b). ICMPv6-Nachrichten des Typs Solicitation und Advertisement sind durch ein Type-Feld mit den Werten `135` & `136` (entspricht ARP) sowie `141` & `142` (entspricht RARP) gekennzeichnet. Werden MAC-Adressen innerhalb der ICMPv6-Nachrichten durch MAT getauscht, muss zusätzlich die Prüfsumme des ICMPv6-Headers aktualisiert werden.

Sonderbehandlung von DHCP(v6) DHCP ist ein Protokoll auf UDP-Basis und regelt die Adressvergabe in IPv4-Umgebungen. Abbildung A.1c zeigt den Aufbau eines DHCP-Headers. DHCP ist durch ein Ethernet-Typ-Feld mit dem Wert `0x0800` (IP), durch ein IP-Protokoll-Feld mit dem Wert `17` (UDP) und durch die UDP-Ports `67` (Server) bzw. `68` (Client) gekennzeichnet. Für MAT entscheidend ist die im DHCP-Header enthaltene Client Hardware Address (chaddr). Zwei verschiedene Fälle müssen unterschieden werden. Einerseits kann eine direkte Kommunikation zwischen Teilnehmern und DHCP-Server (zumeist auf dem BRAS) erfolgen (Abbildung A.6). Andererseits kann die Kommunikation über einen zwischengeschalteten DHCP-Agenten (Relay) auf dem Zugangsknoten erfolgen, der die DHCP-Pakete an den korrekten DHCP-Server weiterleitet (Abbildung A.7). Die MAC-Adressen von Server und Relay (MAC_Server, MAC_Relay) werden nie getauscht, da sie zur Netzwerkinfrastruktur gehören und durch gesetzte `W`-Flags (White List) gekennzeichnet sein müssen.

Abbildung A.6 zeigt das Vorgehen bei direkter Kommunikation mit einem DHCP-Server.

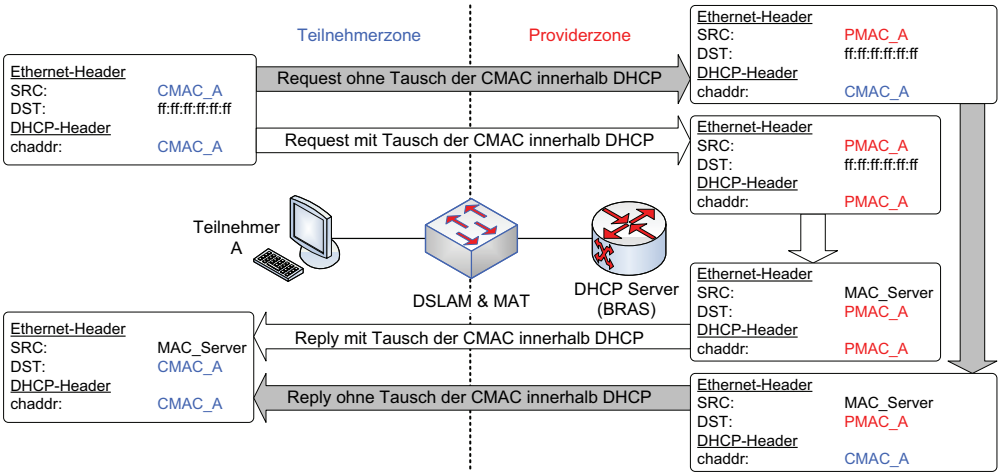


Abbildung A.6.: MAT bzgl. DHCP bei direkter Kommunikation mit dem DHCP-Server

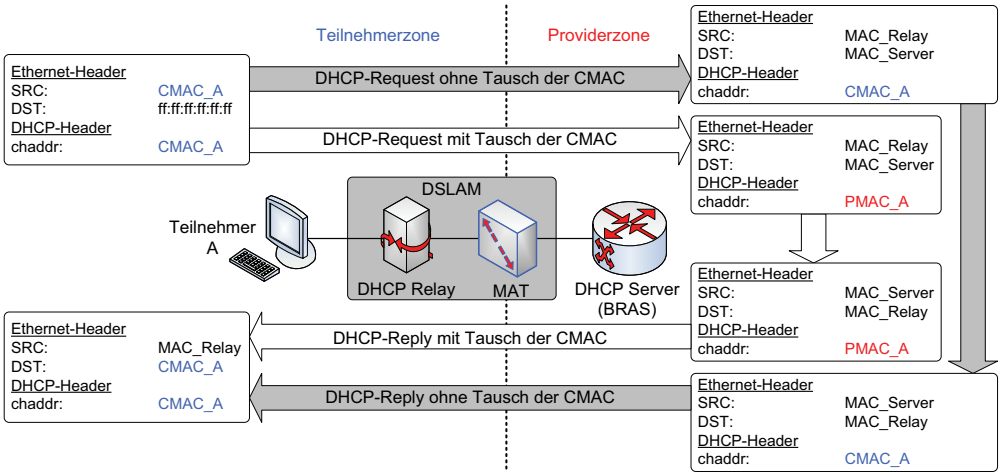


Abbildung A.7.: MAT bzgl. DHCP mit zwischengeschaltetem DHCP-Relay

Der DHCP-Server vergibt IP-Adressen auf Basis der chaddr im DHCP-Header. Für eine n:1-Umsetzung zwischen CMACs und PMACs ist die CMAC im DHCP-Header *nicht* zu tauschen, damit jeder CMAC eine individuelle IP-Adresse zugewiesen werden kann. Andernfalls würden ständig DHCP-Requests für die gleiche PMAC erfolgen. Dieses Vorgehen zeigt der graue Pfad in Abbildung A.6. Bei einer 1:1-Umsetzung ist es rein logisch betrachtet unerheblich, ob die CMAC innerhalb des DHCP-Headers getauscht wird oder nicht. Aufgrund der Konsistenz sollten aber alle CMACs getauscht werden, damit die Adressbindungen konsistent sind. Dieses Vorgehen zeigt der weiße Pfad in Abbildung A.6. In beiden Fällen ist jedoch die CMAC im Ethernet-Header

durch die PMAC auszutauschen.

Abbildung A.7 zeigt das Vorgehen bei einem zwischengeschalteten DHCP-Relay auf dem Zugangsknoten. Die Überlegungen sind prinzipiell dieselben wie bei direkter Kommunikation mit dem DHCP-Server. Der Unterschied besteht darin, dass der Relay-Agent seine eigene MAC (MAC_Relay) in den Ethernet-Header einsetzt und an den DHCP-Server weiterleitet. D. h., MAC-Adressen im Ethernet-Header werden in diesem Szenario *nicht* getauscht.

In IPv6-Umgebungen wird DHCPv6 genutzt, welches im UDP-Header durch die Ports `546` (Client) und `547` (Server, Relay Agent) gekennzeichnet ist. Abbildung A.1d zeigt die Struktur eines DHCPv6-Headers. Abhängig vom Message-Feld sind verschiedene Optionen an den DHCP-Header angehängt. Einige Optionen, z. B. vom Typ `OPTION_INTERFACE_ID`, enthalten den IPv6 Interface Identifier oder den DUID, welche jeweils aus der CMAC gebildet sein können. Die Ersetzung dieser Adressinformationen ist ähnlich wie im Fall von IPv4 abhängig von der Existenz eines DHCPv6 Relay Agenten und der genutzten Ersetzungsstrategie von MAT.

A.2. Schutz vor MAC- und ARP-Spoofing

ARP-Spoofing ARP Spoofing nutzt die Schwäche von standardkonformen ARP-Implementierungen in Netzwerkgeräten aus. Diese Schwachstelle zeichnet sich durch die automatische Aktualisierung des lokalen ARP-Caches durch Antworten auf ARP-Anfragen aus. Jedes Gerät, das Daten versenden will, benötigt die MAC-Adresse des Zielgerätes, welche per ARP-Anfrage bezogen werden kann. Abbildung A.8 zeigt dazu ein vereinfachtes, ungeschütztes Netz vor einem ARP-Angriff. Aufgrund der Switching-Funktion des DSLAMs kann Eve keine Frames von und zu Alice mithören.

ARP-Anfragen werden per Broadcast periodisch durchgeführt. Der Angreifer (Eve) sieht somit alle ARP-Anfragen und antwortet mit seiner eigenen MAC-Adresse auf die Anfrage, wodurch in den ARP-Cache des anfragenden Opfers (Alice) gefälschte Einträge geschrieben werden. Abbildung A.9 stellt dies dar. Alle Frames von Alice zum BRAS werden nun zu Eve gesendet und können mitgehört werden, z. B. persönliche oder sensitive Daten in E-Mails. Um unerkannt zu bleiben, leitet Eve die Daten trotzdem in Form einer transparenten Zwischenstation zum BRAS weiter, was i. Allg. als Man-in-the-Middle-Angriff bekannt ist. Dazu ist es zusätzlich erforderlich, den ARP-Cache des BRAS' zu manipulieren, indem ein weiteres verfälschtes ARP-Reply gesendet wird. Dies zeigt Abbildung A.10. Alle Frames zwischen Alice und dem BRAS passieren nun Eve.

Im Gegensatz dazu zeigt Abbildung A.11 ein durch MAT abgesichertes Szenario. Von Eve generierte ARP-Pakete werden durch die MAT-Funktionalität gesondert behandelt und mit ge-

tauschten Adressinformationen weitergeleitet (siehe dazu Abschnitt A.1), wenn für den Schlüssel aus CMAC und IP ein konfigurierter Eintrag in den MAT-Adresstabellen vorliegt. Für die dezentrale sMAT-Variante kann darüber hinaus die Zugangsportnummer der Linecard als Kriterium herangezogen werden. Ist für einen Schlüssel keine Regel in der statischen MAT-Adresstabelle auffindbar, so kann je nach der vom ISP bevorzugten Vorgehensweise der Frame blockiert werden, der Fehler der administrativen Instanz signalisiert werden, oder nach mehrfachem Auftreten sogar das Zugangsport gesperrt werden. In jedem Fall sind in der Providerzone ausschließlich konfigurierte PMACs sichtbar und ARP-Spoofing wird unterbunden.

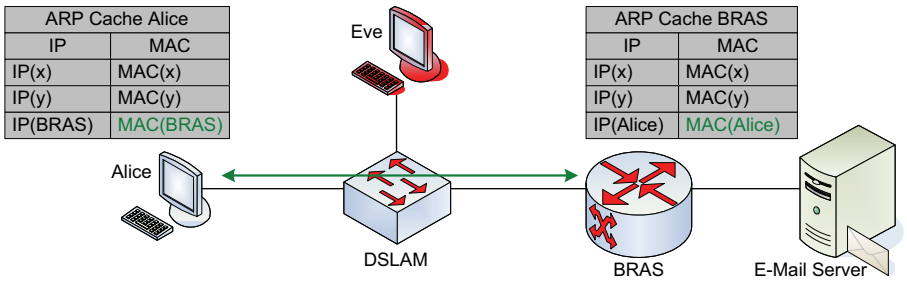


Abbildung A.8.: Vereinfachtes Netzwerkszenario vor einem ARP-Spoofing-Angriff

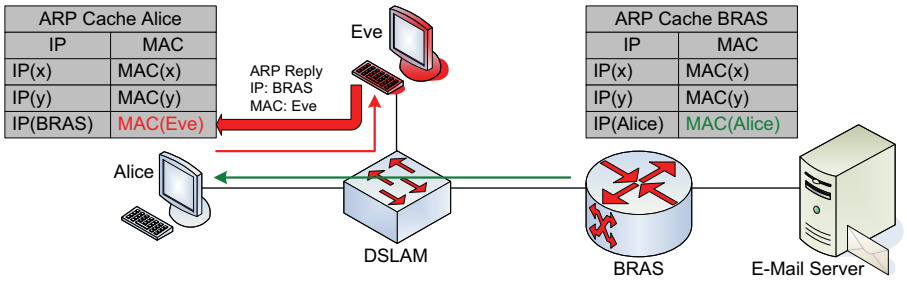


Abbildung A.9.: Netzwerk nach einem unidirektionalen ARP-Spoofing-Angriff

MAC-Spoofing & Flooding Die Mechanismen MAC-Spoofing und Flooding beruhen beide auf dem Versand von Ethernet-Frames mit gefälschter SRC MAC. Während beim Spoofing der Identitätsklau mit einer fremden, gültigen MAC-Adresse im Vordergrund steht, ist es beim Flooding das Überschwemmen mit einer Vielzahl an unterschiedlichen Adressen. Beim ARP-Spoofing ist der lokale ARP-Cache eines Gerätes im Netzwerk das Ziel der Manipulationsversuche. Beim MAC-Spoofing und Flooding ist die FDB der Switche innerhalb der Netzwerkinfrastruktur das Ziel, da neue Einträge auch dort durch eintreffende Frames automatisch gelernt werden.

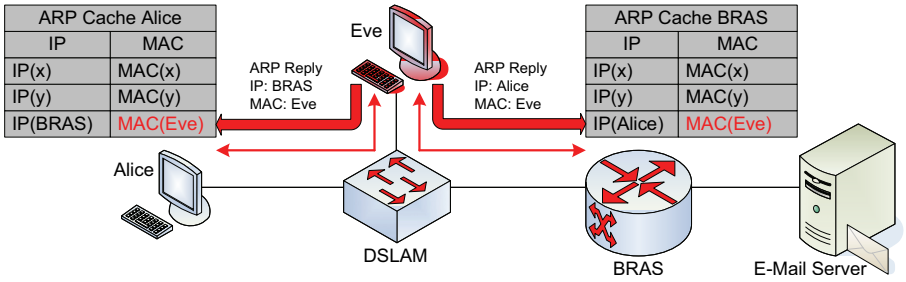


Abbildung A.10.: Netzwerk nach einem transparenten, bidirektionalen ARP-Spoofing-Angriff

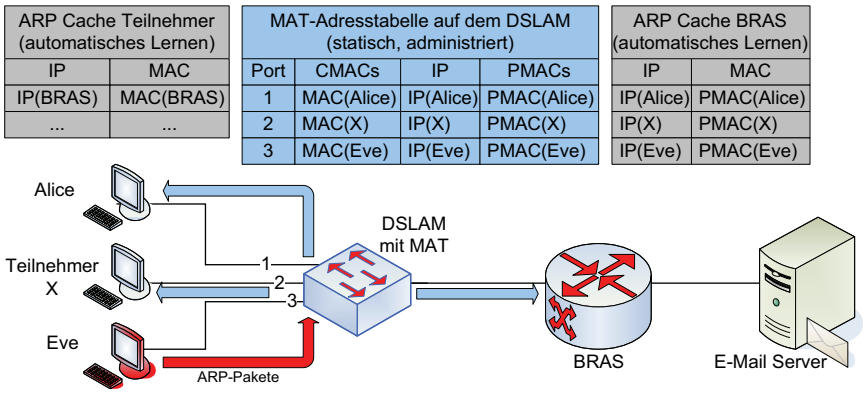
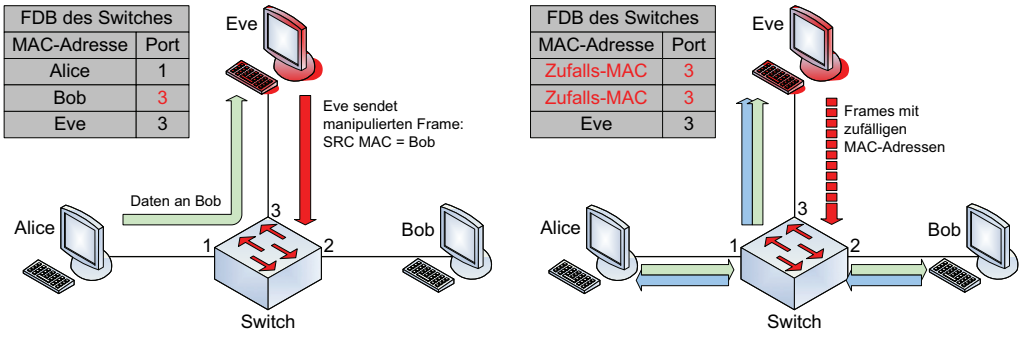


Abbildung A.11.: Netzwerk gesichert durch MAT auf dem DSLAM



(a) Erfolgreicher MAC-Spoofing-Angriff **(b)** Erfolgreicher MAC-Flooding-Angriff

Abbildung A.12.: Netzwerk nach erfolgreichem MAC-Spoofing und MAC-Flooding

Beim MAC-Spoofing wird eine bestimmte SRC-MAC genutzt, um den Datenfluss des Opfers in einem Switch umzuleiten. Abbildung A.12a zeigt ein einfaches Netzwerk nach erfolgreichem MAC-Spoofing. Eve hat die FDB des Switches manipuliert, indem sie einen Frame mit der gültigen MAC-Adresse von Bob verschickt. Daraufhin werden Frames, die Bob zum Ziel haben, fälschlicherweise an Eve umgeleitet. Abbildung A.12b zeigt dasselbe Netz nach einem erfolgreichen MAC-Flooding-Angriff. Durch das massenweise Versenden von Frames mit zufälligen, verschiedenen MAC-Adressen wurde die FDB des Switches überladen. Normalerweise leitet ein Switch Frames aufgrund der MAC-Einträge der FDB weiter. Ist kein Eintrag vorhanden, wird der Frame gebroadcasted (geflutet). Nach einem Tabellenüberlauf kann keine MAC mehr eindeutig einem Port zugewiesen werden, weshalb der Switch in den sogenannten Failopen-Modus wechselt und alle eintreffenden Frames an allen Ports geflutet. Dadurch kann Eve alle Daten zwischen Bob und Eve mithören.

MAT schützt vor diesen Angriffen in ähnlicher Art und Weise wie beim ARP-Spoofing im vorherigen Abschnitt, da es durch die statische Natur der MAT-Adresstabellen die dynamische Aktualisierung der FDB unterbindet. In der sMAT-Variante kann zusätzlich pro Teilnehmerport eine maximale zulässige Anzahl an MAC-Adressen definiert werden.

A.3. Konfigurationstool des MATMUNI Prototyps

Für den Prototyp des MATMUNI-Systems wurde ein grafisches Nutzerinterface für die Konfiguration des Systems [KWD+07] auf der Hardware-Entwicklungsplattform ML-405 von Xilinx [ML405] realisiert. Dazu wurden die Open Source-Version der QT-Entwicklungsumgebung von Trolltech [QT] sowie die freie WinPCap-Bibliothek [WPC] genutzt. Auf Basis eines einfachen Ethernet-basierten Protokolls werden Konfigurationsdaten an das MATMUNI-System gesendet, welche anhand einer festgelegten MAC-Adresse (MAC_{config}) erkennbar sind. Die in Abschnitt 4.3.1 beschriebene Architektur des MATMUNI-Prototyps zeigt u. a. eine CPU-Schnittstelle, welche das Interface für die Administration und Konfiguration des MATMUNI-Systems darstellt. Ein einfacher Filter überprüft eintreffende Frames im Upstream auf MAC_{config} und leitet diese zur CPU-Schnittstelle um, über welche dann die Konfiguration des Systems erfolgt. Am angedachten Einsatzort für MATMUNI im DSLAM (siehe Abbildung 4.11) ist diese Schnittstelle direkt mit einer administrativen Instanz verbunden, z. B. einer CPU.

Abbildung A.13 zeigt das MATMUNI-Konfigurationstool. Die Hauptfunktionen sind:

- Konfiguration der MATMUNI-Funktionalität auf dem Board zur Laufzeit
- Globale Parameter wie eigene MAC-Adresse, MTU, Angaben zum Schlüsselset

- Konfiguration individueller Regeln und Parameter pro Funktionsmodul (MAT, TM, MPLS) über eine separate Ansicht
- Laden/Speichern von kompletten Konfigurationen im XML-Format für eine einfache „Pushbutton“-Konfiguration des Gesamtsystems

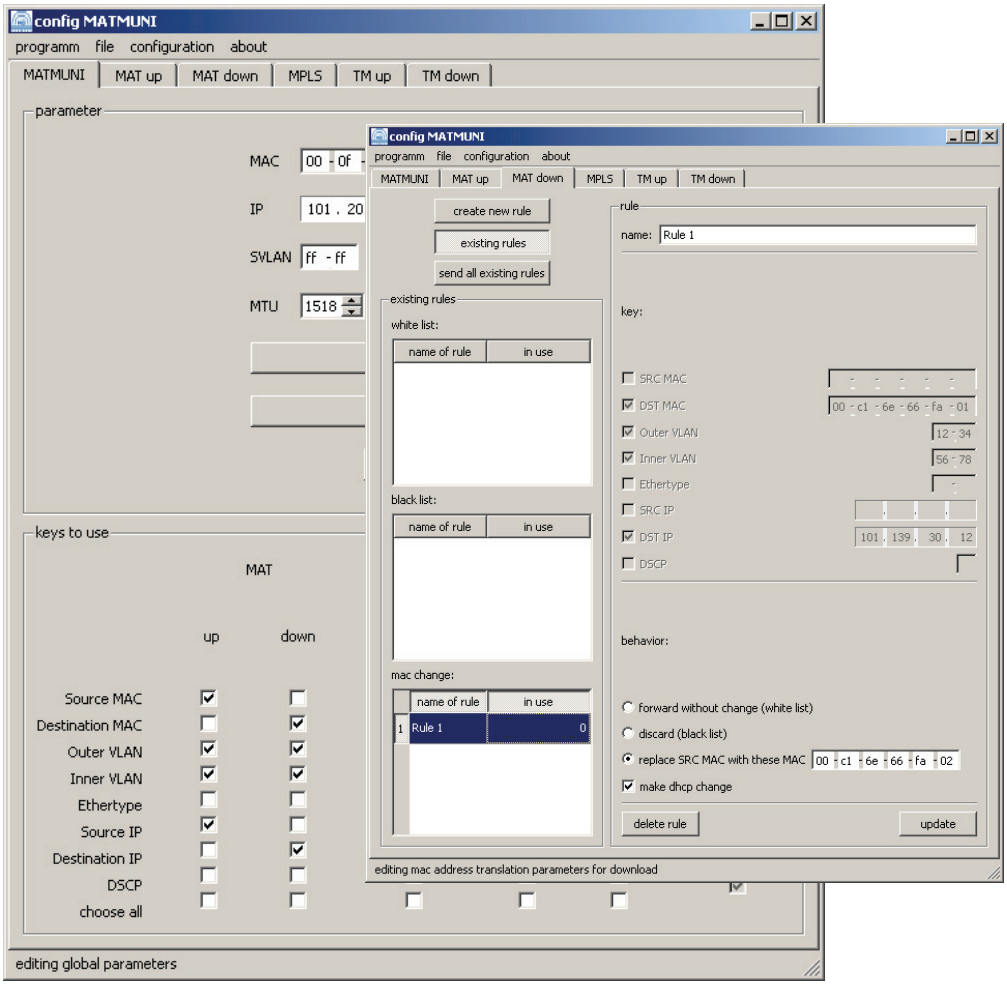


Abbildung A.13.: Konfigurationstool des MATMUNI-Systems, Beispielansicht für allgemeine Systemparameter (hinten) sowie der MAT-Funktionalität (vorne)

I am fully functional.
(LtCdr. Data)

Anhang B.

IPclip

B.1. Weitere Optionstypen

In Abschnitt 5.1.2 wurden die IPclip-Optionstypen `0x01` und `0x03` beschrieben. Tabelle 5.1 definiert jedoch noch weitere mögliche Typen.

RFC 3825 [RFC3825] definiert eine DHCP-Option für eine Koordinaten-basierte Ortsinformation. Diese sogenannte Geospatial Location Information (GLI) enthält sowohl Angaben zu Länge und Breite als auch zur aktuellen Höhe und sind als IPclip-Typen `0x02` und `0x04` definiert. Die Kodierung der einzelnen Parameter ist in Tabelle B.1 gezeigt.

Tabelle B.1.: Format der GLI-Information in einer IPclip-Option des Typs `0x02` bzw. `0x04`

Parameter		Wertebereich	# der Bits
Breite	Auflösung	0...34	6
	Grad (Ganzzahl)	(-90)...(+90)	9
	Minuten (Nachkommateil)	0...(1-2^{-26})	25
Länge	Auflösung	0...34	6
	Grad (Ganzzahl)	(-180)...(+180)	9
	Minuten (Nachkommateil)	0...(1-2^{-26})	25
Höhe	Höhentyp	0...2	4
	Auflösung	0...30	6
	Wert (Ganzzahl)	0...(2^{22}-1)	22
	Wert (Nachkommateil)	0...(1-2^{-9})	8
Datum		1...3	8

Die Werte für Auflösung geben jeweils die Anzahl der relevanten Bits für Länge, Breite und Höhe an. Der Höhentyp gibt an, ob die Höhe in Meter oder Etagen zu interpretieren ist.

Datum gibt den Koordinatentyp dieser Option an. Für gewöhnlich werden WGS84-Koordinaten [NIM00] genutzt und durch den Wert 1 repräsentiert. Die Struktur der gesamten IP-Option mit einer IPclip-Option vom Typ `0x04` ist in Abbildung B.1 gezeigt. Für reine GLI-Daten ergibt sich für die gesamte IP-Option eine Länge von 20 Byte (Typ `0x02`). Inklusive ID des Zugangsknotens und Portnummer der Linecard (Typ `0x04`) beträgt die Länge 24 Byte.

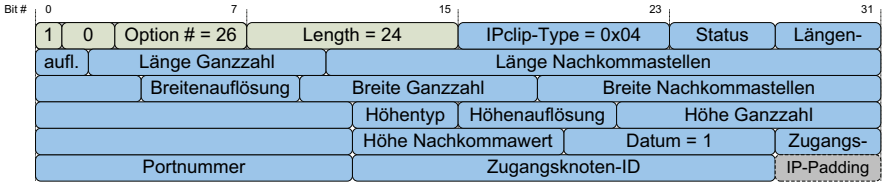


Abbildung B.1.: Format einer IPclip-IP-Option mit GLI-Informationen

GPS-Daten können neben Positionsangaben auch Zeitinformationen in Form der koordinierten Weltzeit (UTC) beinhalten. Diese Zeitinformationen werden in einer IPclip-Option des Typs `0x05` transportiert. Die relevanten Parameter sind Stunden, Minuten, Sekunden und Millisekunden. Die Kodierung dieser Informationen ist in Tabelle B.2 gezeigt. Die Struktur der entsprechenden IP-Option ist in Abbildung B.2 ersichtlich. Die Gesamtgröße der IP-Option beträgt 8 Byte.

Tabelle B.2.: Format der GPS-Zeitinformationen in einer IPclip-Option des Typs `0x05`

Parameter	Wertebereich	# der Bits
Stunden	$0 \ldots 23$	5
Minuten	$0 \ldots 59$	6
Sekunden	$0 \ldots 59$	6
Millisekunden	$0 \ldots (1 - 2^{-11})$	10

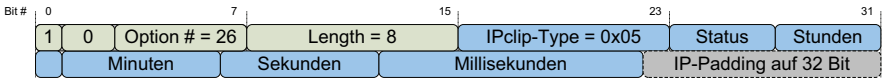


Abbildung B.2.: Format einer IPclip-IP-Option mit GPS-Zeitinformationen

Neben der primären Nutzung des IPclip-Mechanismus zur Anreicherung von IP-Paketen mit Ortsreferenzen kann IPclip auch als Container für arbiträre Informationen bzw. Metadaten weitergenutzt werden. In [DKW+08b] wird das IPclip-Gerüst genutzt, um z. B. den ursprünglichen Hop-Count des IP-Headers innerhalb einer IPclip-Option zu transportieren.

B.2. IPclip-Hardware-Prototyp

Der gesamte IPclip-Mechanismus wurde als voll funktionsfähiger Prototyp in Hardware realisiert und in [DKW+08a] und [DKW+09] vorgestellt. Abbildung B.3 zeigt die Architektur des Prototyps, welcher aus den folgenden funktionalen Submodulen zusammengesetzt ist. Weitere Details bietet [WKD+08].

Parser Der Parser extrahiert aus jedem eintreffenden Ethernet-Frame die SRC IP und das VLAN-Tag. In einer Adresstabelle wird die zugehörige Portinformation gesucht und an nachfolgende Module parallel zum eigentlichen Frame weitergereicht.

MTU Adaptation Module (MAM) Das MAM dient im Fall von IPoE der unter Abschnitt 5.1.5 beschriebenen automatischen Anpassung der PMTU während der PMTUD. MAM ist im Up- und Downstream aktiv.

PPPoE MTU Adaptation Module (PAM) Das PAM dient ebenfalls der automatischen Anpassung der PMTU. Jedoch greift PAM ausschließlich im Fall von PPPoE in die Signalisierung während der Aushandlung der PMTU ein.

Option Verification Module (OVM) Zur Verifikation der Plausibilität einer vom Nutzer eingefügten LI wird das OVM genutzt. Es führt eine Vergleich der eigenen geographischen Position mit der LI des Nutzers in Bezug auf die SCA durch. Das Ergebnis wird zusammen mit dem IP-Paket dem AIA übergeben.

Additional Information Adder (AIA) Der AIA wurde in der Projektarbeit von Krukowski [Kru06] angefertigt. Die Hauptaufgabe ist das Einfügen von Optionen in den IP-Header von IP-Paketen des Upstreams. Abhängig von der Entscheidung des OVM wird entweder eine existierende LI überschrieben, eine neue IP-Option eingefügt oder das IP-Paket verworfen. In jedem Fall werden die Status-Bits gesetzt und die Prüfsummen für den IP-Header aktualisiert.

Additional Information Remover (AIR) Das optionale AIR-Modul arbeitet ausschließlich im Downstream. Es ist für das Entfernen von IPclip-Optionen zuständig. Dazu wird das RF-Bit einer IPclip-Option ausgewertet, wenn das Entfernen nicht als Default-Aktion konfiguriert worden ist.

Sync-FIFO Diese FIFO-Speicher dienen der Taktsynchronisation und Pufferung von Frames.

Filter Die Übertragung von Konfigurationsdaten zum Prototyp erfolgt mittels eines einfachen, proprietären Protokolls auf Basis von Ethernet-Frames. Ein Filter vor MAM untersucht den Upstream auf Frames mit bestimmter MAC-Adresse (MAC_{config}), welche an ein zum

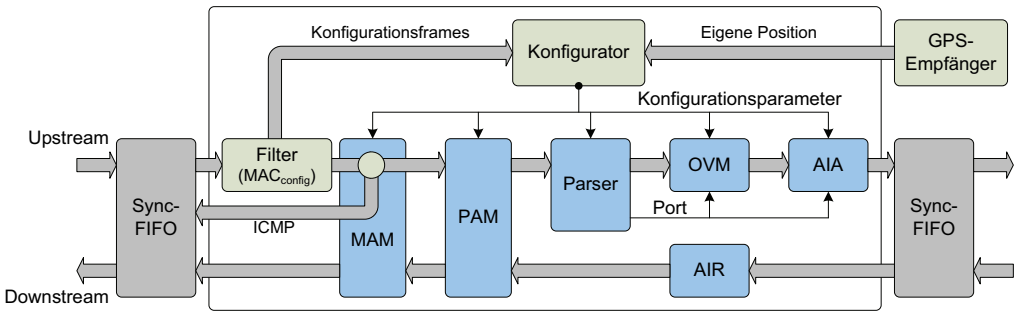


Abbildung B.3.: Architektur des Hardware-Prototyps des IPclip-Systems

eigentlichen IPclip-System orthogonales Konfiguratormodul umgeleitet werden. Das Konfiguratormodul extrahiert die notwendigen Informationen und steuert die entsprechenden Teilmodule zur Laufzeit. Zu den wichtigsten Parametern zählen u. a. die eigene GPS-Position (direkt vom GPS-Empfänger bereitgestellt), die SCA, die Adresstabellen des Parsers sowie MTU_{config} und L_{IP-Opt}.

Der nächste Abschnitt stellt die für den IPclip-Prototyp entwickelten Konfigurations- und Visualisierungstools vor.

B.3. Konfigurations- und Analysetool des IPclip-Prototyps

Für den Prototyp des IPclip-Mechanismus wurden zur Konfiguration von Systemparametern, zur Analyse und Verifikation der korrekten Funktion sowie zur Prototyp-Demonstration [DKW+08a] grafische Tools entwickelt. Zwei unabhängige Tools waren notwendig, um Sender und Empfänger in einer IPclip-fähigen Testumgebung physisch voneinander trennen zu können. Zur Entwicklung beider Tools wurde die Qt-Entwicklungsumgebung von Trolltech [QT] genutzt. Die freie WinPCap-Bibliothek [WPC] stellt die nötigen Netzwerkfunktionen bereit.

Abbildung B.4 zeigt das Konfigurationstool. Die Hauptfunktionen des Konfigurators sind:

- Zuweisung von Systemparametern des Hardware-Prototyps auf dem Entwicklungsboard, z. B. der SCA oder der eigenen MAC- und IP-Adresse.
- Festlegung der eigenen geographischen Position und ID des Prototyps (in GPS- oder in GLI-Koordinaten)
- Definition von Regeln für die Konfiguration der (Adress)-Tabellen des Prototyps
- Speichern & Laden von Konfigurationen im XML-Datenformat

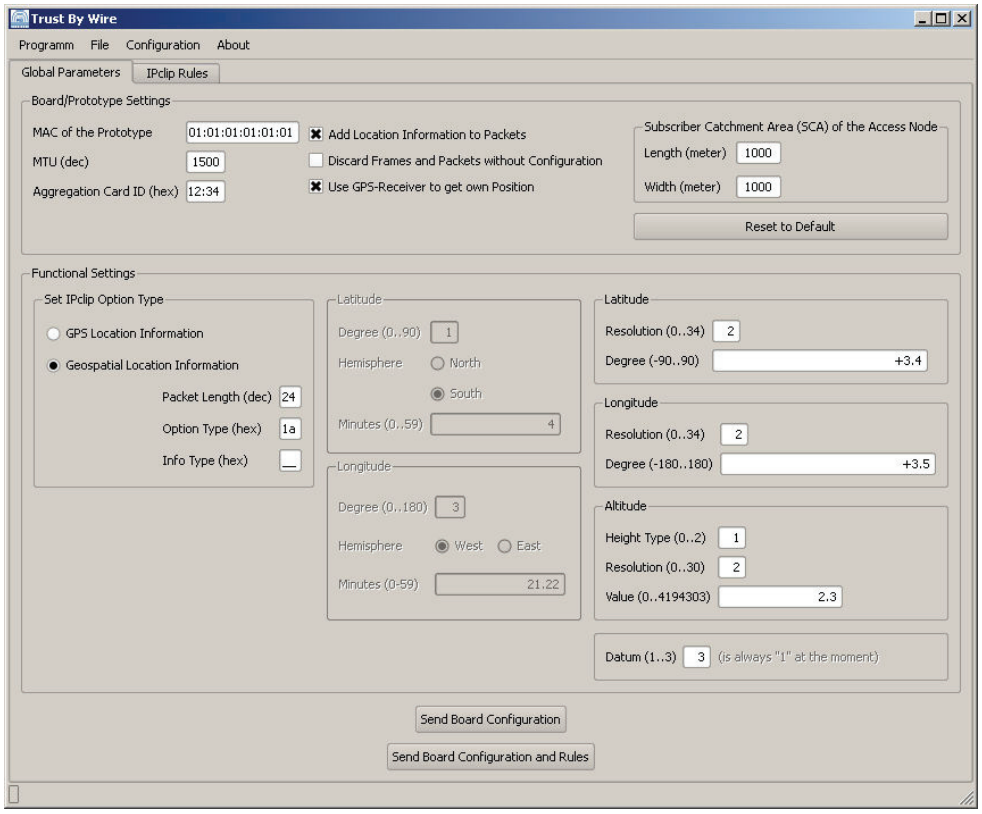


Abbildung B.4.: IPclip-Konfigurationstool, Ansicht für allg. Systemparameter

Abbildung B.5 zeigt sowohl das Analysetool als auch das zur Visualisierung der Ortsreferenzen genutzte Tool GoogleEarth [GOO] (im Bild Darstellung der SCA eines Zugangsknotens). Die Hauptfunktionen des Analysators sind u. a.:

- Das Empfangen und die Analyse von Ethernet-Frames
- Untersuchung auf IP-Pakete und möglicherweise vorhandenen IPclip-Optionen
- Extraktion der enthaltenen LI, des Optionstyps und der Status-Bits
- Konfiguration und Darstellung der SCA und ID eines beliebigen Zugangsknotens in GoogleEarth (Abbildung B.5)
- Darstellung der extrahierten LI in GoogleEarth durch Nutzung des offenen Interfaces Google Earth COM API [COM]

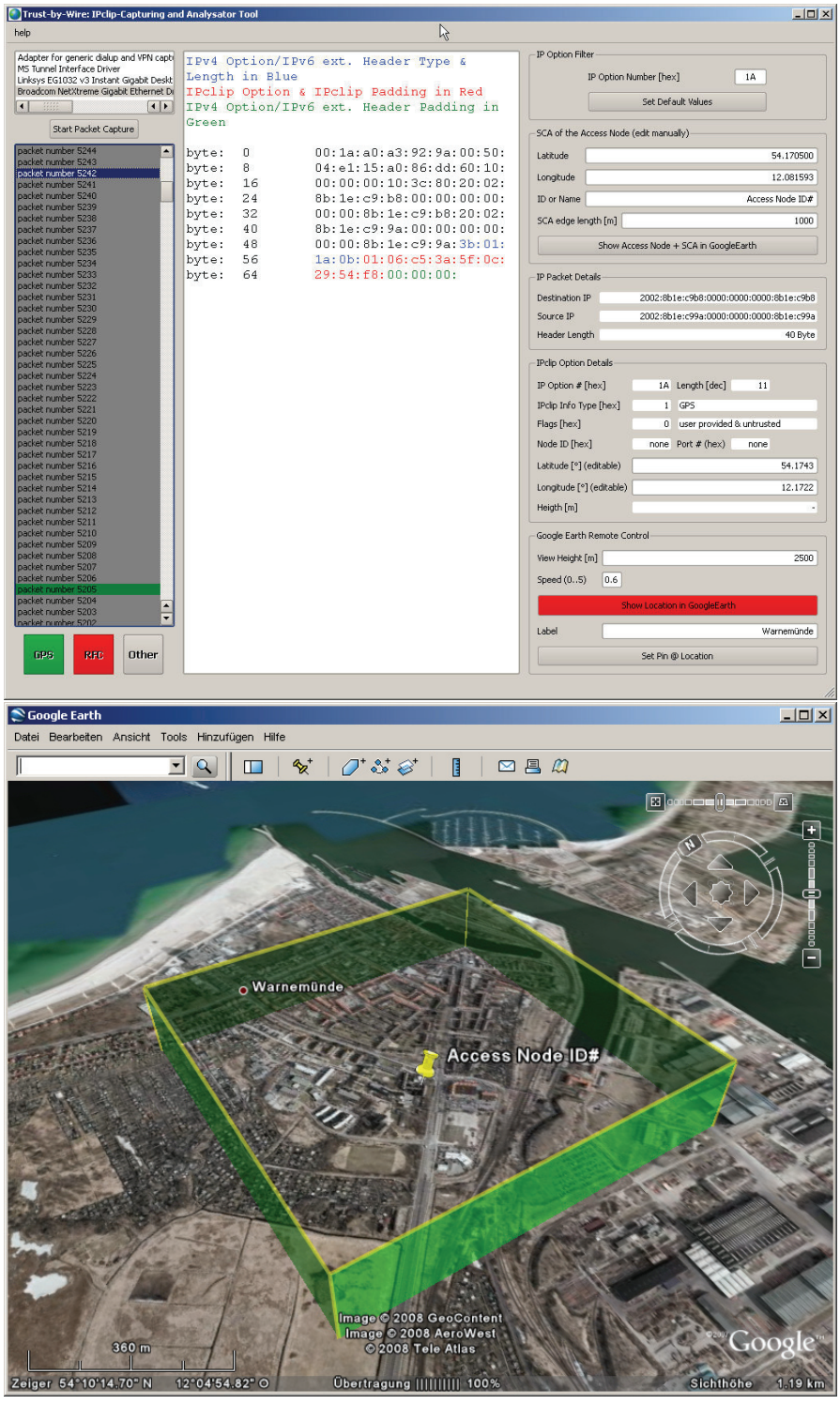


Abbildung B.5.: Analysetool des IPclip-Prototyps (oben) und Visualisierung mit Google Earth

Victory is the beautiful, bright coloured flower.
Transport is the stem without it could never have blossomed.
(Winston Churchill, brit. Politiker & Nobelpreisträger)

Anhang C.

Networks-on-Chip

C.1. Flusskontrollverfahren

In Abschnitt 6.2 wurden verschiedene Flusskontrollverfahren beschrieben. Im Folgenden sind zum besseren Verständnis Ansichten des Simulationstools ModelSim gezeigt. Es wird die blockierungsfreie Übertragung eines Pakets ($n_{blocked} = 0$, $n_{flits} = 10$) in einem einfachen NoC simuliert. Abbildung C.1 skizziert das Szenario. Das NoC ist ein 3-facher 2-Würfel. Der Übertragungspfad von der Quelle zur Senke ($0;0 \Rightarrow 2;2$) auf Basis von XY-Routing ist hervorgehoben ($d = 5$). Die Bezeichnungen der Übertragungskanäle entsprechen den Signalnamen in den Wellendiagrammen C.2, C.3, C.4 und C.5. Die Simulation wurde mit einer Frequenz von $f_{NoC} = 250$ MHz ($T_{clk} = 4$ ns) durchgeführt.

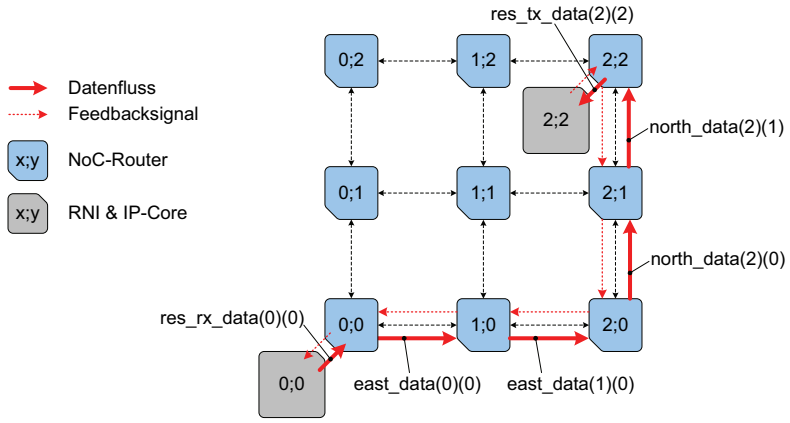


Abbildung C.1.: Simulationsszenario einer Nachrichtenübertragung in einem 3-fachen 2-Würfel

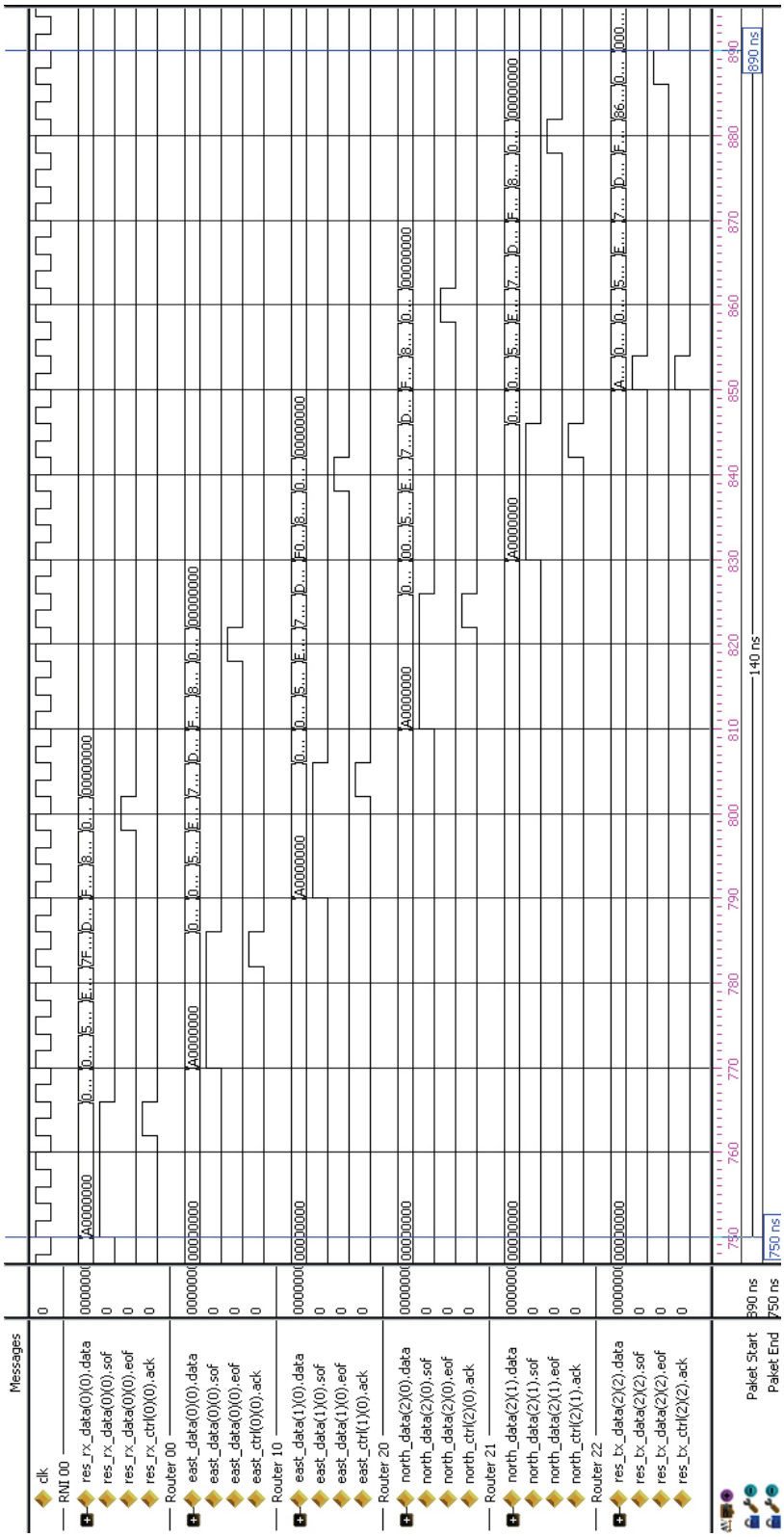


Abbildung C.2.: Simulation einer Nachrichtenübertragung für VCTS. $t_{0,VCTS}$ berechnet sich nach Formel (6.9) und $n_{route} = 5$ zu $t_{0,VCTS} = 140$ ns.

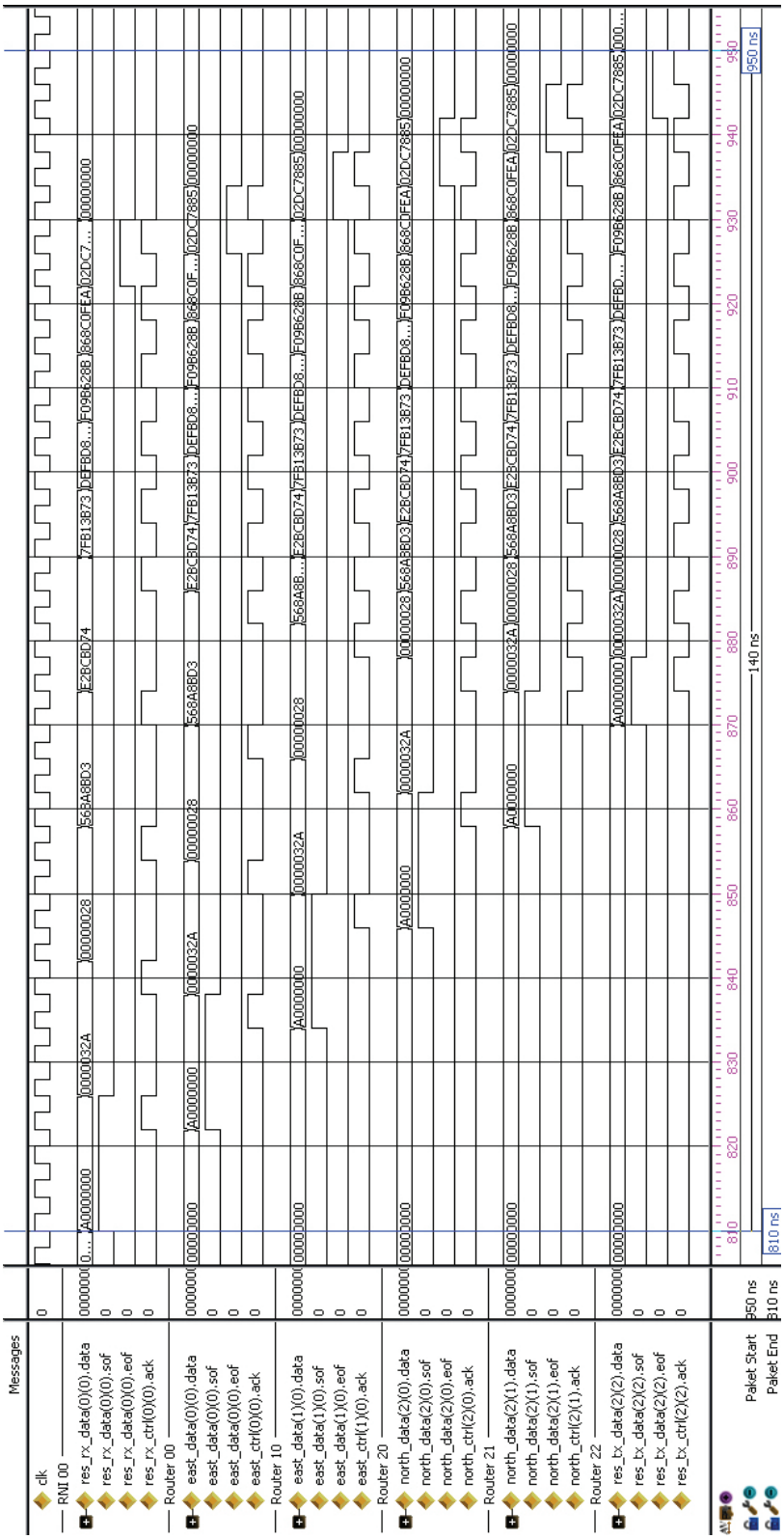


Abbildung C.3.: Simulation einer Nachrichtenübertragung für GWHS. $t_{0,GWHS}$ berechnet sich nach Formel (6.10) und $n_{route} = 3$ zu $t_{0,GWHS} = 140$ ns.

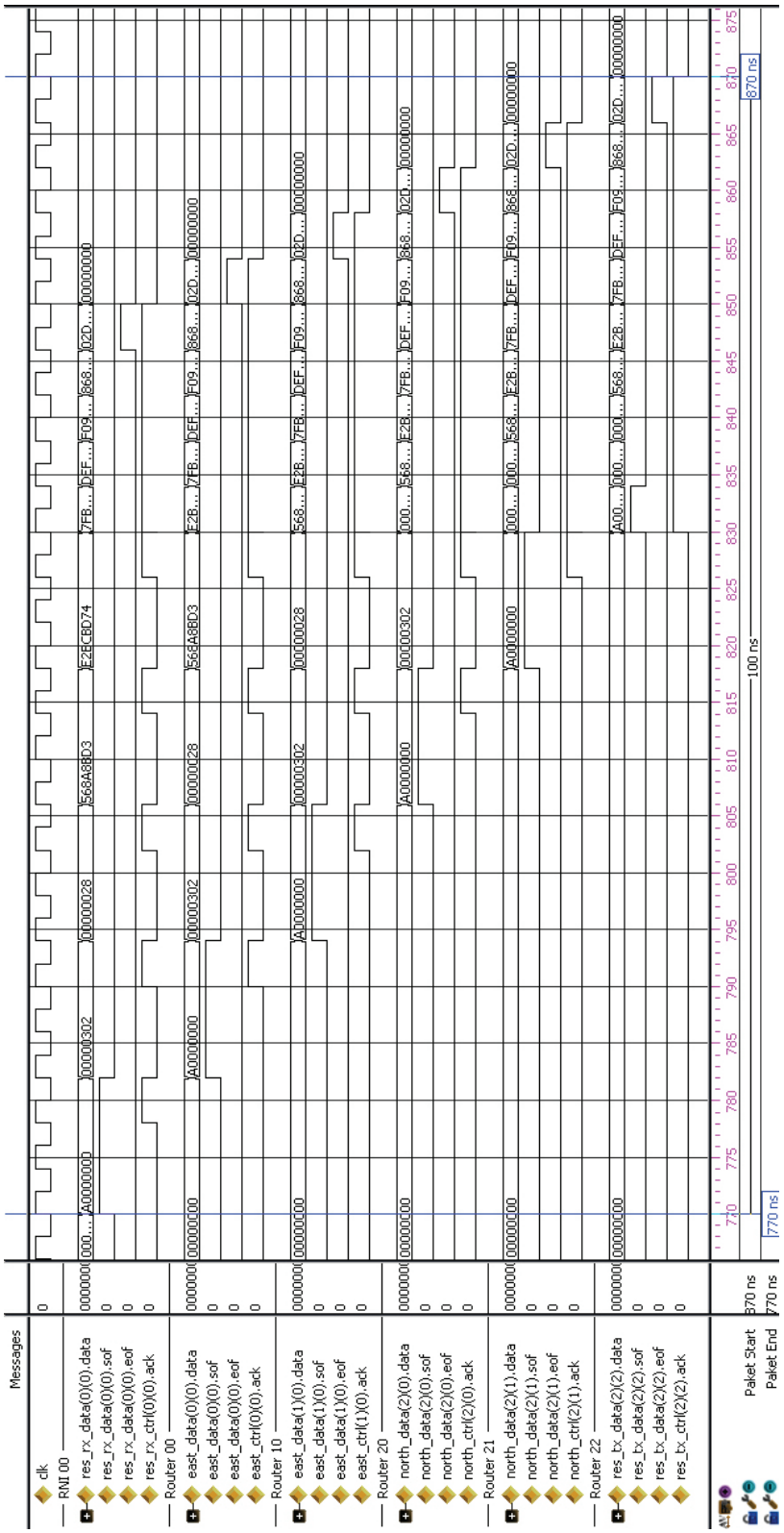


Abbildung C.4.: Simulation einer Nachrichtenübertragung für KWHS. $t_{0,KWHS}$ berechnet sich nach Formel (6.9) und $n_{route} = 3$ zu $t_{0,KWHS} = 100$ ns.

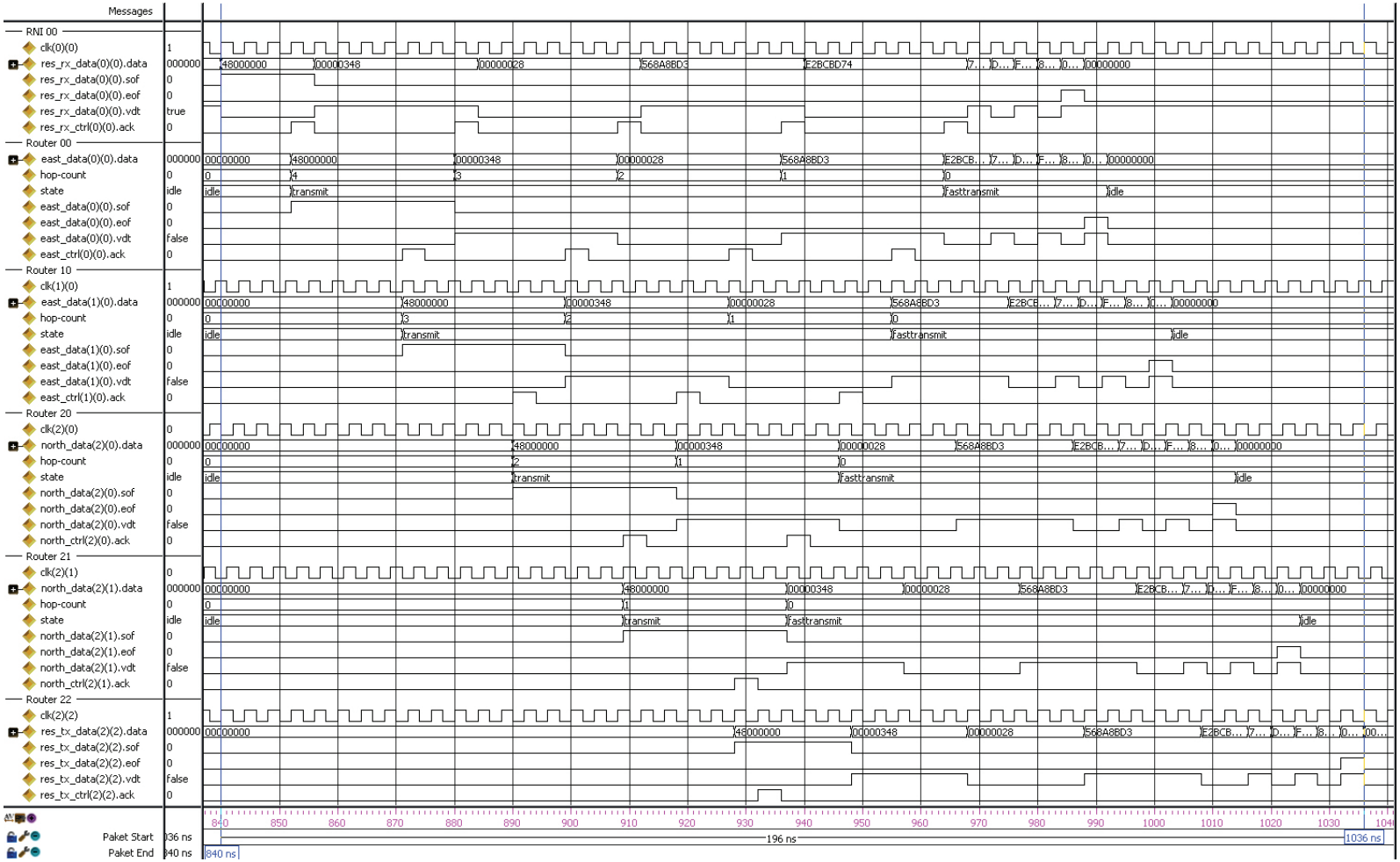


Abbildung C.5.: Simulation einer Nachrichtenübertragung für HSM. Die obere und untere Grenze von $t_{0,HSM}$ berechnen sich nach Formel (6.14) zu 196 ns $\leq t_{0,HSM} \leq 228$ ns. Die im Diagramm dargestellte $t_{0,HSM}$ stellt die untere Schranke dar.

C.2. Anpassung von XY-Routing an BEAM-Topologien

Abschnitt 6.4 stellt mit BEAM eine angepasste Topologie für ein schlankes NoC auf FPGA-Basis vor. An dieser Stelle sind Auszüge des VHDL-Quellcodes dargestellt, um die Modifikationen des XY-Routing-Algorithmus für die BEAM-Topologie zu demonstrieren. Prinzipiell beruht das in Abbildung C.6 dargestellte herkömmliche XY-Routing auf dem Vergleich der jeweiligen Router-Adresse (`own`) mit der Zieladresse des Pakets (`dest`). Durch die Schachtelung der `if`-Anweisungen wird die Reihenfolge der Dimensionen, in denen geroutet wird, streng eingehalten.

```
1  function xy_routing(
2    own  : NOC_ADRESS_TYPE;   -- Router-Adresse
3    dest : NOC_ADRESS_TYPE)   -- Zieladresse
4    return string is
5  begin
6    if own = dest then        -- Wenn Router-Adresse gleich Zieladresse
7    | return "local";         -- Paket zum lokalen IP-Core
8    else                      -- prüfe X-Adresse
9    | if own(x) > dest(x) then
10   | | return "west";        -- Paket zum West-Port
11   | elsif own(x) < dest(x) then
12   | | return "east";        -- Paket zum Ost-Port
13   | else                    -- falls X-Koordinate korrekt, prüfe Y-Koordinate
14   | | if own(y) > dest(y) then
15   | | | return "south";     -- Paket zum Süd-Port
16   | | else
17   | | | return "north";     -- Paket zum Nord-Port
18   | | end if;
19   | end if;
20   end if;
21 end xy_routing;
```

Abbildung C.6.: Herkömmlicher XY-Routing-Algorithmus in einem k-fachen 2-Würfel

In einer BEAM-Topologie ist in Routern der westlichen und östlichen Kante sowie an den Ecken des NoC-Gitters der XY-Algorithmus anzupassen (siehe Abschnitt 6.4.2). Abbildung C.7 zeigt die Funktion des modifizierten XY-Routings für den südwestlichen Router einer BEAM-Topologie. Der blau hervorgehobene Abschnitt stellt den wesentlichen Unterschied dar. Die Reihenfolge der Routing-Dimensionen wird abhängig von der X-Koordinate verändert. Die Modifikationen für den südöstlichen, nordwestlichen und nordöstlichen Eck-Router einer BEAM-Topologie verhalten sich analog. Abbildung C.8 zeigt die Anpassungen für Router der westlichen Kante einer BEAM-Topologie, die *keine* Eck-Router sind. Die Änderungen an der gegenüberliegenden Kante verhalten sich ebenfalls analog. Router der nördlichen und südlichen Kante und im Zentrum einer BEAM-Topologie benötigen keine Anpassungen.

```
1  function xy_routing_beam_southwest(  [...]
2  begin
3    if own = dst then          -- Wenn Router-Adresse gleich Zieladresse
4    | return "local";          -- Paket zum lokalen IP-Core
5    else                       -- Modifikation für südwestliche Ecke
6    | if dst(x) = 0 then       -- Wenn nach Westen gesendet werden soll...
7    | | if dst(y) = 1 then     -- ...prüfe erst die Y-Koordinate
8    | | | return "west";      -- Paket zum West-Port
9    | | else                  -- weil dst(y) > 1 sein muss
10   | | | return "north";     -- Paket zum Nord-Port
11   | | end if;
12   | elsif dst(x) > 1 then   -- Wenn nach Osten gesendet werden soll, tue das!
13   | | return "east";         -- Paket zum Ost-Port
14   | elsif dst(x) = 1 then   -- falls X-Koordinate korrekt, prüfe Y-Koordinate
15   | | if dst(y) < 1 then
16   | | | return "south";     -- Paket zum Süd-Port
17   | | else
18   | | | return "north";     -- Paket zum Nord-Port
19   | | end if;
20   | end if;
21   end if;
22 end xy_routing_beam_southwest;
```

Abbildung C.7.: XY-Routing für den südwestlichen Eck-Router einer BEAM-Topologie

```
1  function xy_routing_beam_west(  [...]
2  begin
3    if own = dst then            -- Wenn Router-Adresse gleich Zieladresse
4    | return "local";            -- Paket zum lokalen IP-Core
5    else                         -- Modifikation für westliche Kante
6    | if dst(x) = 0 then         -- Wenn nach Westen gesendet werden soll...
7    | | if dst(y) = own(y) then -- ...prüfe erst die Y-Koordinate
8    | | | return "west";        -- Paket zum West-Port
9    | | elsif dst(y) < own(y) then
10   | | | return "south";       -- Paket zum Süd-Port
11   | | else                    -- weil dst(y) > 1 sein muss
12   | | | return "north";       -- Paket zum Nord-Port
13   | | end if;
14   | elsif dst(x) > 1 then     -- Wenn nach Osten gesendet werden soll, tue das!
15   | | return "east";          -- Paket zum Ost-Port
16   | elsif dst(x) = 1 then     -- falls X-Koordinate korrekt, prüfe Y-Koordinate
17   | | if dst(y) < own(y) then
18   | | | return "south";       -- Paket zum Süd-Port
19   | | else
20   | | | return "north";       -- Paket zum Nord-Port
21   | | end if;
22   | end if;
23   end if;
24 end xy_routing_beam_west;
```

Abbildung C.8.: XY-Routing für Router an der westlichen Kante einer BEAM-Topologie

C.3. Framerate verschiedener Ethernet-Varianten

Die theoretisch maximale Anzahl von Frames pro Sekunde, die Framerate (FR), ist abhängig von der Framelänge L_{Frame} und der Bandbreite BW_{Eth} der Ethernet-Variante. FR kann nach Formel C.1 ermittelt werden. Pro Ethernet-Frame müssen zusätzlich 20 Byte für Inter Frame Gap, Präambel und Start Frame Delimiter einbezogen werden [IEE05]. Tabelle C.1 enthält Werte für FR verschiedener Ethernet-Varianten von 10 Mbit/s bis 10 Gbit/s. Zudem ist der effektive Datendurchsatz von Ethernet Θ_{Eth} durch Inter Frame Gap, Präambel und Start Frame Delimiter geringer als die absolute physikalische Bandbreite. Die 4 Byte große FCS wird nicht mit zu den Nutzdaten gezählt, da sie an den physikalischen Ethernet-Schnittstellen i. Allg. abgetrennt und erst beim Versenden wieder neu berechnet wird. Θ_{Eth} ergibt sich somit nach Formel (C.2). Tabelle C.2 enthält Werte für Θ_{Eth} für verschiedene Ethernet-Ausprägungen.

$$FR = \frac{BW_{Eth}}{8 \cdot (L_{Frame} + 20)} \tag{C.1}$$

$$\Theta_{Eth} = \frac{(L_{Frame} - 4) \cdot BW_{Eth}}{L_{Frame} + 20} \tag{C.2}$$

Tabelle C.1.: Maximale Framerate verschiedener Ethernet-Varianten in Frames/s

L_{Frame} [B]	64	128	256	512	1024	1280	1518
10 Mbit/s	14881	8446	4529	2350	1198	962	813
100 Mbit/s	148810	84460	45290	23497	11973	9616	8128
1 Gbit/s	1488096	844595	452899	234963	119732	96154	81275
10 Gbit/s	14880952	8445946	4528986	2349625	1197318	961539	812744

Tabelle C.2.: Effektiver Durchsatz verschiedener Ethernet-Varianten in bit/s

L_{Frame} [B]	64	128	256	512	1024	1280	1518
10 Mbit/s	7,14 M	8,38 M	9,13 M	9,55 M	9,77 M	9,81 M	9,84 M
100 Mbit/s	71,4 M	83,8 M	91,3 M	95,5 M	97,7 M	98,2 M	98,4 M
1 Gbit/s	714,2 M	837,8 M	913 M	954,9 M	977 M	981,5 M	984,4 M
10 Gbit/s	7,14 G	8,38 G	9,13 G	9,55 G	9,77 G	9,81 G	9,84 G

Erklärung

Ich erkläre, dass ich die eingereichte Dissertation selbstständig und ohne fremde Hilfe verfasst, die von mir genutzten Quellen und Hilfsmittel angegeben und den benutzten Werken wörtlich oder inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Rostock, 09.03.2009

Stephan Kubisch

curriculum vitæ

Persönliche Daten

Stephan Kubisch

✉ Gellertstrasse 10
18057 Rostock

☎ 0176 22 11 11 17

@ stephan.kubisch@gmx.de

Geb. am 26. 11. 1978 in Rostock
Ledig, deutsch

Schulausbildung

08/1985–07/1990	Polytechnische Oberschule Graal-Müritz
08/1991–06/1997	Gymnasium Rövershagen Abitur mit Auszeichnung

Zivildienst

08/1997–08/1998	Ableistung des Zivildienstes im AKG Reha-Zentrum Graal-Müritz

Studium

10/1998–03/2004	Studium der Informationstechnik/Technischen Informatik an der Universität Rostock, Schwerpunkt Informationstechnik Diplomingenieur Informationstechnik
10/2001–02/2002	Studienbegleitendes Praktikum bei Texas Instruments in Freising, Bereich analoges und digitales Schaltungsdesign

Berufserfahrung

02/1999-06/2000	Werkstudent bei der DGW-Datennetze GmbH, Niederlassung Rostock, Bereich Sparkassen-, Finanz- und Netzwerktechnik
seit 04/2004	Wissenschaftlicher Mitarbeiter am Institut für Angewandte Mikroelektronik und Datentechnik der Universität Rostock

Forschungsaktivitäten & Projekte

DFG O.C.	DFG Schwerpunktprogramm Organic Computing *Energy Aware Self Organized Communication in Complex Networks* ℹ `http://www.organic-computing.de/spp`
LFS IuK	Landesforschungsschwerpunkt Informations- und Kommunikationstechnik zum Thema *Multimediales Content-Management in mobilen Umgebungen mit multimodalen Nutzungsschnittstellen* Teilgruppe *Wired Networks* ℹ `http://www.m6c.de`
NSN	Kooperation mit dem Industriepartner Nokia Siemens Networks, Broadband Access Division, Greifswald *Ethernet-basierte Zugangsnetzarchitekturen* ℹ `http://www.imd.uni-rostock.de/networking`
NoC	Forschungen im Bereich Networks-on-Chip ℹ `http://www.networks-on-chip.com`
Lehre	Durchführung/Betreuung von Seminaren der Veranstaltungen *Spezielle Anwendungen des VLSI-Entwurfs* und *Algorithmen der Datentechnik / Systemgerechte Algorithmen* Betreuung von Praktika *Grundlagen Elektrotechnik* *VHDL-Entwurf* Betreuung zahlreicher Beleg-, Studien- und Diplomarbeiten

Rostock, 09.03.2009

Architekturen für Ethernet-basierte Teilnehmerzugangsnetzwerke und deren Umsetzung in Hardware

Thesen der Dissertation
zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)
der Fakultät für Informatik und Elektrotechnik
der Universität Rostock

vorgelegt von
Kubisch, Stephan, geb. am 26.11.1978 in Rostock
Rostock, 09.03.2009

Thesen

1. Die Begriffe Skalierbarkeit und Komplexität spielen in (Tele)Kommunikationsnetzen – sowohl makroskopische Datennetze als auch mikroskopische Chip-interne Verbindungsstrukturen – eine entscheidende Rolle.
2. Der Telekommunikationssektor ist momentan durch starke Globalisierungs- und Konsolidierungsprozesse gekennzeichnet. Ehemals unterschiedliche dienstbezogene Netze wachsen zusammen. Klassische Kommunikationsdienste wie Rundfunk und Telefonie migrieren in das Internet. Durch das Wachstum des Internets und dessen Integration in den Alltag entstehen neue ubiquitäre digitale Dienste. Der Nutzwert des Internets steigt.
3. Das Internet hat sich von einem reinen Wissenschaftsnetz zu einem globalen Kommunikationsmedium entwickelt. Die Anforderungen an die Netzinfrastruktur und Netzwerkdienste haben sich radikal verändert, z. B. bezogen auf Sicherheit und Skalierbarkeit. Das ursprüngliche Internet und dessen Kernprotokolle sind nicht für die zunehmenden Nutzerzahlen und Dienstevielfalt entworfen worden, die sich aus der Konvergenz der Netze ergeben.
4. Die Bedeutung der sogenannten Teilnehmerzugangsnetze wächst, da diese die zunehmend intelligentere Schnittstelle zum breitbandigen aber funktional unterspezifizierten Kernnetz darstellen. ATM wird durch Ethernet-basierte Übertragungstechnologien abgelöst, welche aufgrund ihrer Kosteneffizienz, Einfachheit, Vielseitigkeit und Leistungsfähigkeit eine immer größere Rolle im Bereich der Teilnehmerzugangsnetze spielen.
5. Durch steigende Nutzerzahlen vergrößert sich die Anzahl der Ethernet-MAC-Adressen im Zugangs- und Kernsegment. Dies erhöht die Arbeitslast paketverarbeitender Geräte in diesen Bereichen und kann zu Tabellenüberläufen führen.
6. MAC Address Translation (MAT) ist ein Verfahren, das die Anzahl der MAC-Adressen im Kernbereich senkt und Tabellenüberläufe verhindert. Gleichzeitig wird die Sicherheit auf Ebene der Sicherungsschicht erhöht, da grundlegende Bedrohungsszenarien wie Spoofing und Flooding durch die Eigenschaften des MAT-Verfahrens verhindert werden.
7. Die inhärenten Eigenschaften des paketvermittelten und per se offenen Internets unterscheiden sich stark von klassischen weitestgehend leitungsvermittelten Kommunikationsnetzen, z. B. zur Sprachkommunikation. Die allgemeine Erwartungshaltung ist, dass sowohl migrierenden als auch neue digitale Dienste trotz der veränderten technologischen Basis ein vergleichbares bzw. höheres Niveau an Sicherheit und Funktionalität gewährleisten.
8. Internet Protocoll Calling Line Identification Presentation (IPclip) ist eine neue Sicherheitsarchitektur auf Ebene der Vermittlungsschicht, welche einerseits IP-Pakete mit Ortsreferen-

zen zu deren Ursprung anreichert und andererseits die Konsistenz und Glaubhaftigkeit dieser Informationen auch über Providergrenzen hinaus wahrt. IPclip schafft dadurch ein Vertrauensverhältnis zwischen Kommunikationsendpunkten.

9. Durch die Garantie eines Vertrauensverhältnisses und einer Art von Senderidentifikation bietet IPclip in einer Vielzahl aktueller Anwendungsszenarien, z. B. Notrufe über Voice-over-IP, einen Mehrwert bezogen auf Sicherheit und Funktionalität.
10. Anwendungen und Funktionen der Paketverarbeitung werden zunehmend in Hardware umgesetzt, da die Leistungsfähigkeit Software-basierter Lösungen den steigenden Anforderungen der Dienste und den hohen Datenvolumina nicht mehr genügt. In der Telekommunikation werden dazu hoch integrierte Schaltkreise in Form von Systems-on-Chip (SoC) und feldprogrammierbaren Bausteinen verwendet.
11. Die Komplexität integrierter Bausteine steigt durch verringerte physikalische Strukturgrößen sowie durch die Zunahme des Funktionsumfangs in SoCs und der Anzahl modularer Funktionsgruppen. Klassische SoC-Systemarchitekturen stoßen dadurch an ihre physikalischen, funktionalen und ökonomische Grenzen.
12. Neue Architekturansätze für SoCs sind notwendig. Networks-on-Chip (NoCs) sind ein möglicher Lösungsansatz für Chip-interne Verbindungs- und Kommunikationsstrukturen. NoCs besitzen auf verschiedenen Ebenen entscheidende Vorteile gegenüber klassischen Ansätzen: Verringerung parasitärer physikalischer Einflüsse, Wiederverwendbarkeit und Modularität, Abstraktion, Skalierbarkeit, Parallelität.
13. Im NoC-Entwurf können Langzeiterfahrungen aus dem Bereich makroskopischer Datennetze wie dem Internet nachgenutzt werden. Durch funktionale Unterspezifikation der NoC-Infrastruktur und der Verlagerung von Funktionalität in die Schnittstellen – dem „Teilnehmerzugang“ zum NoC – wird ein schlankes und performantes NoC realisiert.
14. Der für eine mesochron angesteuerte NoC-Infrastruktur entwickelte Hybride Switching-Mechanismus (HSM) ist nach diesem Prinzip entworfen. HSM vereint die positiven Eigenschaften verschiedener anderer Flusskontrollverfahren: minimale Puffergrößen, niedrige Übertragungslatenz, effiziente Bandbreitenausnutzung.
15. Die in der Arbeit neu entwickelte Border-Enhanced-Mesh-Topologie (BEAM) für NoCs ist eine kostenfreie Weiterentwicklung k-facher 2-Würfel. Der BEAM-Ansatz senkt den Hardwarebedarf der Kommunikationsinfrastruktur. Darüber hinaus wird die durchschnittliche Pfadlänge im NoC und damit die durchschnittliche Kommunikationslatenz gesenkt.
16. Durch die Nutzung der durch HSM und BEAM optimierten NoC-Infrastruktur kann die Leistungsfähigkeit eines SoC gegenüber klassischen Architekturansätzen erhöht werden.

Vollständige Liste eigener Veröffentlichungen und Fachvorträge

Die in der Dissertation dargestellten Inhalte wurden zum Teil auf Konferenzen und Tagungen veröffentlicht. Neben den in der Arbeit bereits referenzierten Publikationen sind an dieser Stelle auch zusätzliche Beiträge und Veröffentlichungen aufgeführt, die nicht direkt mit den Themengebieten der Dissertationsschrift in Zusammenhang stehen.

[1] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; ROHRBECK, Jens ; ALTMAN, Vladyslav ; SKODZIK, Jan ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk: Trust-by-Wire in Packet-Switched IPv6 Networks: Tools and FPGA Prototype for the IPclip System. In: *Proceedings of the 6th Annual IEEE Consumer Communications and Networking Conference (CCNC'09)*. Las Vegas, Nevada, USA, Januar 2009. – ISBN 978–1–4244–2309–5, S. 1–2

[2] WIDIGER, Harald ; KUBISCH, Stephan ; DANIELIS, Peter ; SCHULZ, Jens ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk: IPclip: An Architecture to restore Trust-by-Wire in Packet-switched Networks. In: *Proceedings of the 33rd IEEE Conference on Local Computer Networks (LCN'08)*. Montreal, Quebec, Kanada, Oktober 2008. – ISBN 978–1–4244–2413–9, S. 312–319

[3] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: Trust-by-Wire in Packet-switched IP Networks: Calling Line Identification Presentation for IP. In: *Proceedings of the 1st ITU-T Kaleidoscope Conference: Innovations in Next Generation Networks - Future Network and Services (K-INGN'08)*. Genf, Schweiz, Mai 2008. – ISBN 92–61–12441–0, S. 375–382

[4] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; TIMMERMANN, Dirk: A Conceptual Framework for Increasing Physical Proximity in Unstructured Peer-To-Peer Networks. In: *Proceedings of the 2008 IEEE Sarnoff Symposium (auf CD-Rom)*. Princeton, New Jersey, USA, April 2008. – ISBN 978–1–4244–1843–5

[5] KUBISCH, Stephan: Networks: Complexity and Scalability. In: *TCPP PhD-Forum of the 22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08)*. Miami, Florida, USA, April 2008. – `http://www.ipdps.org/ipdps2008/2008_phdforum.html`

[6] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: Countering Phishing Threats with Trust-by-Wire in Packet-switched IP Networks – A Conceptual Framework. In: *Proceedings of the 22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08), 4th International Workshop on Security in Systems and Networks (SSN'08) (auf CD-Rom)*. Miami, Florida, USA, April 2008. – ISBN 978–1–4244–1694–3

[7] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: IPclip – An Innovative Mechanism to Reestablish Trust-by-Wire in Packet-switched IP Networks. In: *3. Essener Workshop „Neue Herausforderungen in der Netzsicherheit" (EWNS 2008)*. Essen, Deutschland, April 2008. – http://wiki.uni-due.de/TdR/index.php/Essener_Workshop_zur_Netzsicherheit_2008_%28EWNS08%29

[8] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; SCHULZ, Jens ; TIMMERMANN, Dirk ; BAHLS, Thomas ; DUCHOW, Daniel: Complementing E-Mails with Distinct, Geographic Location Information in Packet-switched IP Networks. In: *Proceedings of the 2008 MIT Spam Conference (auf CD-Rom)*. Cambridge, Massachusetts, USA, März 2008. – http://www.spamconference.org

[9] DANIELIS, Peter ; KUBISCH, Stephan ; WIDIGER, Harald ; SCHULZ, Jens ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; LANGE, Christian: Trust-by-Wire in Packet-switched IP Networks: Calling Line Identification Presentation for IP – Hardware Prototype Demonstration. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE'08), University Booth*. München, Deutschland, März 2008. – http://www.edacentrum.de/eda-netzwerke/universitybooth/ubooth08-full-program.pdf

[10] KUBISCH, Stephan ; HEINRICH, Enrico ; TIMMERMANN, Dirk: A Mesochronous Network-on-Chip for an FPGA. In: *Proceedings of the Annual Doctoral Workshop on Mathematical and Engineering Methods in Computer Science (MEMICS)*. Znojmo, Tschechische Republik, Oktober 2007. – ISBN 978–80–7355–077–6, S. 113–120, ** ausgezeichnet mit einem Best Paper Award **

[11] DANIELIS, Peter ; KUBISCH, Stephan ; TIMMERMANN, Dirk: Realisierung und Implementierung eines Algorithmus zur Echtzeit-Mustererkennung in einem Ethernet-Datenstrom. In: *Tagungsband des 12. Symposium Maritime Elektrotechnik, Elektronik und Informationstechnik*. Rostock, Deutschland, Oktober 2007. – ISBN Universitätsdruckerei Rostock 685–07, S. 191–196

[12] WIDIGER, Harald ; KUBISCH, Stephan ; TIMMERMANN, Dirk: A Structural Architecture for HW Packet Processing. In: *Proceedings of the IEEE 11th Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM'07)*. Victoria, B.C., Kanada, August 2007. – ISBN 1–4244–1190–4, S. 363–366

[13] KUBISCH, Stephan ; CORNELIUS, Claas ; HECHT, Ronald ; TIMMERMANN, Dirk: Mapping a Pipelined Data Path onto a Network-on-Chip. In: *Proceedings of the IEEE 2nd International Symposium on Industrial Embedded Systems (SIES'07)*. Lissabon, Portugal, Juli 2007. – ISBN 1–4244–0840–7, S. 178–185

[14] KUBISCH, Stephan ; WIDIGER, Harald ; TIMMERMANN, Dirk ; DUCHOW, Daniel ; BAHLS, Thomas: sMAT – A Simplified MAC Address Translation Scheme. In: *Proceedings of the 15th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN'07) (auf CD-Rom)*. Princeton, New Jersey, USA, Juni 2007. – ISBN 1–4244–1100–9. – http://www.ieee-lanman.org

[15] SALZMANN, Jakob ; KUBISCH, Stephan ; REICHENBACH, Frank ; TIMMERMANN, Dirk: Energy and Coverage Aware Routing Algorithm in Self Organized Sensor Networks. In: *Proceedings of the 4th International Conference on Networked Sensing Systems (INSS'07)*. Braunschweig, Deutschland, Juni 2007. – ISBN 1–4244–1231–5, S. 77–80

[16] KUBISCH, Stephan ; WIDIGER, Harald ; CORNELIUS, Claas ; TIMMERMANN, Dirk ; STRZELETZ, Andy: E-Core – A Configurable IP Core for Application-specific NoC Performance Evaluation. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE'07), Proceedings of the Workshop on Diagnostic Services in Network-on-Chips*. Nizza, Frankreich, April 2007

[17] KUBISCH, Stephan ; WIDIGER, Harald ; DANIELIS, Peter ; DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; LANGE, Christian ; RÖWER, Oliver: Configuration Tool and FPGA-Prototype of a Hardware Packet Processing System. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE'07), University Booth Proceedings on CD-Rom*. Nizza, Frankreich, April 2007

[18] KUBISCH, Stephan ; WIDIGER, Harald ; HECHT, Ronald ; TIMMERMANN, Dirk ; SIEMROTH, Martin: Architektur einer Flexiblen, Wiederverwendbaren Testbench zur Verifikation Paketverarbeitender Hardware in SystemC. In: *Proceedings of the 10th Workshop Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen (MBMV'07)*. Erlangen, Deutschland, März 2007. – ISBN 978–3–8322–5965–3, S. 9–18

[19] KUBISCH, Stephan ; WIDIGER, Harald ; HECHT, Ronald ; TIMMERMANN, Dirk: Network-on-Chip Communication Grids for High Performance Packet Processing (Poster & Abstract). In: *Proceedings of the 2007 ACM/SIGDA 15th International Symposium on Field Programmable Gate Arrays (FPGA'07)*. Monterey, Kalifornien, USA, Februar 2007. – ISBN 978–1–59593–600–4, S. 228

[20] KUBISCH, Stephan ; RENNERT, Rüdiger ; PFÜLLER, Hartmut ; TIMMERMANN, Dirk: LoGen – Generation and Simulation of Digital Logic on the Gate-Level via Internet. In: *Proceedings of the 1st IEEE International Conference on E-Learning in Industrial Electronics (ICELIE'06)*. Hammamet, Tunesien, Dezember 2006. – ISBN 1–4244–0324–3

[21] WIDIGER, Harald ; KUBISCH, Stephan ; TIMMERMANN, Dirk ; BAHLS, Thomas: An Integrated Hardware Solution for MAT, MPLS-UNI, and TM in Access Networks. In: *Proceedings of the 31st Annual IEEE Conference on Local Computer Networks (LCN)*. Tampa, Florida, USA, November 2006. – ISBN 1–4244–0419–3, S. 272–279

[22] DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; WIDIGER, Harald ; KUBISCH, Stephan: ACIP: An Access Control and Information Protocol for Ethernet-based Broadband Access Networks. In: *Proceedings of the 12th International Telecommunications Network Strategy and Planning Symposium (Networks'06) on CD-Rom*. Neu Delhi, Indien, November 2006. – ISBN 978–3–8007–2999–9

[23] CORNELIUS, Claas ; KUBISCH, Stephan ; TIMMERMANN, Dirk: Rechnergestützter Entwurf – Internet im Taschenformat. In: *Landestechnologieanzeiger Mecklenburg-Vorpommern 3/2006* (2006), Oktober, S. 17

[24] KUBISCH, Stephan ; HECHT, Ronald ; SALOMON, Ralf ; TIMMERMANN, Dirk: Intrinsic Flexibility and Robustness in Adaptive Systems: A Conceptual Framework. In: *Proceedings of the 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals/06)*. Logan, Utah, USA, Juli 2006. – ISBN 1–4244–0166–6, S. 98–103

[25] SILL, Frank ; CORNELIUS, Claas ; KUBISCH, Stephan ; TIMMERMANN, Dirk: Mixed Gates: Leakage Reduction techniques applied to Switches for Networks-on-Chip. In: *Proceedings of the Reconfigurable Communication-centric Systems-on-Chip Workshop 2006 (ReCoSoC'06)*. Montpellier, Frankreich, Juli 2006. – ISBN 2–9517461–2–1, S. 76–82

[26] KUBISCH, Stephan ; WIDIGER, Harald ; DUCHOW, Daniel ; TIMMERMANN, Dirk ; BAHLS, Thomas: Wirespeed MAC Address Translation and Traffic Management in Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006

[27] WIDIGER, Harald ; KUBISCH, Stephan ; DUCHOW, Daniel ; TIMMERMANN, Dirk ; BAHLS, Thomas: A Simplified, Cost-Effective MPLS Labeling Architecture for Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006

[28] DUCHOW, Daniel ; BAHLS, Thomas ; TIMMERMANN, Dirk ; KUBISCH, Stephan ; WIDIGER, Harald: Efficient Port-based Network Access Control for IP DSLAMs in Ethernet-based Fixed Access Networks. In: *Proceedings of the World Telecommunications Congress 2006 (WTC'06) on CD-Rom*. Budapest, Ungarn, April 30 - Mai 3 2006

[29] HECHT, Ronald ; KUBISCH, Stephan ; MICHELSEN, Harald ; ZEEB, Elmar ; TIMMERMANN, Dirk: A Distributed Object System Approach for Dynamic Reconfiguration. In: *Proceedings of the 20th IEEE International Parallel & Distributed Processing Symposium (IPDPS), 13th Reconfigurable Architectures Workshop (RAW'06)*. Rhodos, Griechenland, April 2006. – ISBN 1–4244–0054–6, S. 8

[30] KUBISCH, Stephan ; HECHT, Ronald ; TIMMERMANN, Dirk: Adaptive Hardware in Autonomous and Evolvable Embedded Systems. In: *Proceedings of the embedded world 2006 Conference*. Nürnberg, Deutschland, Februar 2006. – ISBN 3–7723–0143–6, S. 297–306

[31] HECHT, Ronald ; KUBISCH, Stephan ; HERRHOLTZ, Andreas ; TIMMERMANN, Dirk: Dynamic Reconfiguration with hardwired Networks-on-Chip on future FPGAs. In: *Proceedings of the 15th International Conference on Field Programmable Logic and Applications (FPL'05)*. Tampere, Finnland, August 2005. – ISBN 0–7803–9362–7, S. 527–530

[32] KUBISCH, Stephan ; HECHT, Ronald ; TIMMERMANN, Dirk: Design Flow on a Chip - An Evolvable HW/SW Platform. In: *Proceedings of the 2nd IEEE International Conference on Autonomic Computing (ICAC'05)*. Seattle, Washington, USA, Juni 2005. – ISBN 0–7695–2276–9, S. 393–394

[33] HECHT, Ronald ; TIMMERMANN, Dirk ; KUBISCH, Stephan ; ZEEB, Elmar: Network-on-Chip basierte Laufzeitsysteme fuer dynamisch konfigurierbare Hardware. In: *Lecture Notes in Informatics, Workshop Proceedings of the International Conference on Architecture of Computing Systems 2004 (ARCS 2004) - Organic and Pervasive Computing Workshop*. Augsburg, Deutschland, März 2004. – ISBN 3–88579–370–9, S. 185–194

Kurzreferat

Nutzerzahlen, Datenraten, sowie die Diversität von Medien, Protokollen und Diensten im Internet nehmen zu. Die Integrationsdichte, der Funktionsumfang und die Anzahl paralleler Recheneinheiten digitaler Systeme steigen ebenso. Um sowohl in der Telekommunikation als auch im Entwurf integrierter Digitalschaltungen dieser allgemein wachsenden Systemkomplexität und den damit einhergehenden Problemstellungen zu begegnen, bedarf es innovativer und skalierbarer Konzepte. Aus diesem Grund stehen die Begriffe *Komplexität* und *Skalierbarkeit* zurzeit im Mittelpunkt vieler Forschungsaktivitäten in diesen beiden Bereichen. Effiziente Formen der Kommunikation spielen somit eine entscheidende Rolle – dies gilt für den off-Chip- als auch für den on-Chip-Bereich.

Diese Arbeit analysiert den Ist-Zustand in der Telekommunikation und leitet daraus Anforderungen an zukünftige Kommunikationsinfrastrukturen ab, welche den vorgestellten Lösungsansätzen als Motivation dienen. Im Fokus stehen Aspekte der Netzwerksicherheit im Internet, der Umgang mit zunehmenden Nutzerzahlen sowie Systemarchitekturen paketverarbeitender Hardwarekomponenten. Mit MAC Address Translation (MAT), einem Mechanismus zur Übersetzung von Adressen auf Ebene der Sicherungsschicht, wird im Bereich Ethernet-basierter Teilnehmerzugangsnetze eine Architektur präsentiert, welche sich auf die Aspekte Sicherheit und Skalierbarkeit konzentriert. Durch eine flexibel konfigurierbare Maskierung von MAC-Adressen wird deren absolute Anzahl in Richtung der Kernnetze und damit die Größe der dort verwalteten Adresstabellen reduziert. Zudem werden typische Angriffsszenarien auf der Sicherungsschicht, z. B. Spoofing und Flooding, unterbunden, um die Konsistenz der Adresstabellen zu wahren. Ein weiteres Konzept zur Erhöhung der Sicherheit wird mit IP Calling Line Identification Presentation (IPclip) vorgestellt. Die IPclip-Architektur schlägt eine Brücke zwischen konventionellen leitungsvermittelten und modernen paketvermittelten Kommunikationsnetzen. Die Möglichkeit, sich unerkannt und anonym im Internet zu bewegen, ist mit einer der Hauptgründe der zunehmenden Bedrohungen und Internetkriminalität. Das Hinzufügen von Ortsinformationen auf IP-Ebene durch den IPclip-Mechanismus gibt hingegen Aufschluss über die Herkunft eines IP-Pakets und die physikalische Teilnehmerleitung. Auswirkungen auf verschiedene sicherheitskritische Anwendungen und Fragestellungen werden diskutiert.

Da mittlerweile ein Großteil der Paketverarbeitung aufgrund der steigenden Datenraten in Hardware durchgeführt werden muss, richtet sich die Aufmerksamkeit im zweiten Themenstrang dieser Arbeit auf den Entwurf digitaler integrierter Systeme. Dabei werden interessante

Parallelen zwischen dem aktuellen Entwicklungsprozess Chip-interner Kommunikations- und Verbindungsstrukturen und dem Bereich der makroskopischen Kommunikationsnetze wie dem Internet aufgezeigt und diskutiert. Es werden die Entwicklung einer schlanken und skalierbaren Network-on-Chip-Architektur (NoC) sowie verschiedene funktionale bzw. topologische Optimierungsansätze vorgestellt und bewertet. Mit HSM (Hybrid Switching Mechanism) wird ein effizientes und ressourcensparendes Flusskontrollverfahren für ein mesochrones NoC vorgestellt. Die Border-Enhanced-Mesh-Topologie (BEAM) ist hingegen eine Weiterentwicklung bzw. Modifikation klassischer Gitter-Topologien, welche einerseits enormes Einsparungspotential bzgl. des Hardware-Overheads des NoC-Subsystems bietet und sich andererseits aufgrund kurzer Kommunikationswege günstig auf die System-interne Kommunikation auswirkt.

Die Praktikabilität der Beiträge der Arbeit wird darüber hinaus durch deren Umsetzung in voll funktionsfähigen Hardware-Prototypen unterstrichen.

Abstract

User numbers, data rates, and the diversity of media, protocols, and services in the Internet are escalating. The integration density, the functional spectrum, and the number of parallel processing cores of integrated digital systems are rising similarly. Innovative and scalable concepts are required to face this increasing system complexity and involved problems and questions in the telecommunication area as well as in the design of integrated circuits. For this reason, the terms *complexity* and *scalability* take center stage of many research activities in these areas now. Consequently, efficient ways of communication do play a decisive role—this is true for the off-chip and on-chip domain.

This thesis analyzes the current state in the telecommunication area and derives needs and requirements on prospective communication infrastructures. Those will serve as motivation for the proposed approaches. Thereby, aspects of network security in the Internet, the handling of increasing user numbers, and architectures of integrated systems for packet processing are in the focus of this thesis. The first contribution, MAC Address Translation (MAT), especially aims at security and scalability in the area of Ethernet-based access networks. MAT is a mechanism to convert and replace data link layer addresses. Using a flexibly configurable translations scheme, MAT reduces the absolute number of MAC addresses towards to core networks and thereby the size of the address tables within core network devices. Furthermore, the MAT architecture prevents typical attack scenarios on the data link layer, e. g., spoofing and flooding, to guarantee consistency of the address tables. A different approach to increase network security is IP Calling Line Identification Presentation (IPclip). The IPclip architecture builds a bridge between conventional circuit-switched and modern packet-switched communication networks. One of the most crucial reasons for the soaring threats and cybercrime is the possibility to remain incognito and anonymous while browsing and scanning the World Wide Web. Contrary, the addition of location information on the IP layer by the IPclip-mechanism sheds light on the origin of IP packets and a subscriber's physical access line. The thesis discusses impacts and implications on various safety-critical applications and problem statements.

By now, packet processing is done in hardware for the most part due to continuously rising data rates. This is why the second strand of this thesis brings the design of integrated digital systems into focus. Interesting similarities and commonalities between the current development process of on-chip communication and interconnection networks and the area of macroscopic communication networks like the Internet exist and are pointed out and discussed. The deve-

lopment of a straightforward and scalable Network-on-Chip (NoC) architecture as well as its stepwise refinement on the functional and topological level are presented and evaluated. On the one hand, an efficient and resource-saving flow control scheme for a mesochronous NoC is introduced with HSM (Hybrid Switching Mechanism). On the other hand, the Border-Enhanced-Mesh (BEAM) topology is a further development of classical two-dimensional mesh topologies. BEAM exhibits an enormous savings potential regarding the hardware overhead of the NoC subsystem and benefits the system-internal communication with short communication paths.

Moreover, applicability and feasibility of the thesis' contributions are emphasized with fully functional hardware prototypes using FPGA development boards.