

Untersuchungen zu einer konfigurierbaren Netzwerkemulation auf FPGA-Basis

Belegverteidigung

TU Dresden, 03.07.2015

**Roman Guskov
Guskov.roman@gmail.com**

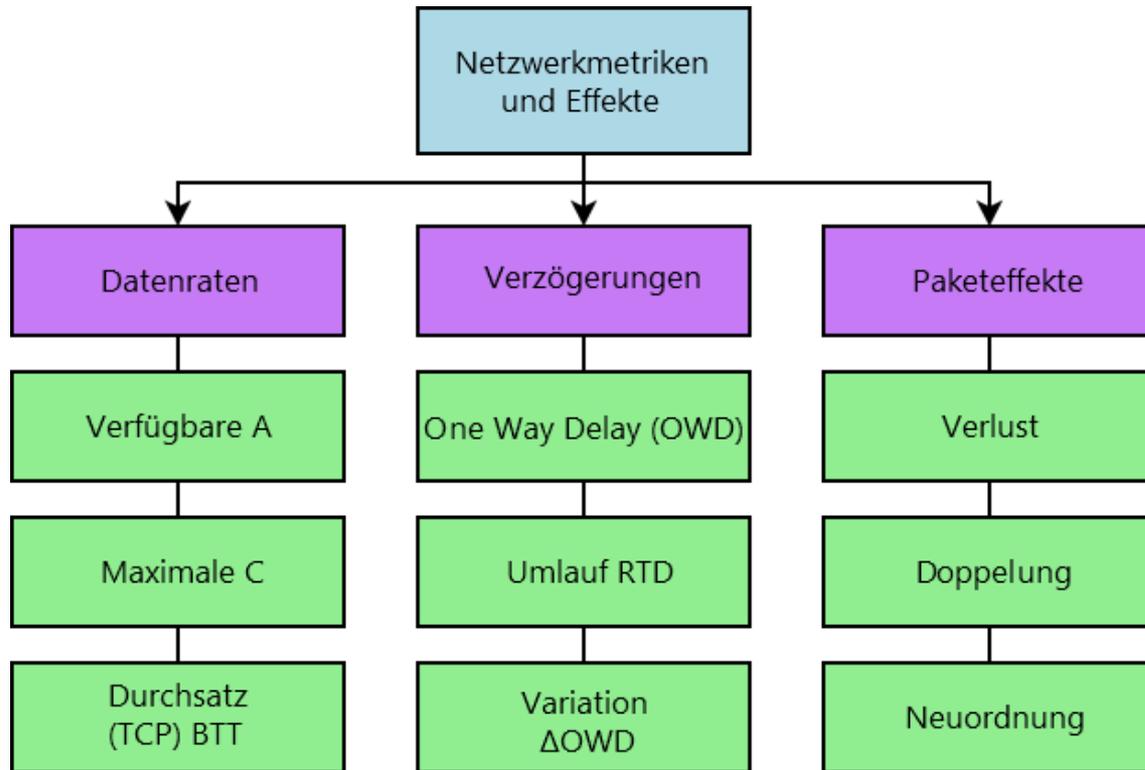
Gliederung

- 1. Einleitung**
- 2. Netzwerkmetriken**
- 3. Entwurf und Implementierung**
- 4. Testumgebung**
- 5. Ergebnisse und Auswertung**
- 6. Einsatzgebiet**
- 7. Zusammenfassung und Ausblick**

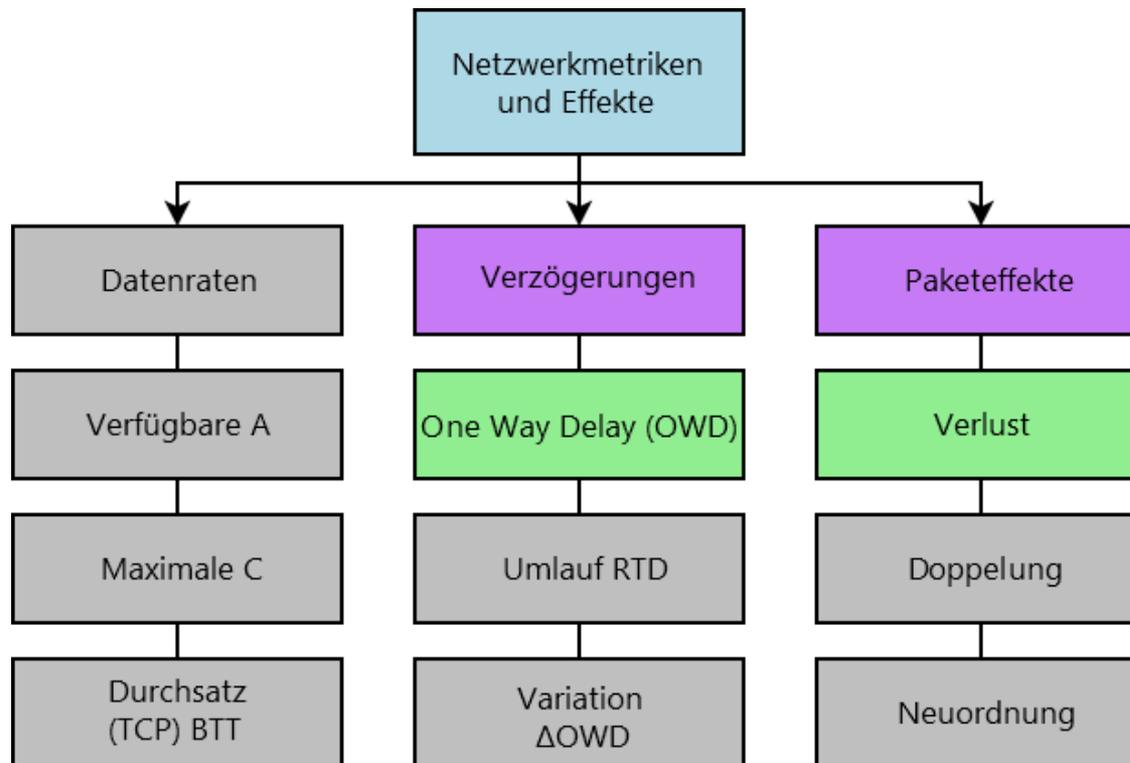
1. Einleitung

- **Unterschied zwischen Emulation und Simulation**
- **Aufgabenstellung:**
 - **Untersuchung der Netzwerkmetriken**
 - **Entwicklung eines Hardware-Netzwerkemulators**
 - **Evaluation des Systems**

2. Netzwerkmetriken – Übersicht



2. Netzwerkmetriken – die Auswahl

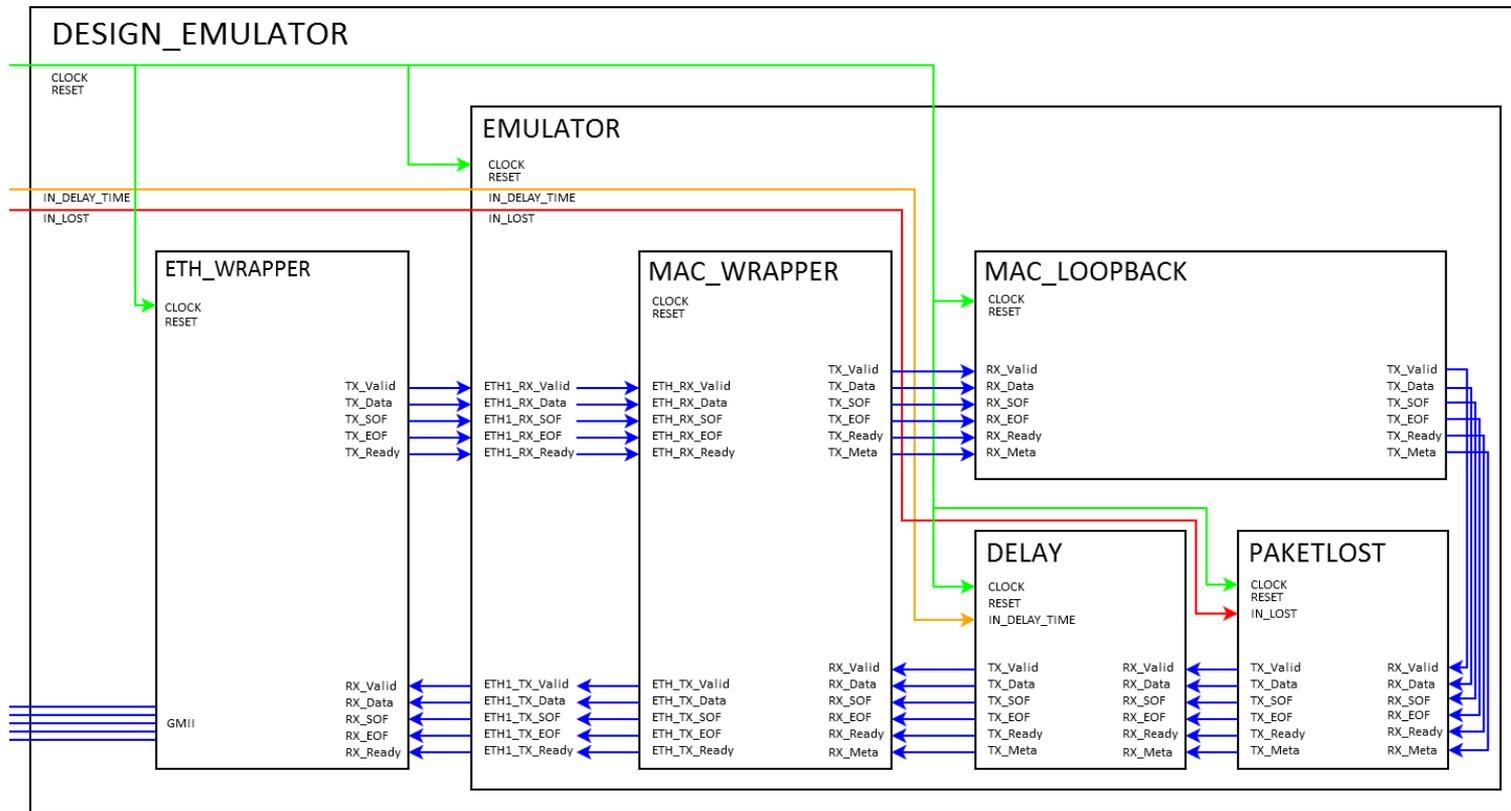


3. Entwurf und Implementierung – Architektur

Bestandteile des Emulators sind:

- **ETH_WRAPPER** – Paketsender sowie -Empfänger,
- **MAC_WRAPPER** – Kodierung und Dekodierung des Ethernet-Frames,
- **MAC_LOOPBACK** – Kurzzeitig in internen Buffer puffert und unverändert ausgibt.
Dabei werden die SRC- und DST-Adressen vertauscht,
- **DELAY** – Emuliert die Verzögerung,
- **PAKETLOST** – Emuliert der Paketverlust

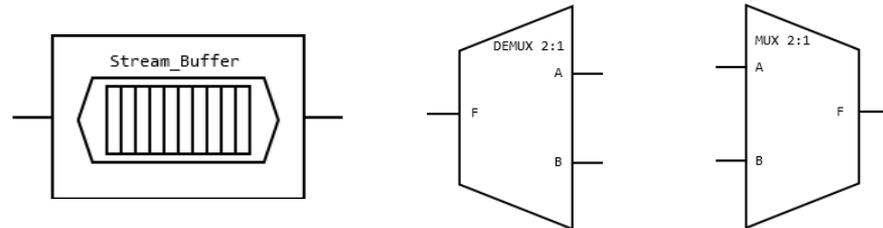
3. Entwurf und Implementierung – Blockdiagramm



3. Entwurf und Implementierung – Grundmodul

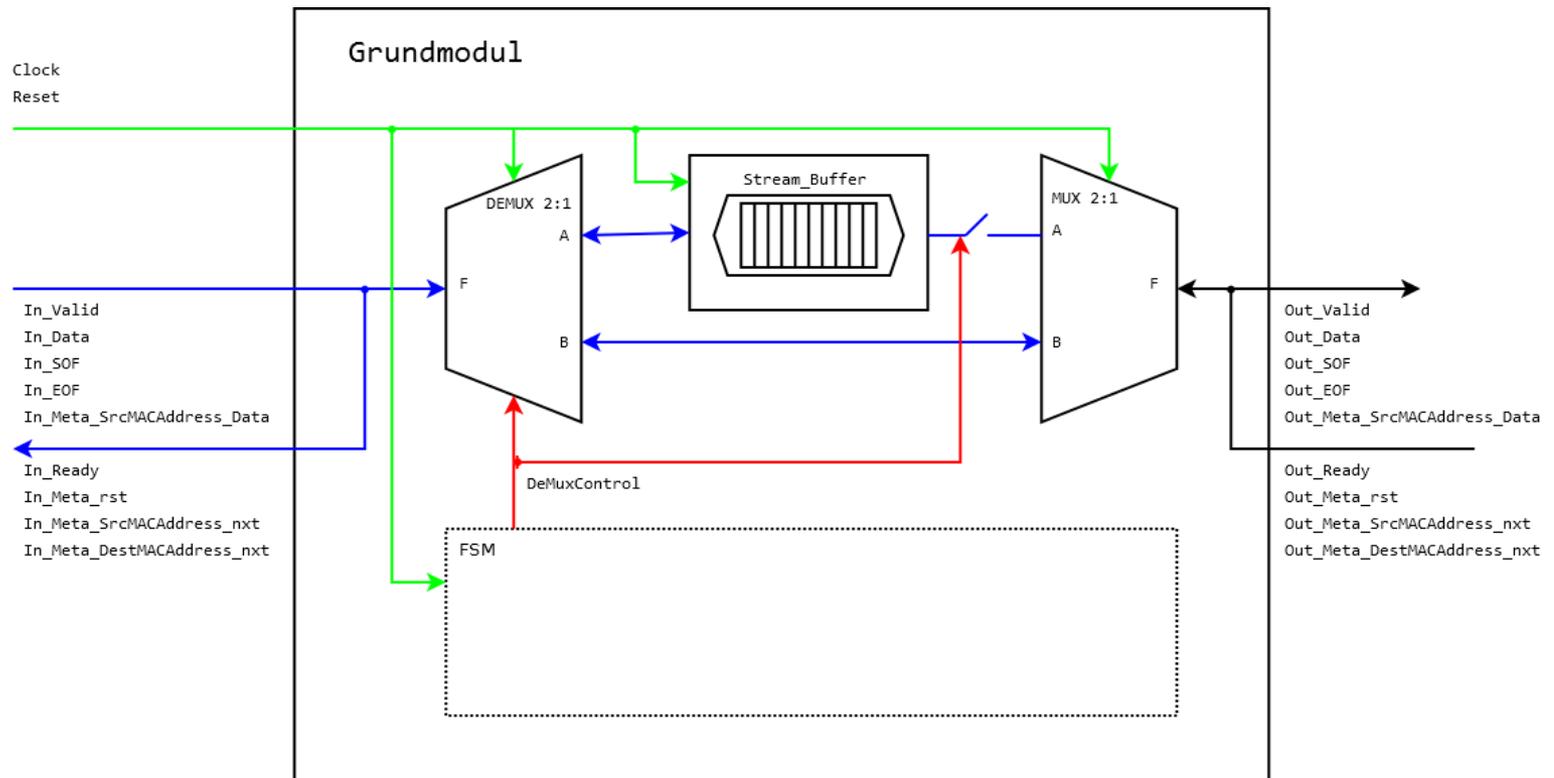
- Bestandteile sind:

- Stream Buffer von 64 KByte
- Multiplexer (2:1)
- Demultiplexer (1:2)



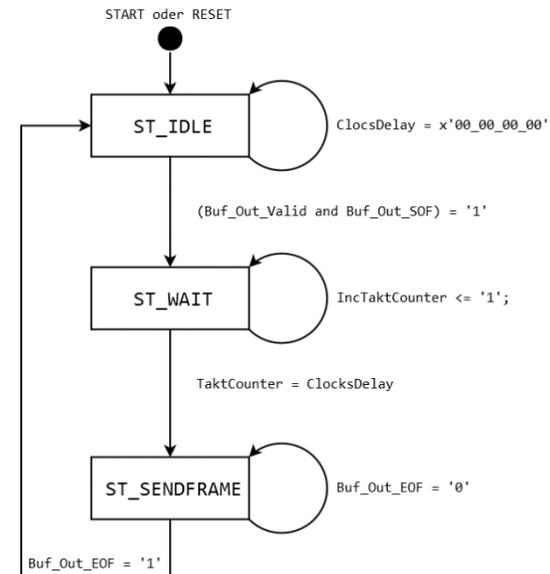
- Grundmodul wird nicht direkt für das Projektes verwendet
- Ziel: Test sowohl der Hauptkomponenten wie Buffer, Multiplizierer und Demultiplexer als auch deren Zusammenschaltung.

3. Entwurf und Implementierung – Grundmodul



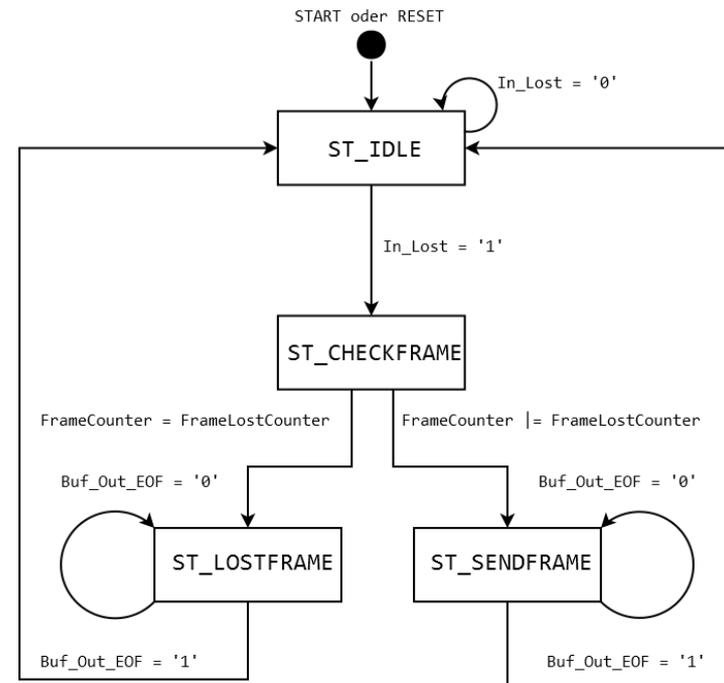
3. Entwurf und Implementierung – Verzögerungsmodul

- Verzögerungsmodul basiert auf Grundmodul und besteht aus Buffer,
- Multiplexer, Demultiplexer
- Minimale Verzögerung – 8 ns, maximale – 1 s
- Im Fall Verzögerung 0,
 - Datenstrom wird durch den Demultiplexer
 - und den Multiplexer geleitet.
- Im Fall Verzögerung > 0 ,
 - Datenstrom wird durch den Puffer geleiten
 - Um bestimmte Zeit zu verzögern



3. Entwurf und Implementierung – Paketverlustmodul

- Paketverlustmodul basiert auf Grundmodul und besteht aus Buffer, Multiplexer, Demultiplexer
- 10% Paketverlust eingestellt
- Im Fall Paketverlust 0,
 - Datenstrom durch den Demultiplexer und den Multiplexer geleitet.
- Im Fall Paketverlust 1,
 - Datenstrom durch den Puffer leiten
 - Jeden 10 Paket wird verloren



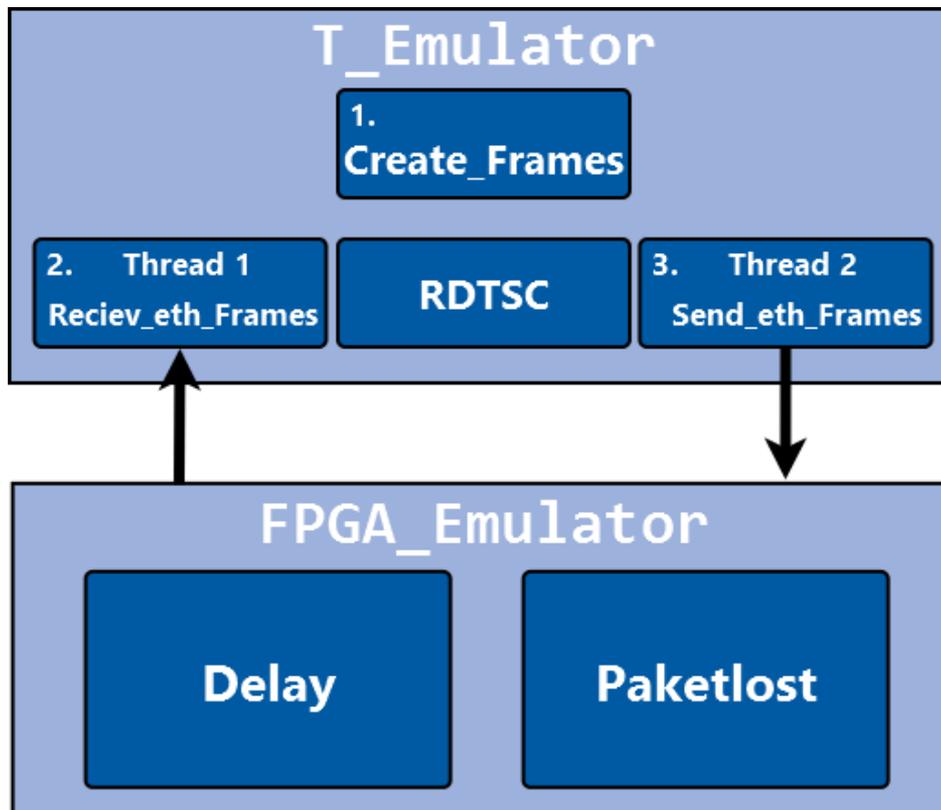
4. Testumgebung – Einleitung

- Aufgabe:
 - Ethernet-Frames zusammenbauen und zum Emulator senden
 - Eine Antwort vom Emulator bekommen und die Verzögerungszeit berechnen
 - Paketverluste erkennen
- Werkzeuge:
 - Ubuntu 14.04
 - Eclipse Luna 4.4
 - GCC 5.1
 - Sprache C++

4. Testumgebung – Zeitberechnung

- **Ziel:** Wie kann möglichst präzise Berechnung der Verzögerung
- **Problem:** Die Standardfunktionen ***gettimeofday()***, ***clock()*** oder ***clock_gettime()*** sind ungenau
- **Grund:** Die Auflösung des Systemtimers
- **Lösung:** Time Stamp Counter (***TSC***) benutzen
 - Anzahl der Zyklen seit dem letzten Zurücksetzen
 - Zuverlässiger Wert beträgt 50 Takten
 - Kalibrierung ist notwendig

4. Testumgebung – Testprogramm

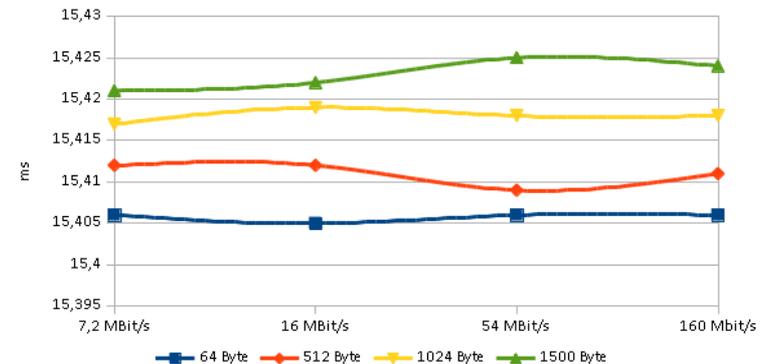


4. Testumgebung – Testablauf

- **4 Datenraten:**
 - HSDPA (3G QPSK, 16-QAM) – 7.2 Mbit/s,
 - ADSL (ANSI T1.413) – 16 Mbit/s,
 - WLAN (802.11g) – 54 Mbit/s,
 - Kabel (DOCIS 3.0) – 160 Mbit/s.
- **4 Paketgröße (64, 512, 1024, 1500 Byte):**
 - 64 Byte – die kleinsten erlaubte Ethernet Paketgröße.
 - Begründung: die Erkennung von Paketkollisionen bei maximaler Netzausdehnung (IEEE 802.2 CSMA/CD (Ethernet))
 - 1500 Byte – die größte erlaubte Ethernet Paketgröße (MTU)
- **1000 Paketen pro Testablauf**
- **3 Verzögerungen:** 15 ms, 100 ms, 1 s.
- **Paketverlust** von 10 %

5. Ergebnisse und Auswertung – Verzögerung

Datenraten	Paketgröße			
	64 Byte	512 Byte	1024 Byte	1500 Byte
7.2 MBit/s	15,406 ms	15,412 ms	15,417 ms	15,421 ms
16 MBit/s	15,405 ms	15,412 ms	15,419 ms	15,422 ms
54 MBit/s	15,406 ms	15,409 ms	15,418 ms	15,425 ms
160 MBit/s	15,406 ms	15,411 ms	15,418 ms	15,424 ms



Die Messwerte bei der Emulation der Verzögerung von 15 ms.

5. Ergebnisse und Auswertung – Paketverlust

```
95. 64 bytes to (8_9E_1_10_CD_7B): ttl=800, time=15 95. 512 bytes to (8_9E_1_10_CD_7B): Request timed out
96. 64 bytes to (8_9E_1_10_CD_7B): ttl=800, time=15 96. 512 bytes to (8_9E_1_10_CD_7B): ttl=800, time=100
97. 64 bytes to (8_9E_1_10_CD_7B): ttl=800, time=15 97. 512 bytes to (8_9E_1_10_CD_7B): ttl=800, time=100
98. 64 bytes to (8_9E_1_10_CD_7B): Request timed out 98. 512 bytes to (8_9E_1_10_CD_7B): ttl=800, time=100
99. 64 bytes to (8_9E_1_10_CD_7B): ttl=800, time=15 99. 512 bytes to (8_9E_1_10_CD_7B): ttl=800, time=100
100. 64 bytes to (8_9E_1_10_CD_7B): ttl=800, time=15 100. 512 bytes to (8_9E_1_10_CD_7B): ttl=800, time=100

/// -----Statistic----- /// /// -----Statistic----- ///

100 Frames transmitted, 90 received, 10% Frames loss 100 Frames transmitted, 90 received, 10% Frames loss
roman@ITE194:~/workspace/T_Emulator/Debug$ roman@ITE194:~/workspace/T_Emulator/Debug$
```

Paketverlust von 10 % bei 7.2 MBit/s

Paketverlust von 10 % bei 54 MBit/s

5. Ergebnisse und Auswertung – Puffergröße

- Der Emulator hat den inneren Puffer von 64 KByte
- Mit der Formell kann die maximale Verzögerung berechnet werden

$$D_{max} = A_p \cdot D_p$$

Paketgröße	Paketanzahl
64 Byte	1000
512 Byte	125
1024 Byte	62
1500 Byte	42

Datenraten	max. Verzögerung
7.2 MBit/s	71 ms
16 MBit/s	32 ms
54 MBit/s	9 ms
160 MBit/s	3 ms

6. Einsatzgebiet

- Cisco CallManager oder Cisco IOS® H.323 gateways

Geschwindigkeit des Codecs	Voice payload sizes (Byte)	Voice payload sizes (ms)	Packets Per Second (PPS)	Bandbreite für FRF.12 (Kbit/s)	Bandbreite für FRF.12 mit cRTP (Kbit/s)	Bandbreite (Ethernet)
G.711 (64 KBps)	160 Bytes	20 ms	50	82.8 KBps	67.6 KBps	87.2 KBps
G.726 (32 KBps)	80 Bytes	20 ms	50	50.8 KBps	35.6 KBps	55.2 KBps
G.726 (24 KBps)	60 Bytes	20 ms	50	42.8 KBps	27.6 KBps	47.2 KBps
G.728 (16 KBps)	60 Bytes	30 ms	33.3	28.5 KBps	18.4 KBps	31.5 KBps
G.729 (8 KBps)	20 Bytes	20 ms	50	26.8 KBps	11.6 KBps	31.2 KBps

Überblick über der Netzwerkcharakteristik für Cisco CallManager und Cisco IOS® H.323 gateways. (Quelle: [cisco])

- **Formel für die Berechnung der nötige Bandbreite pro Anruf**
 - Total packet size = (L2 header: MP or FRF.12 or Ethernet) + (IP/UDP/RTP header) + (voice payload size)
 - PPS = (codec bit rate) / (voice payload size)
 - Nötige Bandbreite = total packet size * PPS

6. Einsatzgebiet

- Der Emulator hat den inneren Puffer von 64 KByte
- Mit der Formell kann man die maximale Verzögerung berechnen

$$D_{max} = A_p \cdot D_p$$

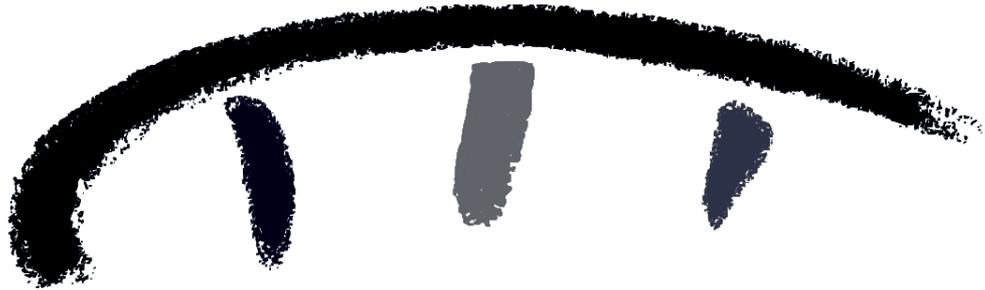
Datenraten	Paketanzahl	max. Verzögerung
G.711 (64 Kbps)	380	7 s
G.726 (32 Kbps)	727	14 s
G.726 (24 Kbps)	941	18 s
G.728 (16 Kbps)	941	28 s
G.729 (8 Kbps)	2285	45 s

7. Zusammenfassung und Ausblick

- **Zusammenfassung**
 - **Untersuchung der erforderlichen Netzwerkmetriken**
 - **Literaturstudium zur Netzwerkemulation**
 - **Entwurf und Implementierung eines Netzwerkemulators**
 - **Analyse und Test der ausgewählten Netzwerkmetriken**
- **Ausblick**
 - **Erweiterung des Puffers durch externen DDR Speicher**
 - **Weitere Netzwerkmetriken nachrüsten**

Quelle

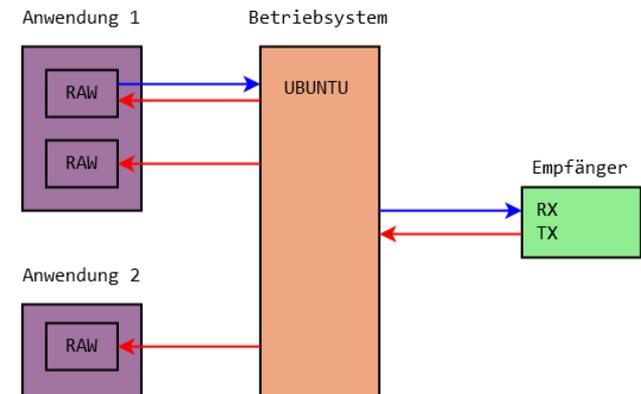
- **[Büss] Büschel, Peter: Diplomarbeit : Entwicklung und Umsetzung einer Methode zur Genauigkeits- und Leistungsanalyse von Netzwerkemulatoren. (2014).**
- **Technische Universität Dresden**
- **[cisco] CISCO Systems - <http://www.cisco.com>**
- **[itu] International Telecommunication Union – <http://www.itu.int>**



»Wissen schafft Brücken.«

8. Anhang – RAW-Sockets

- Zwei Besonderheiten
 - Datenkapselung
 - Zugriff auf die Header eines Pakets
- Empfangen
 - Anwendung bekommt das ganze Datagramm
 - alle Prüfungen auf Korrektheit werden vom Kern ausgeführt
 - Anwendung bekommt auf keinen Fall ungültige Datagramme
- Senden
 - Zusammenbau des Headers mit die spezielle Option `IP_HDRINCL`
 - „Total Length“ und „Header Checksum“ automatisch vom Kern angegeben



8. Anhang - Einsatzgebiet

- ITU Empfehlungen (International Telekommunikation Union)
 - P.800 (MOS, Mean Opinion Score)
 - P.861 (PSQM, Perceptual Speech Quality Measurement)
 - G.114 (OWT, One-way transmission)
- Toleranz zur Verzögerung bis zu 150 ms (max. 400 ms) in einer Richtung
- Jitter – Effekt von 30 ms bis 300 ms
 - Min. (MTU) Paketverlust
 - Bei G.711 (64 Kbit/s) kann bis 10 % Paketen verloren werden
 - Bei G.729 (8 Kbit/s) 1% Paketverlust unangenehm

8. Anhang – Datenraten

- **Datenübertragungsrate** (*engl. Capacity C*) - maximale mögliche IP-Schicht Datenrate
 - Hängt von dem physikalischen Medium des Kanals ab
 - Von der Zeit und aktuellen Datenverkehr unabhängig

$$C = \min_{i=0,\dots,H} C_i$$

- Verfügbare **Bandbreite** (*engl. Bandwidth A*) – der Anteil von der unbenutzten Kapazität C
 - Hängt von dem Hintergrundverkehr (*engl. Cross Traffic (CT)*) ab

$$C = \min_{i=0,\dots,H} (C_i(1 - u_i))$$

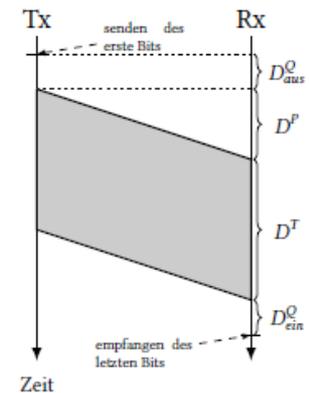
- **Hintergrundverkehr** (*eng. Cross Traffic (CT)*)
 - Oft als Auslastung (*engl. Utilization u_i*) bezeichnet
 - Wert liegt $0 \leq u_i \leq 1$

8. Anhang – Verzögerung

- Die Differenz zwischen dem Sendezeitpunkt T_x und Empfangszeitpunkt R_x eines Paketes.
- Zwei Typen der Verzögerung:
 - **One Way Delay (OWD)** - Verzögerung in eine Richtung,
 - **Round Trip Delay (RTD)** - Umlaufverzögerung.

$$t^{Rx} - t^{Tx} = \begin{cases} D_T^j + D_P^j + D_Q^j, & OWD \\ D_T^j + D_P^j + D_Q^j + D_{T'}^j + D_{P'}^j + D_{Q'}^j, & RTD \end{cases}$$

*



Verzögerungsphasen bei OWD. (Quelle: [Büs])

* Die Einzelverzögerungen bei der Übertragung D_Q^j , Ausbreitung D_T^j und der Warteschlange D_P^j

8. Anhang – Jitter

- Die Schwankungen ΔOWD^j in der Latenzzeit zwischen den einzelnen Paketen

*

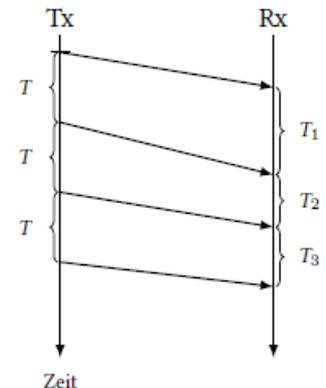
$$\Delta OWD^j = OWD^j - OWD^{j-1} = (D_T^j - D_T^{j-1}) + (D_P^j - D_P^{j-1}) + (D_Q^j - D_Q^{j-1})$$

- Pakete mit der selben Größe über den gleichen Pfad versendet werden

$$\Delta OWD^j = D_Q^j - D_Q^{j-1} \quad *$$

- Somit die D_Q^j ausschlaggebende Größe für **Jitter**

* Die Einzelverzögerungen bei der Übertragung D_Q^j , Ausbreitung D_T^j und der Warteschlange D_P^j



Verzögerungsvariationen. (Quelle: [Büs])

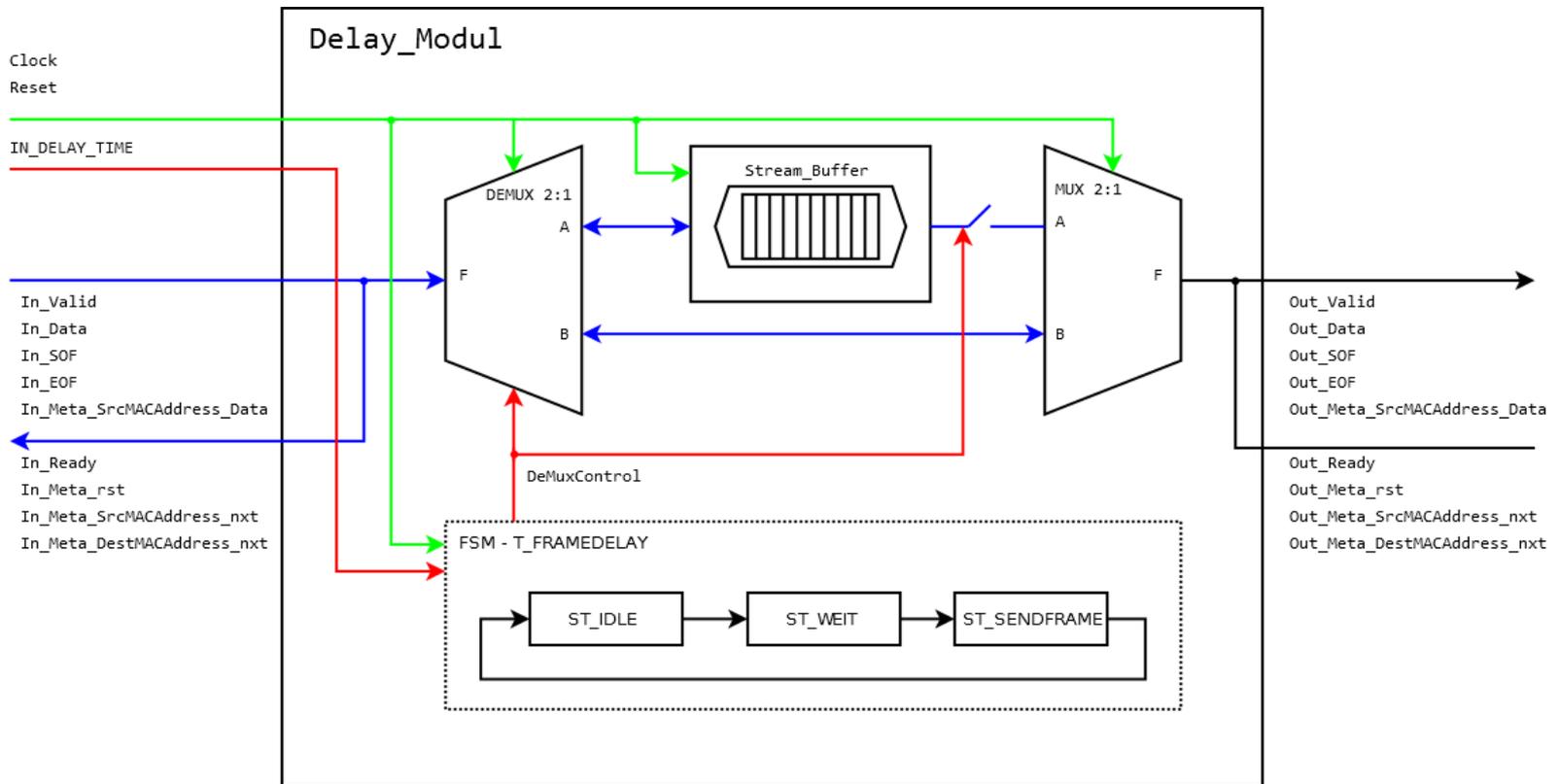
8. Anhang – Paketverlust

- Verschiedene Ursachen für Paketverlust:
 - Fehler des Übertragungsmediums
 - Paketverlust bei der Verarbeitung/(Zwischen-)Speicherung der Pakete
 - Paketverlust aufgrund des Nichteinhaltens von Regeln und Standards
- Paketverlust kann in verschiedenen Schichten des OSI -Modells auftreten
 - Charakteristik : Paketverlustrate (eng. *Paket Loss Rate* **PLR**)

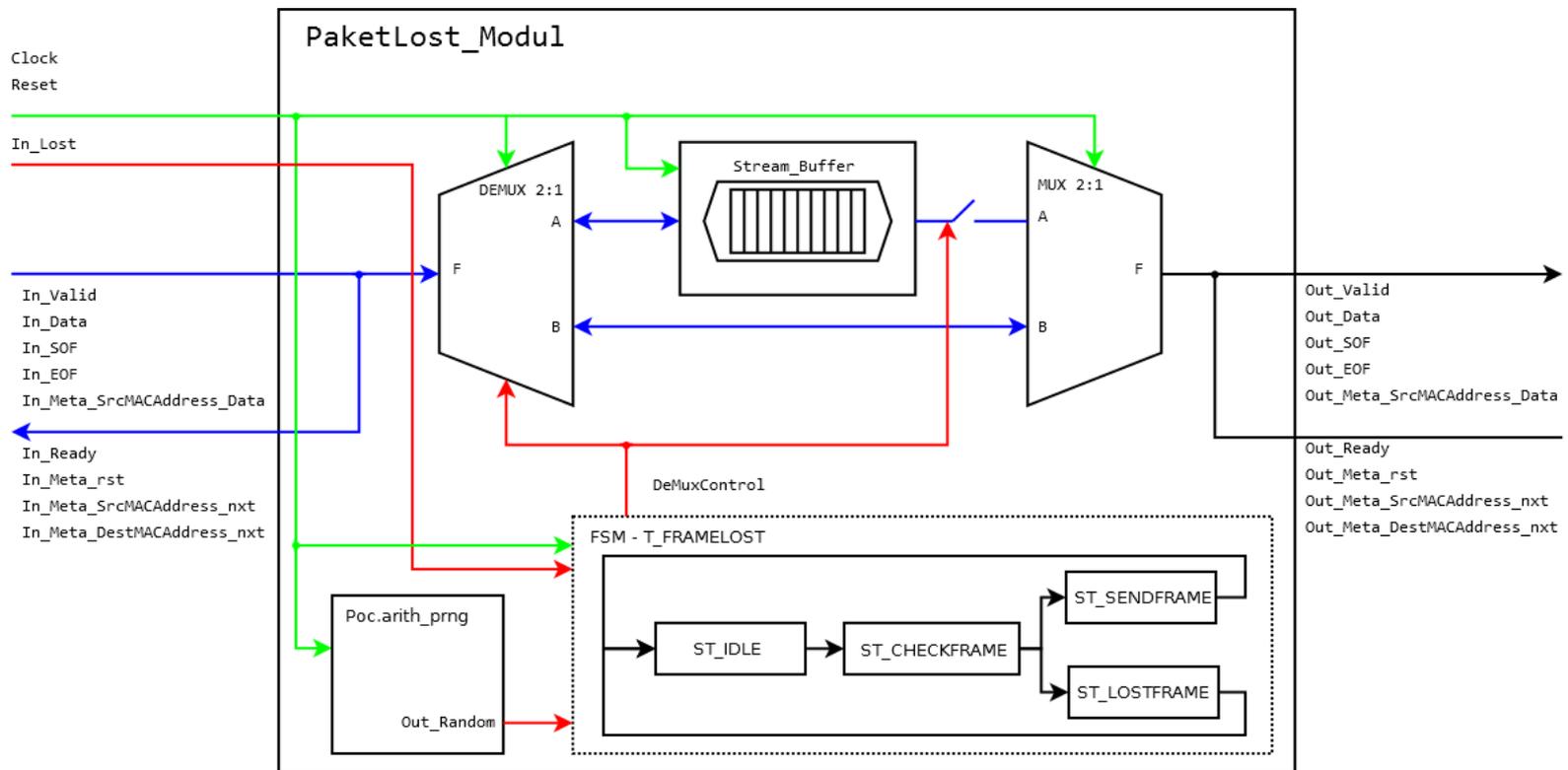
$$PLR = 1 - \frac{M_{Rx}}{M_{Tx}}^*$$

* die Menge der empfangenen Pakete M_{Rx} die Menge der gesendeten Pakete M_{Tx}

8. Anhang – Verzögerungsmodul

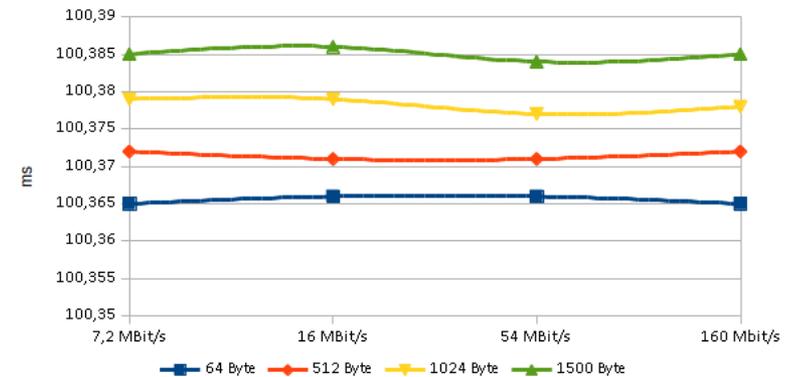


8. Anhang– Paketverlustmodul



8. Anhang – Verzögerung

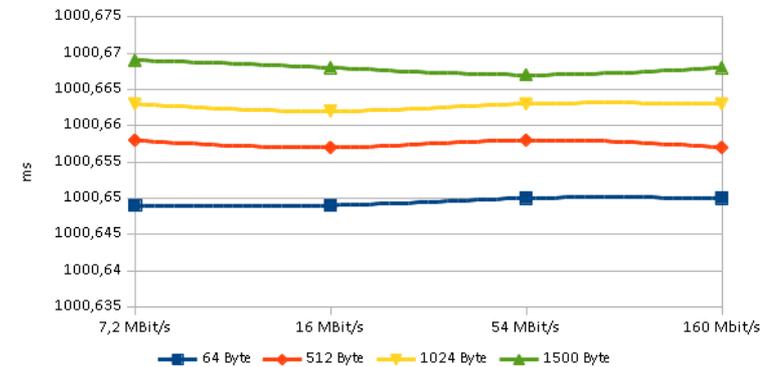
Datenraten	Paketgröße			
	64 Byte	512 Byte	1024 Byte	1500 Byte
7.2 MBit/s	100,385 ms	100,372 ms	100,379 ms	100,385 ms
16 MBit/s	100,366 ms	100,371 ms	100,378 ms	100,386 ms
54 MBit/s	100,365 ms	100,371 ms	100,377 ms	100,384 ms
160 MBit/s	100,367 ms	100,372 ms	100,378 ms	100,385 ms



Die Messwerte bei der Emulation der Verzögerung von 100 ms.

8. Anhang – Verzögerung

Datenraten	Paketgröße			
	64 Byte	512 Byte	1024 Byte	1500 Byte
7.2 MBit/s	1000,649 ms	1000,658 ms	1000,663 ms	1000,679 ms
16 MBit/s	1000,649 ms	1000,657 ms	1000,662 ms	1000,682 ms
54 MBit/s	1000,650 ms	1000,658 ms	1000,662 ms	1000,672 ms
160 MBit/s	1000,667 ms	1000,657 ms	1000,663 ms	1000,672 ms



Die Messwerte bei der Emulation der Verzögerung von 1000 ms.