HyperTransport 3.0

Seminar HWS06

Lehrstuhl für Rechnerarchitektur

Christian Leibig

Übersicht

1. Einführung in HyperTransport
2. Technische Details
3. Neue Features in HyperTransport 3.0

Einführung in HyperTransport

[Abschnitt 1]

1. Was ist HyperTransport?

- Low-latency Chip-to-Chip Interconnect
  - Ein „Bus“ für viele verschiedene Anwendungen
  - HyperTransport Consortium
- Bidirektionale Verbindung, geeignet als
  - Front-Side Bus
  - Multiprocessor interconnect (AMD NUMA)
  - Switch-Bus
  - nahezu beliebige Verbindung (z.B. HTX)

2. Warum HyperTransport?

- keine Lizenzkosten pro Gerät
- geringe Latenz, hohe Bandbreite
- skaliert mit Anforderungen des Einsatzgebiets
- direkte Anbindung an die CPU
- universell anwendbar (CPU, I/O, Peripherie)
- geringer Stromverbrauch
- transparent für PCI, PCI-X und PCI Express
- abwärtskompatibel zu älteren HT-Versionen

3. Wo HyperTransport?

- AMD AMD64 technology CPUs
  - Athlon64, Opteron [HT800/HT1000]
- ATI Radeon Xpress 200 (for AMD)
  - AMD Athlon64 PCIe chipset with HT support
- NVIDIA media communication processors
  - 2001: nForce 2 MCP „Southbridge“ [HT800]
  - 2006: nForce 5xx/6xx SLI MCP [„HT1000“]
- Apple PowerPC G5 Systems
  - 400-800MHz HyperTransport Tunnel

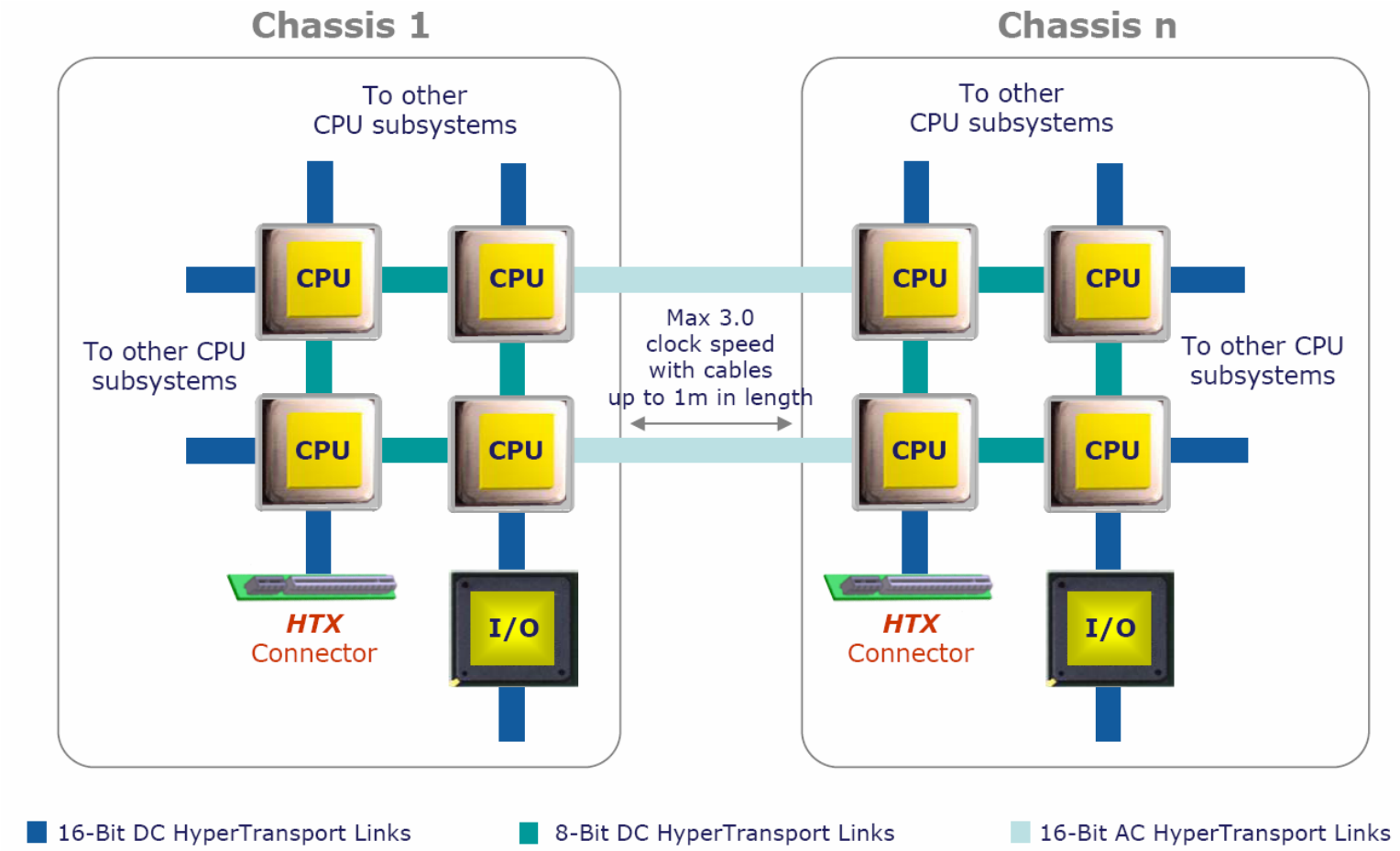
4. Entwicklung von HT [1/2]

- 2001: HyperTransport 1.0
  - 800MHz, 32-bit, 12.8 GB/s (aggregate)
  - DC Operation, LVDS
- 2002: HyperTransport 1.1
  - User Packet Handling
  - Peer-to-Peer Routing
  - Virtual Channels
  - Robust Error Retry Protocol
- 2003: ~30M HT Ports

4. Entwicklung von HT [2/2]

- 2004: HyperTransport 2.0
  - 1.0GHz - 1.4GHz, 22.4 GB/s (aggregate)
  - PCI Express Mapping
  - Backwards Compatibility (auto-detect)
- 2005: HyperTransport Expansion (HTX)
  - 800MHz, 16-bit HyperTransport connector
- 2006: geschätzte 200 Millionen HT Ports
- **Heute: HyperTransport 3.0**

5. Beispiel HT3.0 Netzwerk



Quelle: [3]

Technische Details

[Abschnitt 2]

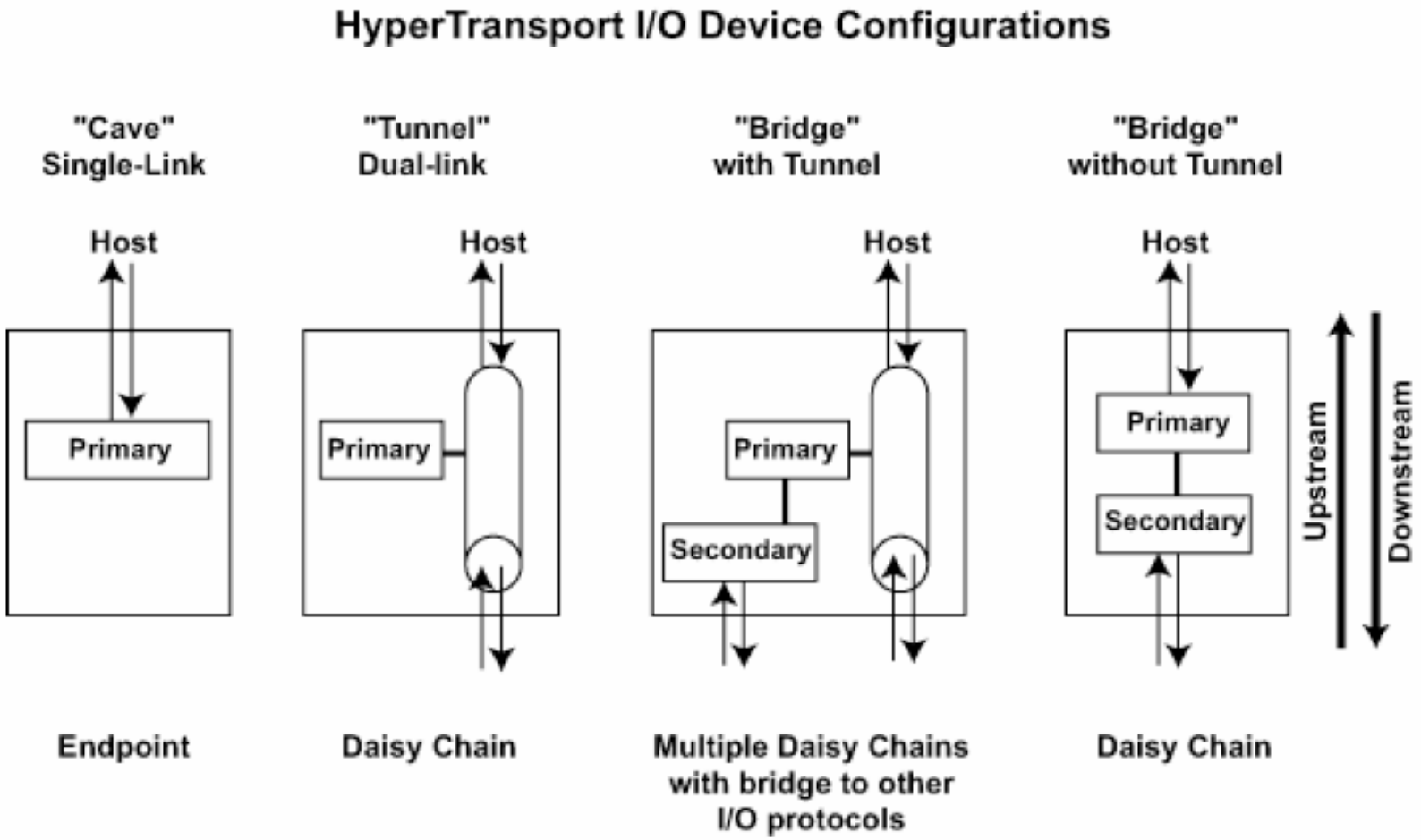
1. Technische Informationen

- Bidirektionale Punkt-zu-Punkt Verbindungen
- Low Voltage Differential Signaling (LVDS)
- Taktrate: 200MHz - 1.4GHz DDR
- Linkbreite: 2 - 32 bit
- Unterstützung für asymmetrische Links
- Paketbasiert (32-bit Wörter)
- posted und non-posted writes
- bis zu 16 virtuelle Kanäle

2. Definitionen

Chain	Eine Gruppe von HT Geräten die in einer Reihe verbunden sind
Cave	Single-Link Interface (Ende einer Chain)
Tunnel	Dual-Link Interface, kann Pakete durchleiten
Bridge	Eine Verbindung zwischen dem primären HT-Bus und einem oder mehreren sekundären Bussen
Host	Ein HT-Host kann mehrere Verbindungen (bridges) zu mehreren chains besitzen
Slave	Tunnel oder Cave ohne Host bridges
Gen1/Gen3	compliancy with HT2.0- / HT3.0+

2. Caves, Tunnels, Bridges…



Quelle: [2]

3. Link Layout [1/2]

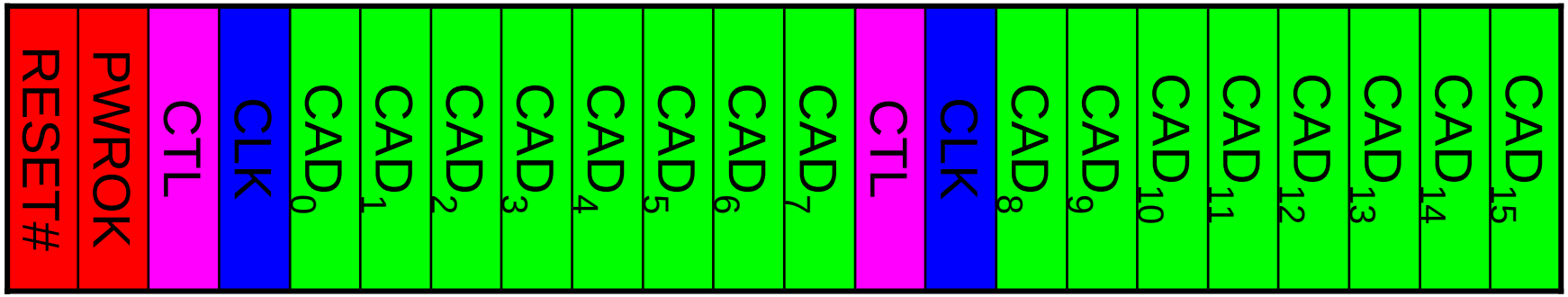
- Shared Signals
  - PWROK: Power und Clock systemweit stabil
  - RESET#: Reset des HyperTransport busses
- Lane Dedicated Signals (HT3.0: every 8-bit)
  - CTL: Signalisiert ein Kommando auf der Leitung
  - CLK: Clock für CAD und CTL
- External Signals (optional)
  - LDTSTOP#: stoppt die Übertragung auf einem Link
  - LDTREQ#: deaktiviert LDTSTOP# falls der schlafende Link von einem Gerät in der Chain benötigt wird

3. Link Layout [2/2]

- **CAD:** Command, Address, Data
- **CTL:** Control Signal
- **CLK:** Clock

HT3.0	Signal Width (bit)				
CAD	2	4	8	16	32
CTL	1	1	1	2	4
CLK	1	1	1	2	4

Beispiel: 16-bit Link Layout (HT3.0)



4. Data Eye

- Zeit in der Daten als gültig erkannt werden können
- wird verkleinert durch Jitter zwischen CLK und CAD/CTL

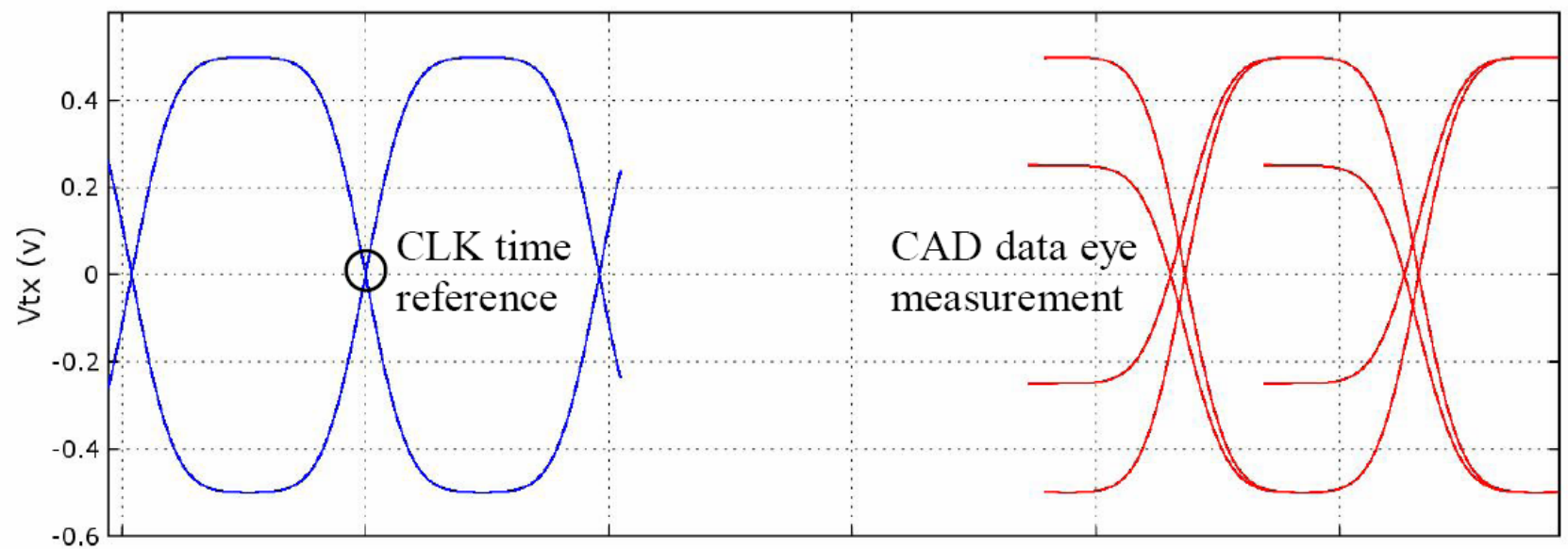


Figure 53. CLK to CAD ideal eye with -6dB of de-emphasis

Quelle: [2]

4. Data Eye

- Zeit in der Daten als gültig erkannt werden können
- wird verkleinert durch Jitter zwischen CLK und CAD/CTL

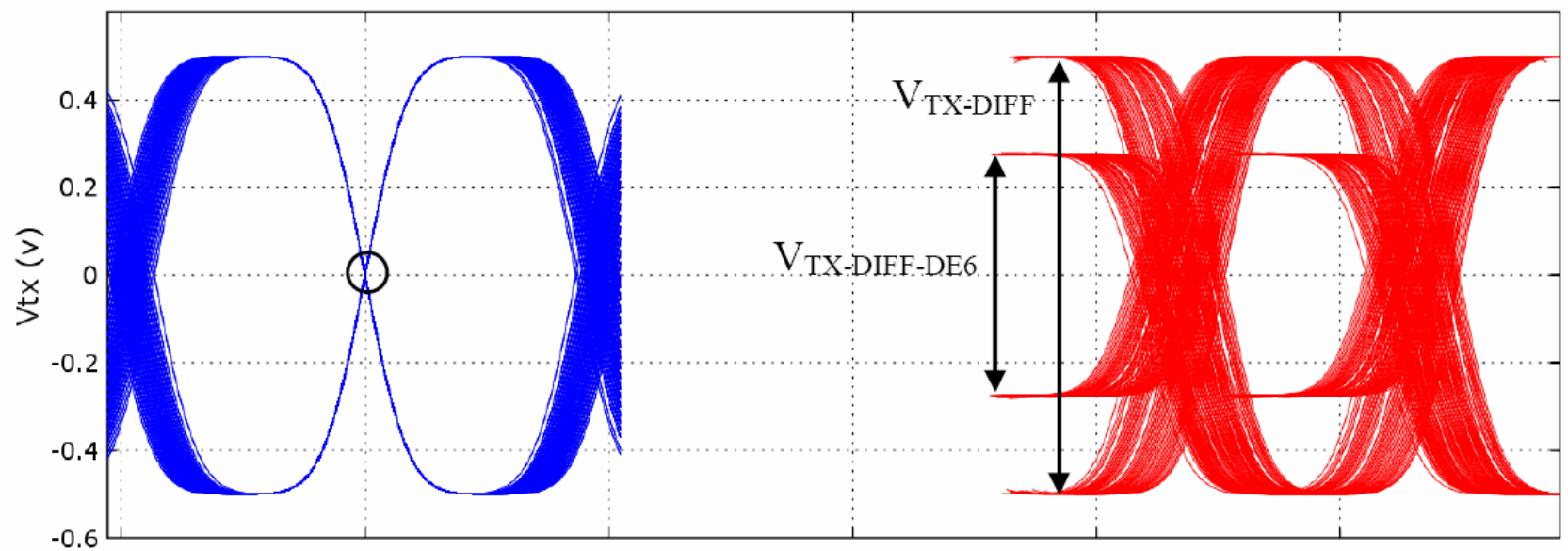


Figure 54. CLK to CAD data eye with some transmit jitter

Quelle: [2]

5. Packet Format [1/3]

- 2 Typen: control und data packets
- Jedes Paket besteht aus einem Set von 32-bit Wörtern und beginnt mit einem command word
- Adressübertragung in control packets:
  - die letzten 8 bit vom command word werden mit den folgenden 32-bit zusammengelegt und bilden eine 40-bit Adresse
  - Falls 64-bit Adressen benötigt werden kann ein weiteres 32-bit control packet vorgeschoben werden
- die restlichen 32-bit Wörter im Paket sind data payload
- Die Länge von data packets ist immer ein vielfaches von 32 bytes, unabhängig davon wieviele Daten sie tatsächlich enthalten

5. Packet Format [2/3]

- Beispiel 1: Request Packet

	7	6	5	4	3	2	1	0
0	**SeqID[3:2]**		**Cmd[5:0]**					
1	**PassPW**	**SeqID[1:0]**		**UnitID[4:0]**				
2	***Command-Specific***							
3	***Command-Specific***							
4	**Addr[15:8]**							
5	**Addr[23:16]**							
6	**Addr[31:24]**							
7	**Addr[39:32]**							

5. Packet Format [3/3]

- Beispiel 2: Request Packet (Extended Address)

	7	6	5	4	3	2	1	0
0	00b		Cmd[5:0]=111110b					
1	Addr[47:40]							
2	Addr[55:48]							
3	Addr[63:56]							
4	SeqID[3:2]		Cmd[5:0]					
5	PassPW	SeqID[1:0]		UnitID[4:0]				
6	*Command-Specific*							
7	*Command-Specific*							
8	Addr[15:8]							
9	Addr[23:16]							
10	Addr[31:24]							
11	Addr[39:32]							

6. CRC Protocol

- periodic CRC computed over 512 bit-times
- each new CRC value is stuffed onto the CAD bits of the link 64 bit-times after the end of the 512-bit-time window and occupies the link for 4 bit-times

CRC Window	Number of Bit-Times	Link Content
1	512	payload: first window
2	64	payload: second window
	4	CRC: first window
	448	payload: second window
3	64	payload: third window
	4	CRC: second window
	448	payload: third window

7. „Gen1“ Link Initialization

- Cold Reset (PWROK, RESET#)
  - CAD[15:0] = 1; CTL, CLK = 0;
- Begin initialization:
  - Takt: 200MHz
  - Breite: 8 bit (maximum)
  - Synchronisierung von transmit und receive clock
  - Zeitpunkt für den Beginn des Pakettransfers setzen
- Link Konfiguration durch die system firmware:
  - Einstellen der maximal unterstützten Linkbreite
  - Einstellen der maximal unterstützten Linkfrequenz
- Warm Reset (RESET#)

8. Bandbreite

- Clock Rate: 200MHz - 1.4GHz DDR
- Link width: 2 - 32 bit (typisch: 16-bit)

- Durchsatz pro Richtung: 2 * [freq] * [width]
  - Faktor 2 wegen DDR
  - Beispielrechnung HyperTransport 2.0:
    2 * 1400MHz * 16bit = 5.6 GB/s

- Bidirektionale Verbindung
  - Faktor 2 für „Aggregate Bandwidth“

HyperTransport 3.0

- specification release date: 21. April 2006
- HT3.0 Dokumentation ist 428 Seiten stark
  - Im Vergleich, HT2.00b: 325 Seiten
- Warum HyperTransport 3.0?
  - Zitat: „Server and High-Performance Workstation Companies Regard HyperTransport 3.0 as Painless Path to Performance Doubling of 16-Bit Link Designs without Added System Design and PCB Real Estate Complexity”

Neue Features in HyperTransport 3.0

[Abschnitt 3]

Neue Features in HT3.0

- Erhöhte Bandbreite
- Fehlerkorrektur auf Paketebene
- Zusätzliche DC-Link Features
- AC Operation Mode (optional, auto-config)
- Training Patterns
- Link-Splitting
- Hot-Plugging
- Enhanced Power Management (optional)

- Trotzdem: Kompatibel zum „alten HT“

1. Erhöhte Bandbreite

- Höhere Taktrate
  - HT 2.0: bis 1.4 GHz
  - HT 3.0: 1.8 bis 2.6 GHz
- Durchsatz pro Richtung: 2 * [freq] * [width]
  - Beispielrechnung HyperTransport 3.0:
    2 * 2600MHz * 16bit = 10.4 GB/s
  - max. aggregate bandwith: 41.6 GB/s
- Relativ einfache Umsetzung, da
  - keine zusätzlichen Leitungen
  - keine zusätzliche Logik im Chip

2. CRC Protocol Changes

- Vorher:
  - periodischer CRC berechnet über 512 bitTimes
- HyperTransport 3.0 (optional):
  - 32-bit CRC angehängt an jedes Paket
  - wird über das gesamte Paket berechnet, inklusive control header
  - benutzt den selben Algorithmus wie der periodische CRC
  - bei Benutzung von packet based CRC wird der periodische CRC deaktiviert

3. Zusätzliche DC Features

- Transmitter
  - Training Pattern für Multi-Bit Skew
  - Support für Scrambling
  - Support für Tx Equalization
  - Retained Clock Forwarding Scheme on Dedicated Lanes

- Receiver
  - Support für Rx Equalization
  - Support für Multi-Bit Skew (Clock based Rx Phase Alignment)

3.1 Scrambling

- Scrambling mit 23bit LFSR: $x^{23}+x^{18}+1$
  - Minimierung von EMI und Übersprechen (crosstalk)
  - Benachbarte CTL and CAD lanes benutzen unterschiedliche scrambling patterns
  - Alternative scrambling patterns werden gewonnen durch XOR unterschiedlicher Abgriffe eines einzelnen LFSR

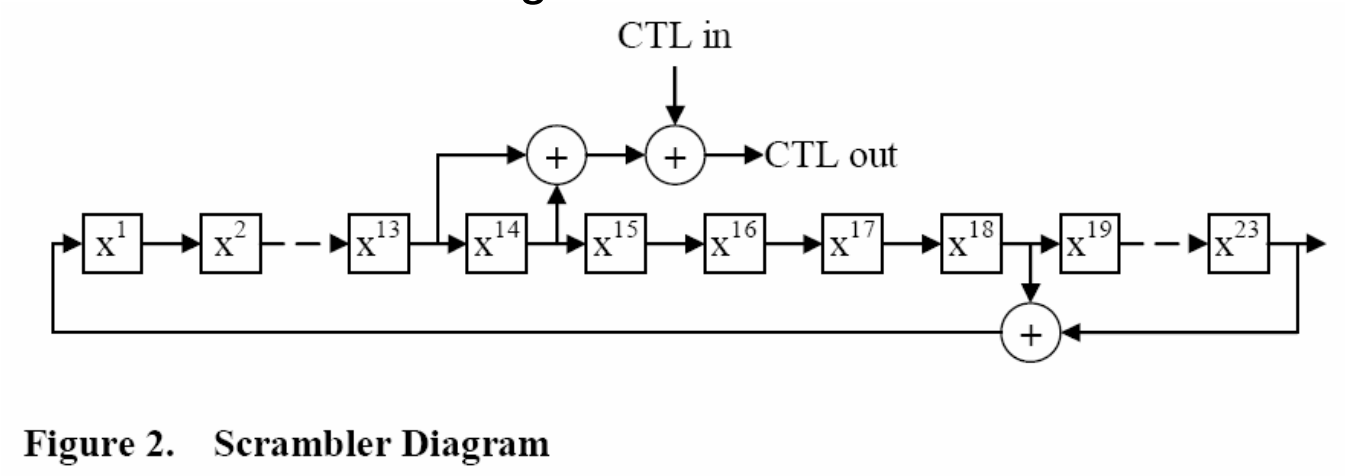


Figure 2. Scrambler Diagram

Quelle: [2]

3.2 Tx Equalization

- Problem: verzögerte Pegelwechsel bei hochfrequenten Signalen
- Lösung: schnellere Pegelwechsel durch vorher reduzierte Pegel
- post-cursor de-emphasis: -3dB / -6dB / -8dB

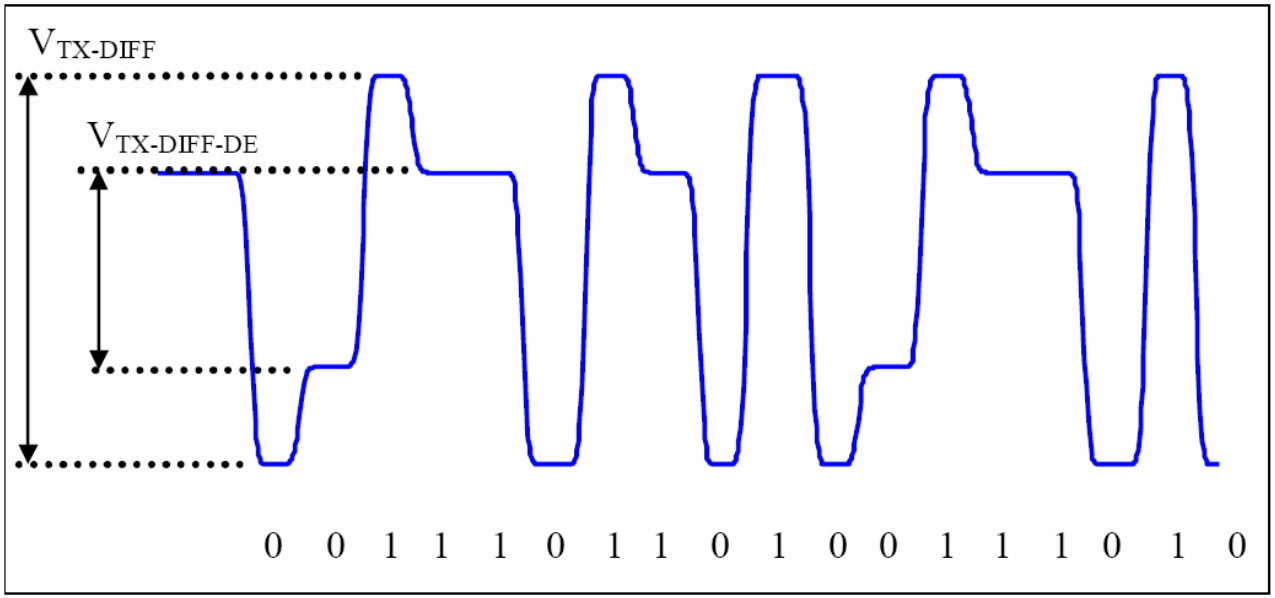
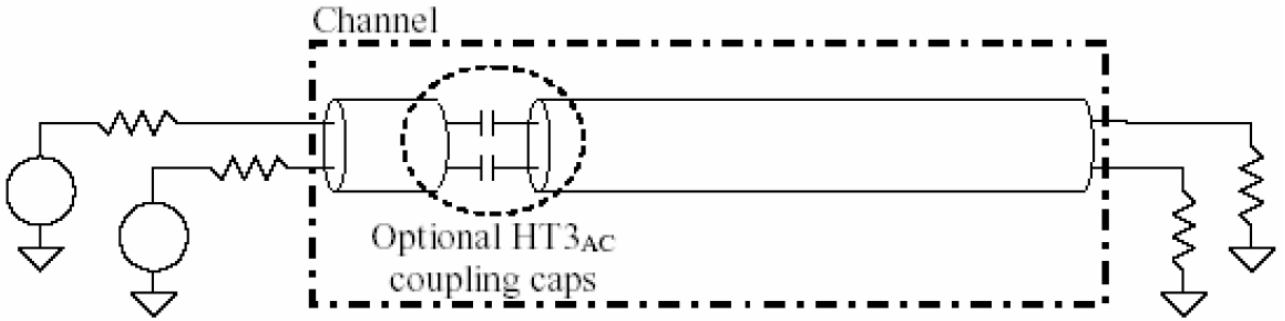


Figure 50. Differential transmitter post-cursor de-emphasized waveform

Quelle: [2]

4. AC Operation Mode [1/3]

- AC-Coupling durch Kapazitäten am Transmitter-Ende der Leitung



- Automatische Erkennung und Konfiguration
- 8b10b Encoding (verhindert DC Signal, erhält den Takt)
- Tx Equalization (pre- und post-cursor)
- minimaler Linktakt: 1200MHz
- maximale Linkbreite: 16bit

Quelle: [3]

4. AC Operation Mode [2/3]

- Höhere Anforderungen an die Tx Equalization
- post-cursor de-emphasis: -3dB / -6dB / -8dB / -11dB
- zusätzlich: pre-cursor de-emphasis: -8dB

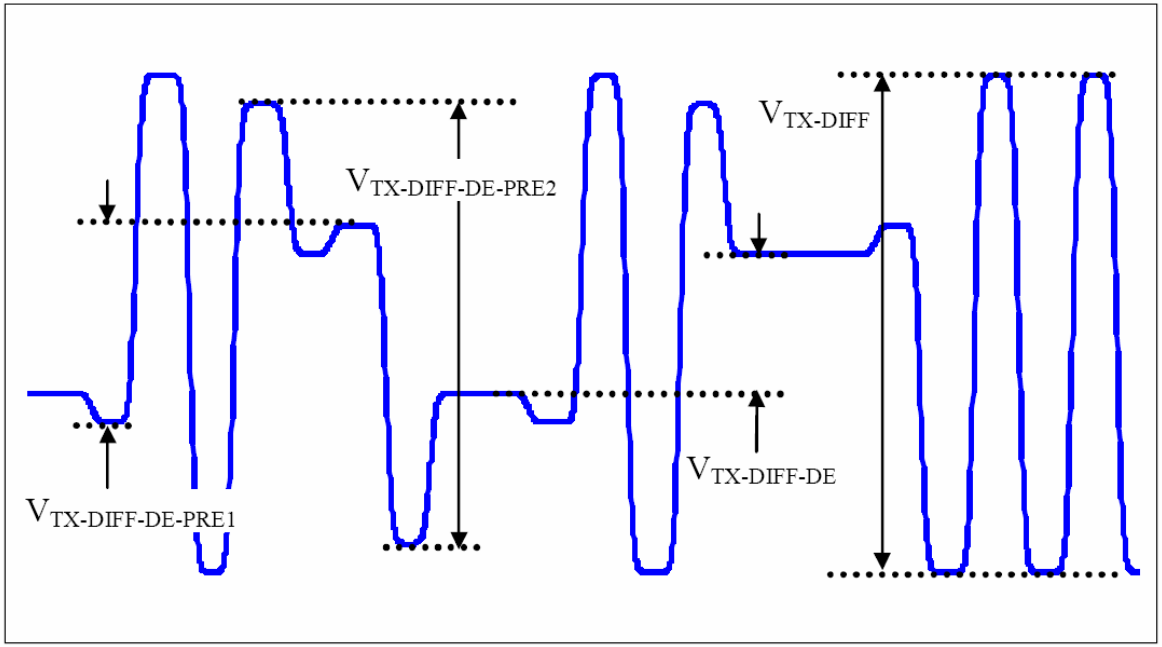


Figure 51. Transmit with pre- and post-cursor de-emphasis

Quelle: [2]

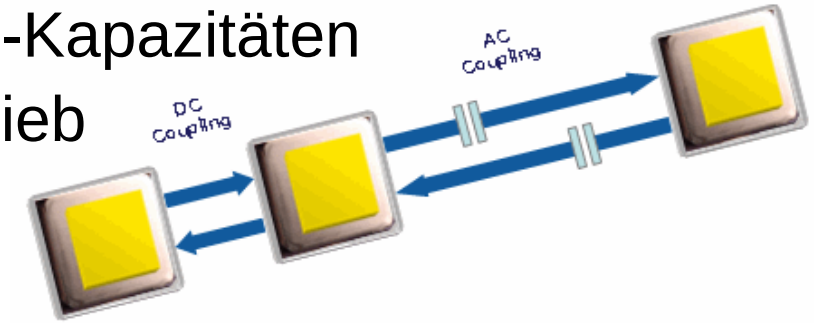
4. AC Operation Mode [3/3]

- Vorteil:
  - Lange Verbindungen möglich:
    Backplanes, Board-to-Board, Chassis-to-Chassis
  - Dadurch breiteres Anwendungsfeld für HT

- Nachteile im Vergleich zum DC-Mode:
  - Geringere Bandbreite (8b10b-Codierung)
  - Höhere Latenz (Scrambling, 8b10b, Kapazitäten)

5. DC/AC Auto-Configuration

- Low Latency DC vs. Long-Reach AC
- Schaltung…
  - prüft Leitung auf Koppel-Kapazitäten
  - schaltet um auf AC-Betrieb



- Vorteil:
  Jeder auf AC vorbereitete Chip kann ohne Änderungen am Silizium an DC oder AC Leitungen angeschlossen werden.

Quelle: [3]

5.1. „Gen 3“ Link Initialization

- PWROK
  - CAD[15:0] = 1; CTL, CLK = 0;
    - Begin initialization, Clock: 200MHz
- DC Detect Phase 1 and 2
  - Check receiver CTL_L and CTL_H levels
    - LVDS → „Gen1“ Initialization
    - both low → AC Detect
- AC Detect
  - PLL ramping for a 1200MHz link
  - TXACDETECT on CAD_H[0] only
  - Link detected: CTL/CAD TXIDLE; CLK TXL0
- Warm Reset

5.2 Training Patterns

- Training Patterns (Übersicht)

TP0	0011111010 0001101011 0001011011 1001010100
TP1	0011111010 0001101011 1100010100 1110010100
TP2	1100000101 1100011011 0001101011 1100101001
TP3	0011111010 1010010100 0101011011 0010010110
TM4	1101101000 1101101000 1101101000 1101101000

5.2 Training Patterns

- Training Patterns (mit Bedeutung)

TP0	0011111010 0001101011 0001011011 1001010100 **lock designated clock, seek data eye**	**Hamming Distanz**
TP1	0011111010 0001101011 1100010100 1110010100 **lock on data eye, initialize 8b10b decoder**	**10 bits**
TP2	1100000101 1100011011 0001101011 1100101001 **maintain lock, test connection stability**	**30 bits**
TP3	0011111010 1010010100 0101011011 0010010110 **initialize receive FIFO, start FIFO read pointers**	**28 bits**
TM4	1101101000 1101101000 1101101000 1101101000 **Operational**	**26 bits**

5.3 Training Patterns

- Training Patterns wurden so gewählt, dass…
  - …sie DC-balanciert sind
  - …sie ausreichend häufige Pegelwechsel aufweisen
  - …die Hamming Distanz zwischen ihnen ausreichend groß ist
- Training Patterns sind identisch in DC und AC-Mode
- Training Patterns werden auf allen aktiven CAD und CTL Leitungen gesendet
- TP3 (initialize FIFO) und TM4 (Operational) sind so gewählt, dass der 8b10b decoder angeworfen wird sobald der Empfänger den FIFO write pointer initialisiert hat

5.3 TP0: lock designated clock

- After Warm Reset, short retry failure or recovery from a disconnect
- Training pattern 0 (TP0) is sent on CTL and CAD lanes to provide a unique pattern for the receive phase recovery mechanism to track

Transmitter	Receiver
- TP0 on CTL and CAD - CLK at programmed frequency	- DLL locking on CLK - Phase Recovery mechanism seeking data eye

5.3 TP1: lock on data eye

- After Completion of TP0 or after a short disconnect
- Training pattern 1 (TP1) is sent on CTL and CAD lanes to begin the training handshake

Transmitter	Receiver
- TP1 sent on CTL and CAD lanes - CLK running at programmed frequency	- Phase recovery mechanism locking on data eye - 8b10b decoder seeking symbol alignment

5.3 TP2: connection stability

- **Receiver:** - has seen *TP1 or TP2* 8 times in a row without error
- **Transmitter:** - all **receiver** lanes have transitioned to Training 2
  - go to Training 2 at the end of a TP1 transmission.

- Training pattern 2 (TP2) is sent on CTL and CAD lanes to complete the training handshake.

Transmitter	Receiver
- TP2 sent on CTL and CAD lanes - CLK running at programmed frequency	- Phase recovery mechanism maintaining lock on data eye - 8b10b decoder symbol alignment locked

5.3 TP3: initialize receive FIFO

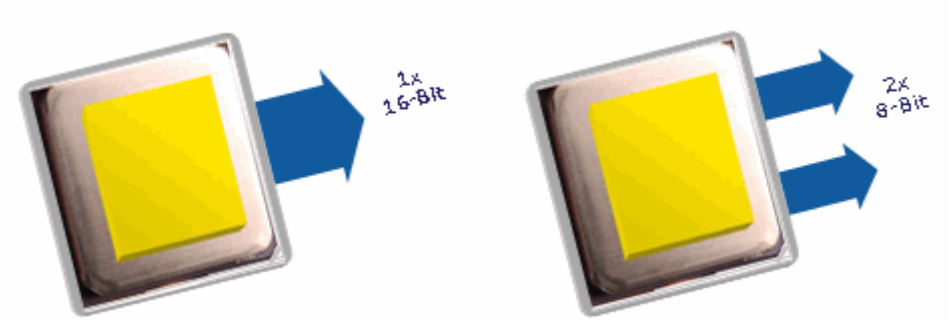
- **Transmitter:** - Wait for one complete TP2 to be received on any lane
  - Send 8 more TP2s, then go to *Training 3*
- **Receiver:** - Each lane waits for one TP2 to be received, then:
  - if anything but TP2 is received, start FIFO write pointer
  - lane goes to *Training 3*

- Training pattern 3 (TP3) is sent on CTL and CAD lanes
  - maintain phase recovery lock
  - initialize the receive FIFO, provide time to start FIFO read pointers.

Transmitter	Receiver
- TP3 sent on CTL and CAD lanes - CLK running at programmed frequency	- Phase recovery mechanism maintaining lock on data eye

6. Link-Splitting [1/2]

- Link-Splitting (auch Un-Ganging):
  Aufteilung eines breiten HyperTransport Links in zwei logische, einzelne Links
- z.B. *1*16-bit* zu *2*8-bit*



- Mehr Ports verfügbar (nützlich z.B. für SMP-Systeme)
- Flexiblere Verteilung der Bandbreite

Quelle: [3]

6. Link-Splitting [2/2]

- Konfiguration direkt bei der Initialisierung:
  - seit HT3.0: dedizierte CTL Leitung alle 8-bit
  - Entscheidung über Link-Splitting anhand deren Pegel

- Beispiel:
  - 16-bit Link hat zwei CTL-Leitungen, CTL[1:0]
  - Transmitter: signalisiert Link-Splitting support über CTL[1]

Signal	ohne LS	mit LS
CTL[0]	0	0
CTL[1]	1	0

7. Hot-Plugging [1/4]

- Was ist Hot-Plugging?
  Verändern der Systemkonfiguration im laufenden Betrieb. Wichtig für Hochverfügbarkeits-Systeme.

- Technische Realisierung:
  - Defined Link Termination Methods
  - Transaction Termination Behaviours
  - Sync Flood Isolation
  - Link Training Times

7. Hot-Plugging [2/4]

- Power, PWROK, RESET#, LDTSTOP# controlled by platform
- Software (or a service processor) is responsible for correctly configuring fields before clearing *Link Control[TXOff]*
- Isolation of sync floods between domains via software *[TXOff]*
- Software is responsible for cleaning up outstanding transactions on links that fail (optional hardware support)
- Changing link termination, ganging, or retry mode requires a *Warm Reset*
- added device must be configured to match all link properties of the devices that are already operational
- *HotPlugEn* modifies link termination in the *PHY OFF* state to protect from electrical transients during attach or detach (*TXGNDTERM*)

7. Hot-Plugging [3/4]

- Hot-add method (HT3.0 capable devices):
  - configure both sides to have matching link speed and width
  - set *LinkTrain[HotPlugEn]*
  - set *GlblLinkTrain[ConnDly]* on both sides
  - clear *LinkControl[TXOff]* and *[EndOfChain]* for the target links
  - assert *LDTSTOP#* or *RESET#* until both devices reach programmed frequency
- Hot-remove method:
  - set *GlblLinkTrain[ConnDly]* on both sides of the target links
  - set *LinkControl[TXOff]* and *[EndOfChain]* for the target links
  - assert *LDTSTOP#* or *RESET#* to disconnect the links

7. Hot-Plugging [4/4]

- Alternative Hot-add method (for legacy devices):
  - platform provides independent control of power and *PWROK* to the added device
  - existing device begins training
  - added device can correctly detect existing device at *Cold Reset*
  - Software later discovers and configures capabilities of the two devices
- allows hot-add of legacy devices
  - without JTAG or SMBus access
  - in systems without service processors but JTAG or SMBus interfaces

8. Power Management [1/4]

- Reduzierung des Strombedarfs in Ruhezeiten
- Dynamische Veränderung von
  - Taktrate der Verbindung
  - Breite der Verbindung
- Rapid Pause-Change-Start Support

- Direkt implementierbar in Hardware
  - Centralized Link Management Controller (CLMC)
  - keine Softwareschicht, effizient und schnell

8. Power Management [2/4]

- Centralized Link Management Controller (CLMC)
- Centralized dynamic link…
  - configuration (CDLC)
  - disconnection (CDLD)
  - width (CDLW)
  - frequency (CDLF)
- Centralized disconnected link refresh (CDLR)
- Centralized inactive lane refresh (CILR)

Registers that <u>must not</u> be modified by Software:
Link Frequency Register
LinkConfiguration[LinkWidthIn]
LinkConfiguration[LinkWidthOut]
GlblLinkTrain[T0Time]
GlblLinkTrain[InLnSt]
GlblLinkTrain[ConnDly]
LinkTrain[8b10b]
LinkTrain[ScrEn]
LinkTrain[LSSel]
LinkTrain[HotPlugEn]
LinkTrain[TestEn]
Tx/RxConfig Registers

8. Power Management [3/4]

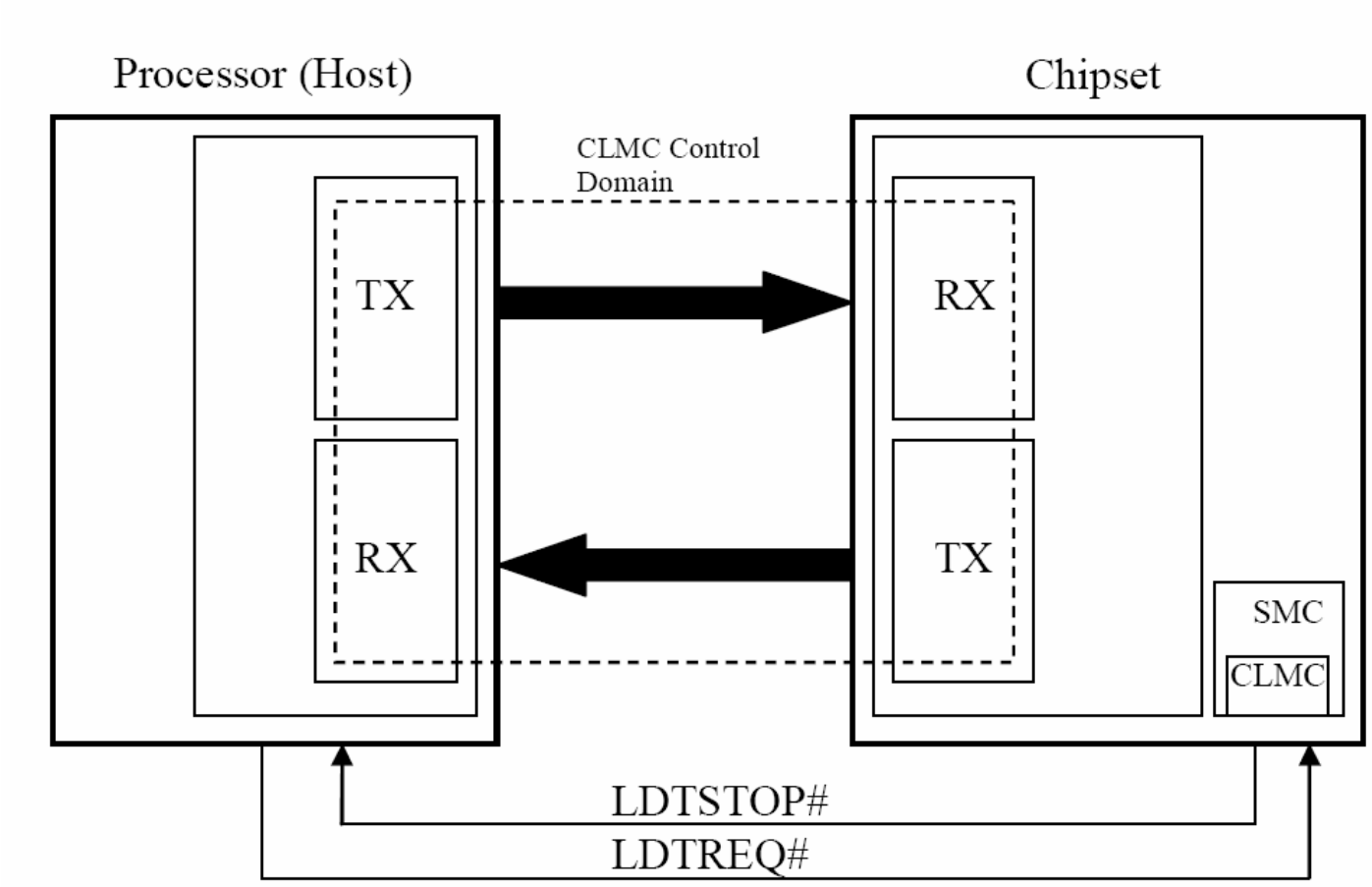


Figure 9. Centralized Link Power Management Control Topology

Quelle: [2]

8. Power Management [4/4]

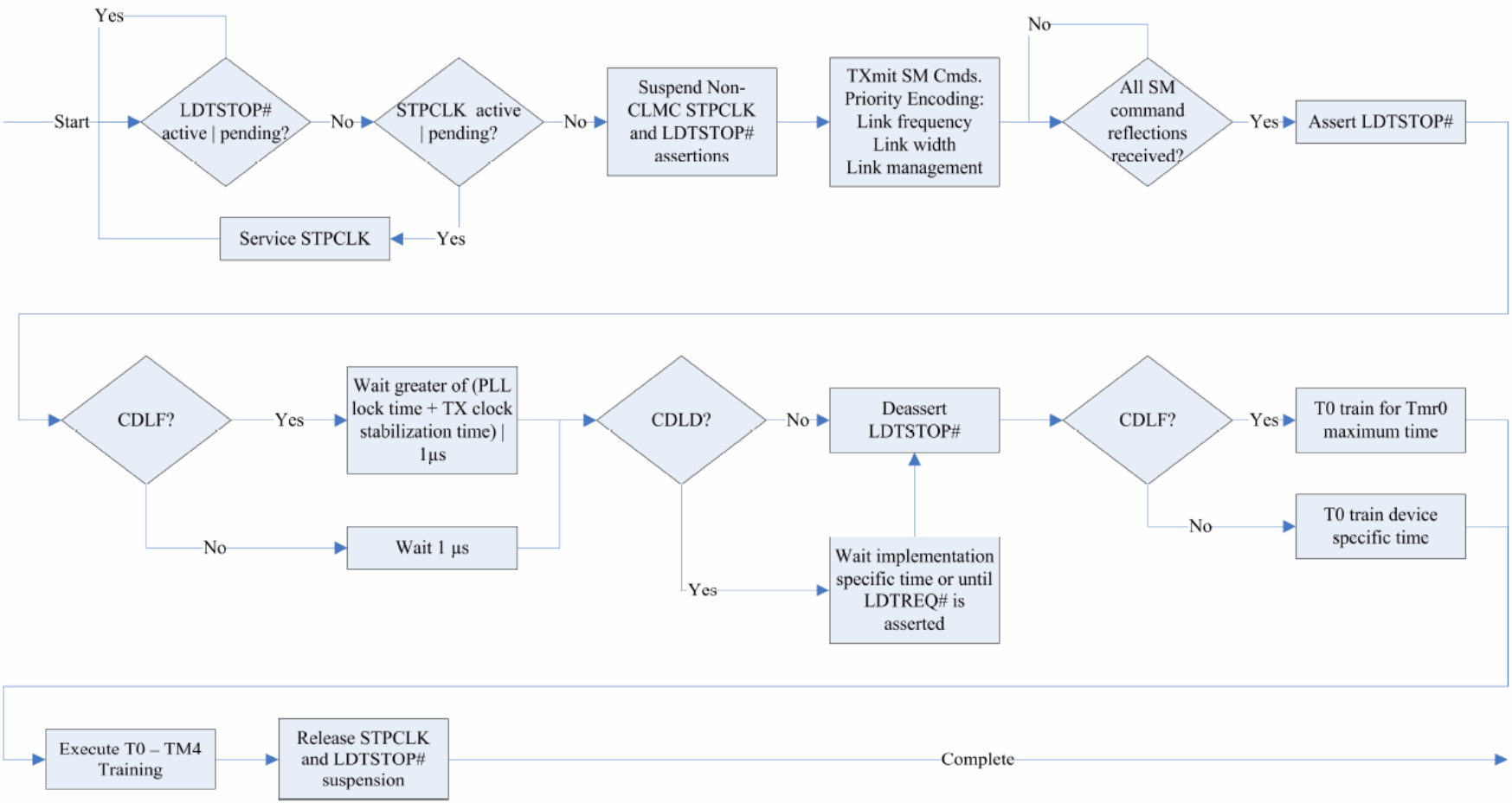


Figure 10. CLMC Control Feature Sequencing Diagram

Quelle: [2]

Aktuelle Anforderungen vs. HT3.0

	aktuell	HT 3.0	Vorteil HT3.0
Bandbreite	8.0GB/s	41.6GB/s	520%
Linkbreite	16-bit	32-bit	100%
Takt	1.0GHz	2.6GHz	162%
Leitung	DC	DC/AC	Distanz

- Fazit: HyperTransport 3.0 erfüllt heutige Anforderungen ohne Abstriche und bietet genügende Reserven für die (nahe?) Zukunft

Ende des Vortrags

Fragen?

Quellenverzeichnis

- [1] Hypertransport Consortium Website:
  - http://www.hypertransport.org/
- [2] Hypertransport 3.0 Specification:
  - http://www.hypertransport.org/docs/tech/HTC20051222-0046-0008-Final-4-21-06.pdf
- [3] Official Hypertransport 3.0 Presentation:
  - http://www.hypertransport.org/docs/tech/HT3pres.pdf
- [4] Mindshare - Hypertransport System Architecture:
  - http://www.mindshare.com/store/page.asp?p=0BA211E3