

Übertragungstechnik Ethernet

Labor Nachrichtentechnik

Ausgabe 1.0, 2016

Autor: Bernhard Klotz

Inhaltsverzeichnis

1	Überblick	4
2	Grundlagen	5
2.1	Entwicklung des Ethernet	5
2.2	Ethernet Schichtenmodell	5
2.3	Local Area Networks	6
2.3.1	Netzwerk Hierarchie	7
2.3.2	Netzelemente	7
2.3.3	Kabeltypen	9
2.3.4	Parameter	9
2.3.5	Kabel und Stecker	10
2.4	Frame-Format	12
2.5	CSMA/CD	13
2.6	10BASE-T	14
2.7	100BASE-TX	15
2.8	Autonegotiation	16
2.9	MDI/MDI-X	18
2.10	Power over Ethernet	18
3	Das Versuchsmodell	21
3.1	Hardware	21
3.1.1	MicroBoard	22
3.1.2	PHY	23
3.1.3	Media Independent Interface (MII)	24
3.1.4	MAC	27
3.1.5	Die Aufzeichnung der Ethernet-Signale	27
3.1.6	Triggersignale	28
3.2	Software	29
3.2.1	Initialisierung	29
3.2.2	Kommando Interpreter	30
3.2.3	Einstellungen der seriellen Schnittstelle	30
3.2.4	PHY-Kommandos	30
3.2.5	Einstellen der MAC-Adresse	30
3.2.6	Senden	31
3.2.7	Empfangen	31
3.3	Versuchsaufbau	31
3.3.1	Blockschaltbild	31
3.3.2	Oszillograph	32
3.3.3	PC	32
4	Die Bedienung des Versuchsmodells	33
4.1	Befehlssyntax	33
4.2	Befehlsüberblick	33
4.3	Beschreibung der Befehle	34
4.3.1	ADxx Advertise	34
4.3.2	AUTN Auto Negotiation	34
4.3.3	DUPL Duplex-Modus	35
4.3.4	FAST Fast Ethernet (100 Mbit/s)	35
4.3.5	HELP Print help text	35
4.3.6	MDXE Enable Auto MDIX	36
4.3.7	MDXF Force MDIX	36
4.3.8	PMAC Print MAC addresses	36
4.3.9	PPHY Print PHY Registers	36
4.3.10	PRCV Print Received Frames	38
4.3.11	PSIG Print Signal Selection	38
4.3.12	RNEG Restart Auto Negotiation	39
4.3.13	SEND Send a Message	39
4.3.14	SENS Send Sequence	40
4.3.15	SIGN Signal Selection	40
5	Versuchsdurchführung	41

5.1	Vorbereitung des Versuchs.....	41
5.2	Versuche mit 10BASE-T	42
5.2.1	Anfang eines Frames bei 10BASE-T	42
5.2.2	Ende eines Frames bei 10BASE-T	43
5.2.3	Gesamter Frame bei 10BASE-T	43
5.2.4	Empfangener Frame bei 10BASE-T	43
5.2.5	Darstellung von Augendiagrammen.....	43
5.2.6	Untersuchung des Spektrums.....	44
5.3	Versuche mit 100BASE-TX.....	44
5.3.1	Gesendete Signale	45
5.3.2	Dekodieren der gesendeten MLT-3-Zustände	45
5.3.3	Augendiagramm	45
5.3.4	Empfangene Signale	45
5.3.5	Spektrum	45
5.4	Autonegotiation	46
5.4.1	Normal Link Puls	46
5.4.2	Normal Link Pulse vom PC	46
5.4.3	Einzelner Fast Link Pulse.....	46
5.4.4	FLP Burst	46
5.5	Kollisionen.....	47
5.5.1	Kollision in einem Frame	47
5.5.2	Wiederholung nach Kollisionen	48
6	Literatur.....	49
7	Abkürzungen.....	50

1 Überblick

Dieser Versuch befasst sich mit der Übertragungstechnik des Ethernets, also mit der physikalischen Schicht und dem Übertragungsmedium. Das Ethernet hat in seinem Bestehen seit den Achtziger Jahren des letzten Jahrhunderts – für Kommunikationstechnik eine Ewigkeit! – eine rasante Entwicklung durchgemacht. Dabei wurde hauptsächlich die Übertragungsgeschwindigkeit in die Höhe getrieben, von anfangs 10 Mbit/s auf heute schon 100 Gbit/s. Als Übertragungsmedien kamen sowohl Kupfer als auch Glas, jeweils in unterschiedlichen Ausführungen zur Anwendung. Von einigen Sonderfällen (z.B. Token Ring oder Fiber Distributed Digital Interface, FDDI) abgesehen hat sich die Ethernet-Technik in allen drahtgebundenen Local Area Networks – den LANs – durchgesetzt. Selbst für größere Entfernungen gibt es inzwischen Lösungen, so dass diese Technik auch im Bereich der Netzbetreiber, der Carrier, Einzug hält und die traditionellen Telekommunikationsverfahren wie SDH und ATM zum Teil verdrängen wird.

Die immense Steigerung der Übertragungsgeschwindigkeit war nur möglich, indem man alle Register der Übertragungstechnik gezogen hat, so dass die höheren Geschwindigkeiten nicht mehr mit einfachen Labormitteln untersucht werden können. Wir beschränken uns daher auf die Geschwindigkeiten 10 Mbit/s und 100 Mbit/s und auf das Medium Kupferkabel.

Für die Durchführung des Versuches wird neben dem Versuchsmodell „Übertragungstechnik Ethernet“ nur ein Oszilloskop und ein PC benötigt.

Neben der reinen übertragungstechnischen Funktion wird auch auf die praktische Realisierung von Local Area Networks – den LANs – auf der Basis von Ethernet eingegangen, so dass man auch Hinweise für den Aufbau eines eigenen Netzes erhält.

Für die Durchführung des Versuchs sind die hier enthaltenen Grundlagen Voraussetzung. Zur Vorbereitung des Laborversuchs werden Fragen gestellt, die vor der Durchführung schriftlich beantwortet werden müssen. Diese vorbereitenden Aufgaben sind so gekennzeichnet:

Frage xx:

Bearbeiten Sie diese Fragen bitte schriftlich vor der Durchführung des Laborversuchs.

2 Grundlagen

2.1 Entwicklung des Ethernet

Die Anfänge des Ethernet in den Siebziger-Jahren kamen aus dem Hause Xerox und wurden später von den drei Firmen DEC (ist in HP aufgegangen) Intel und Xerox vorangetrieben und schließlich 1983 von der IEEE (Institute of Electrical and Electronics Engineers) als 802.3 standardisiert. Dieses Verfahren benutzte ein („dickes“) Koaxialkabel als Bus und arbeitete mit der Übertragungsgeschwindigkeit von 10 Mbit/s. Damals wurde die auch heute noch benutzte Nomenklatur eingeführt: 10BASE5. Die 10 steht für die Übertragungsgeschwindigkeit (10 Mbit/s), BASE für Basisbandübertragung (es gab auch Varianten mit BROAD = Breitband) und die 5 für die erreichbare Segmentlänge, hier 500 m.

Danach gab es die erste Variante, 802.3a mit einem „dünnen“ Koaxialkabel (10BASE2) und schließlich 1990 das legendäre 10BASE-T für ein Twisted-Pair-Kabel – daher das T. Im Jahr 1990 gab es schließlich als 802.3i das 100BASE-TX-Verfahren, auch als „Fast Ethernet“ bezeichnet mit der Übertragungsgeschwindigkeit von 100 Mbit/s. Daneben gab es noch unzählige Varianten für verschieden Kupferleitungen und Glasfaser-Typen. Mit Glasfasern konnte vor allem die Übertragungsgeschwindigkeit leichter erhöht werden und es konnten größere Entfernungen erzielt werden.

Es folgte das Gigabit-Ethernet, zuerst auf Glas und dann auf einem Twisted Pair als 802.3ab. Aus der Bezeichnung ab kann man sehen, dass es noch weitere Zwischenschritte gegeben hat, die meisten davon haben aber keine Bedeutung erlangt. Gigabit-Ethernet, 1000BASE-T, ist heute der Standard auf Netzwerkkarten von modernen PCs, zusammen mit den wegen der Kompatibilität weiterhin unterstützten Verfahren 100BASE-TX und 10BASE-T.

Und es geht weiter: 10 Gbit/s auf Twisted Pair (10GBASE-T) und natürlich Glas, und dann 40 Gbit/s und 100 Gbit/s. Letztere sind hauptsächlich für Glasfasern vorgesehen, oder für kurze, spezielle Kupferkabel. Und z.Z. wird an einem 400 Gbit/s Ethernet gearbeitet.

2.2 Ethernet Schichtenmodell

Der Ethernet-Standard bezieht sich auf die Schichten 1 und zwei des OSI-Modells. Die darüber liegenden Schichten können je nach Anwendung bestimmt werden, heute sind es meist IP und TCP. Das Bestechende an Ethernet ist die weitgehende Beibehaltung von Funktionen, Formaten und Schnittstellen über die vielen Jahre und die vielen Entwicklungsschritte im Standard.

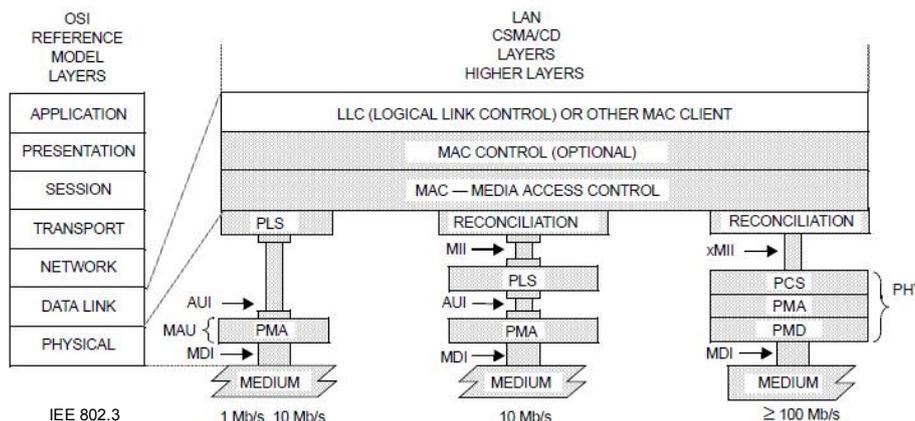


Abbildung 1: Die Schichten des Ethernets

In Abbildung 1 erkennt man, dass die Sicherungsschicht, umgangssprachlich auch als MAC-Layer bezeichnet, unabhängig vom Medium und der Übertragungsgeschwindigkeit ist. Die funktionale Aufteilung und die Schnittstellen des Modells sind eher als konzeptionelle Hilfsmittel gedacht und müssen nicht zwangsweise in einer Implementierung genau so realisiert werden. Trotzdem findet sich manche Schnittstelle als reales Interface z.B. zwischen Chips wieder.

Mit der Einführung des Fast Ethernet (100 Mbit/s) wurde das Medium Independent Interface (siehe 3.1.3) definiert, mit dem alle Informationen zwischen MAC und PHY übertragen werden können. An den bisherigen MAC-Layer musste dafür der Reconciliation-Layer ange-setzt werden.

Darunter befinden sich die Funktionen Physical Layer Signaling (PLS) und Physical Medium Attachment (PMA) bei 10 Mbit/s und Physical Coding Sublayer (PCS), PMA und Physical Medium Dependant Sublayer (PMD). Der PLS für 10 Mbit/s erkennt die Zustände auf dem Medium. Der PCS setzt die Daten des MAC-Frames in den Leitungscodex um, der PMA sorgt für die Synchronisation und scrambled, und der PMD passt die Informationen an das physikalische Medium an.

In dem Laborversuch werden wir uns mit dem Physical Layer und dem MAC-Layer des Ethernets beschäftigen und zwar für 10 und 100 Mbit/s über TP.

2.3 Local Area Networks

Kupfer und Glas sind die beiden Medien, die von Ethernet benutzt werden. Glasfasern sind in modernen Netzen eher für längere Übertragungstrecken und höhere Geschwindigkeiten, z.B. im Backbone- oder Primärbereich gedacht. Im End- oder Tertiärbereich, wo also die einzelnen Endgeräte angeschlossen sind, überwiegt die Verdrahtung mit Kupferkabeln.

In der Anfangszeit des Ethernets wurden Koaxialkabel verwendet, und zwar als Busse. Alle Endgeräte eines Segments wurden mit einem Kabel untereinander verbunden, jedes Endgerät sendet und empfängt über dasselbe Kabel. Senden ist allerdings nur erlaubt, wenn kein anderes Endgerät gerade sendet. Dafür wurde ein Verfahren eingeführt, Carrier Sense Multiple Access/Collision Detection (CSMA/CD), das die Zugriffe regelt und Kollisionen, die trotzdem möglich sind, auflöst.

Seit dem Ethernet aber für die Twisted Pair (TP) Kabel definiert war – beginnend mit 10BASE-T – sind die Koaxialkabel verschwunden. Zumindest für neue Netze werden sie nicht

mehr eingesetzt, da die Twisted Pair-Verkabelung wesentlich einfacher und wirtschaftlicher ist. Mit dem Umstieg auf TP wurde auch die Topologie geädert, aus einer Bus-Verkabelung wurde eine Stern-Topologie. Jedes Endgerät ist jetzt mit einem eigenen Kabel direkt mit einer aktiven Komponente, einem Hub, Switch oder Router, verbunden. Da auch beide Richtungen auf eigenen Adernpaaren laufen, hätte man auch auf CSMA/CD verzichten können, da es auf einer Punkt-zu-Punkt-Verbindung keine Kollisionen mehr geben kann. Aus Gründen der Kompatibilität mit älteren Komponenten oder Netzwerk-Abschnitten wurde dieses Verfahren jedoch erst mit den 10Gbit/s-Varianten abgeschafft.

2.3.1 Netzwerk Hierarchie

Bei großen Firmennetzen werden drei Bereiche unterschieden:

- der Primärbereich oder das Backbonenetz, meist mit Glasfasern, wenige km
- der Sekundärbereich oder Steigbereich, Glasfaser (2000 m) oder Kupfer (100m)
- der Tertiärbereich oder die Etagenverkabelung, meist Kupfer (90m)

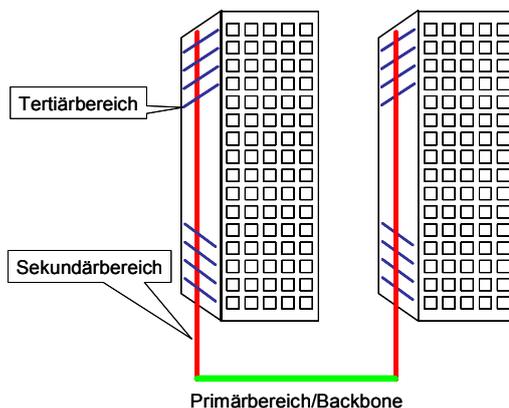


Abbildung 2: Verkabelungshierarchie

An den Übergangspunkten enden die Kabel an den Patchpanels und können von dort mit Patchkabeln mit den aktiven Netzelementen wie Bridges, Switches oder Router verbunden werden. Diese Netzelemente besorgen auch die Umsetzung zwischen verschiedenen Medien und Geschwindigkeiten und können den Verkehr konzentrieren.

2.3.2 Netzelemente

Neben den Endgeräten gibt es in einem LAN auch mehrere Netzelemente, die die empfangenen Frames auf verschiedene Weise weiterleiten können. Je nach Funktion werden diese unterschieden.

2.3.2.1 Netzwerkkarten

Netzwerkkarten sind die Interfaces in den Endgeräten und in den Servern. Bei den PCs und Notebooks sind heute Ethernet-Karten mit 1000/100/10 Mbit/s für TP Standard. Die Vor-

gänger-Modelle unterstützten 100/10 Mbit/s. Voll- und Halbduplex werden generell unterstützt, ebenso wie die Autonegotiation. Karten für Koaxialkabel gibt es nicht mehr.

Für Server werden Karten angeboten mit bis zu 100 Gbit/s, teils schon mit mehreren Interfaces pro Karte und mit TP und Glasfaser-Anschluss.

2.3.2.2 Hub/Repeater

Repeater und Hubs stammen noch aus der Zeit der Koaxialkabel-LANs. Sie verbinden zwei Kabelsegmente miteinander und arbeiten auf der physikalischen OSI-Schicht. Sie regenerieren hauptsächlich die empfangenen Frames, indem sie die Signalamplitude und das Timing wiederherstellen. Kollisionen werden über alle verbundenen Segmente weitergegeben. In modernen Netzen sind sie nicht mehr anzutreffen.

2.3.2.3 Bridge

Eine Bridge verbindet ebenfalls zwei Kabelsegmente, aber anders als beim Repeater werden beide auf der Schicht zwei abgeschlossen. Frames werden nur weitergeleitet, wenn die Adressen im jeweils anderen Segment gefunden werden. Damit wird neben der Regeneration eine Verkehrstrennung erreicht, und die Kollisionen sind auf das jeweilige Segment beschränkt. Eine Bridge braucht dafür aber Tabellen mit den jeweils zugeordneten Adressen, die aber dynamisch gefüllt wird. Bridges sind in modernen Netzen auch nicht mehr zu finden.

2.3.2.4 Switch

Die Ablösung der Koaxialkabel durch Twisted Pairs und Glasfasern hat weitreichende Änderungen auch in der Netztopologie mit sich gebracht. Damit wurden nämlich reine Punkt-zu-Punkt-Verbindungen eingeführt, man konnte im Vollduplex-Betrieb arbeiten und Kollisionen auf der Verbindung gehörten der Vergangenheit an. Um aus diesen Punkt-zu-Punkt-Verbindungen aber ein Netz zu bauen, in dem jedes Endgerät mit jedem kommunizieren kann braucht es zentrale Vermittlungsknoten, nämlich Switches.

Switches haben eine Menge von Ports, an die die einzelnen Endgeräte direkt angeschlossen werden. Jeder Port hat einen eigenen PHY- und einen eigenen MAC-Layer, und alle Ports sind über eine Switch-Matrix miteinander verbunden. Abhängig von den MAC-Adressen werden empfangene Frames an die anderen Ports weitergeleitet. Da Frames von mehreren Ports gleichzeitig für einen einzigen Ausgangsport bestimmt sein können, muss der Switch Frames puffern können.

Switches können auch hierarchisch angeordnet sein, so dass jeweils ein Port der unteren Switches mit einem Port eines darüberliegenden Switches verbunden wird. Auf diese Weise können sehr große Netze gebildet werden. Die Adresstabelle eines Switches wird dynamisch aufgebaut, so dass keine Konfiguration notwendig ist.

Neben diesen eben beschriebenen Switches – den sogenannten Layer-2-Switches – gibt es aber noch Erweiterungen. Zusätzlich wurden in die Switches auch Routing-Funktionen eingebaut, also das Switching auf dem Layer 3 des OSI-Modells, konkret auf IP-Ebene. Man spricht dann von einem Layer-3-Switch oder einem Router.

Damit aber nicht genug: je nach Anwendung können auch Informationen aus den höheren Schichten, ja bis zur Anwendungsschicht, zum Weiterleiten der empfangenen Frames genutzt werden. Die Bezeichnungen sind dann aber nicht mehr streng festgelegt.

2.3.3 Kabeltypen

Mit jeder Erhöhung der Übertragungsgeschwindigkeit werden schärfere Anforderungen an das Twisted Pair Kabel gestellt, die mit unterschiedlichen Kabeltypen erfüllt werden.

Die Kabel sind in verschiedene Kategorien eingeteilt, und ein bestimmtes Übertragungsverfahren erfordert mindestens die passende Kategorie. Kabel einer höheren Kategorie decken die Anforderungen der niedrigeren Kategorien ebenfalls ab.

Für LANs kommen die folgenden Kabel in Frage:

- Cat3: bis 10 MHz für 10BASE-T
- Cat5: bis 100 MHz für 100BASE-TX und für 1000BASE-T
- Cat6a: bis 625 MHz für 10GBASE-T

Cat3 und Cat5-Kabel sind UTP, Cat6a ist STP. Es gibt auch schon verschiedene Cat7-Kabel, die aber keine höheren Geschwindigkeiten ermöglichen. Möglicherweise werden für höhere Geschwindigkeiten im LAN-Bereich künftig Glasfasern statt Kupferkabel verlegt.

LAN-Kabel bestehen grundsätzlich aus 8 Kupferadern, von denen jeweils zwei zu einem Paar verdreht sind. Die Ausführungen unterscheiden sich hauptsächlich in der Art der Schirmung, die mit dem Schema x/y TP bezeichnet werden. Die Buchstaben U (unshielded, ungeschirmt), F(Foil, Folie), S (Screen, Drahtgeflecht) und SF (Screen und Foil) können auftreten:

- UTP: unshielded TP
- FTP: foiled TP, jedes Adernpaar ist mit einer Folie geschirmt
- S/UTP oder F/UTP: gemeinsame Schirmung um alle Adernpaare
- S/FTP, F/FTP oder SF/FTP: jedes Adernpaar ist mit einer Folie geschirmt, alle Adernpaare nochmals mit Drahtgeflecht, Folie oder beidem

Grundsätzlich wird noch unterschieden zwischen Patch- und Verlegekabel. Die Adern der Patchkabel bestehen aus Litze, die der Verlegekabel aus massiven Drähten. Patchkabel sind damit wesentlich flexibler.

Patchkabel werden mit Steckern versehen (meist RJ-45 gecrimpt), Verlegekabel werden an Dosen oder Patchpanel aufgelegt, meist mit Schneid-Klemm-Technik (LSA)

2.3.4 Parameter

Die Qualität eines Kabels wird von den verwendeten Materialien und seinem Aufbau bestimmt, wobei es nicht nur auf die eingesetzten Schirmungen ankommt, sondern auch auf die Präzision der Fertigung. Die Eigenschaften eines Kabels werden aber nicht nur bei der Herstellung bestimmt, sondern auch bei der Verlegung. So müssen bestimmte minimale Biegeradien eingehalten werden, das Kabel darf nicht starkem Zug ausgesetzt sein und darf bei der Befestigung nicht verformt oder beschädigt werden. Verdrehungen dürfen nur in engen Grenzen aufge-

löst werden und die Adernlängen dürfen sich nur gering unterscheiden. Daneben sind natürlich die Umgebungsanforderungen einzuhalten.

Elektrische Parameter von Bedeutung sind:

- Wellenwiderstand (ca. 100 Ω)
- Dämpfung (attenuation)
- Nebensprechen (Crosstalk NEXT, FEXT) oder auch von anderen Kabeln als Fremdnebensprechen (Alien Crosstalk AXTLK)
- Rückflussdämpfung (Return Loss): Reflektionen, die bei unregelmäßigem Kabelaufbau oder äußeren Störungen auftreten können
- Laufzeitunterschiede der Adern (Delay Skew)

2.3.5 Kabel und Stecker

Die Adern der LAN-Kabel sind nach EIA/TIA 568 farbcodiert und zwar mit den Farben blau, grün, orange und braun. Dabei sind die beiden Adern eines Paares mit derselben Farbe, eine uni und die andere abwechselnd mit weiß gekennzeichnet.



Abbildung 3: Steckerbelegung nach EIA/TIA 568B auf Kontaktseite gesehen

In Tabelle 1 ist die Belegung des RJ-45 Steckers nach EIA/TIA 568B gezeigt. Die Farben der Adern sind auch festgelegt. Die 4 Adernpaare sind dabei wie folgt auf die Steckstifte aufgelegt:

Stift	Adernfarbe	Paar
1	weiß/orange	2
2	orange	2
3	weiß/grün	3
4	blau	1
5	weiß/blau	1
6	grün	3
7	weiß/braun	4
8	braun	4

Tabelle 1: Steckerbelegung nach EIA/TIA 568B

Mit aufsteigender Stiftnummer kommt zuerst weiß/Farbe und dann Farbe, außer bei den Stiften 4 und 5, da ist die Reihenfolge umgekehrt. Außerdem fällt auf, dass die Adern des Paares 3 nicht nebeneinander liegen. Die Norm EIA/TIA 568B ist bei konfektionierten Patchkabeln

Standard und auch sonst sehr verbreitet. Es gibt aber auch eine weitere Kodierungsart, EIA/TIA 568A.

Stift	Adernfarbe	Paar
1	weiß/grün	3
2	grün	3
3	weiß/orange	2
4	blau	1
5	weiß/blau	1
6	orange	2
7	weiß/braun	4
8	braun	4

Tabelle 2: Steckerbelegung nach EIA/TIA 568A

Bei 10BASE-T und 100BASE-TX werden ja nur zwei Adernpaare benutzt, das Paar 2 für die Senderichtung (Tx+ und Tx-) und das Paar 3 für die Empfangsrichtung (Rx+ und Rx-). Dies ist die Sicht einer Netzwerkkarte, bei einem Switch liegt der Empfänger am Paar 2 und der Sender am Paar 3. Die normalen Kabel sind 1:1 durchverbunden, an beiden Enden sind also die Stifte 1 miteinander verbunden, usw. Die höheren Geschwindigkeiten nutzen alle vier Adernpaare aus.

Will man zwei Netzwerkkarten miteinander verbinden, so müssen die Sende- und Empfangspaare gekreuzt werden, man braucht ein Crossover-Kabel. Die Steckstifte 1 und 2 sind mit 3 und 6 der jeweils anderen Seite verbunden. Die übrigen Adern sind weiterhin 1:1 verbunden. Dieses Crossover-Kabel erhält man, wenn man auf der einen Seite nach EIA/TIA 568B und auf der anderen nach EIA/TIA 568A belegt.

Bei den meisten Netzwerkkarten und Switches werden die Sende- und Empfangspaare automatisch ermittelt (das Verfahren heißt Auto-MDI-X), so dass Crossover-Kabel überflüssig werden, oder ein fälschlicherweise verwendetes Crossover-Kabel keine Auswirkung hat.

RJ-45 Stecker können mit einer Crimp-Zange mit dem vorbereiteten Kabel verbunden werden. Die Ummantelung des Kabels muss entfernt werden, die Verdrillung der Adern wird aufgelöst, und die Adern werden auf die benötigte Länge gekürzt. Dann werden die Adern in der richtigen Farb-Reihenfolge in den Stecker gesteckt und mit der Crimpzange befestigt.



Abbildung 4: Crimpzange und RJ-45 Stecker

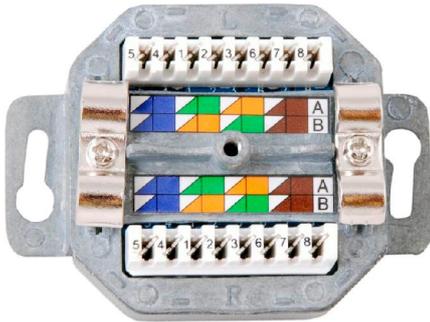


Abbildung 5: Verbindungsseite einer RJ-45 Doppeldose (EFA)

Die Adern des Verlegekabels werden mit einem speziellen LSA-Auflege-Werkzeug mit den Klemmen verbunden. LSA steht dabei für löt- schraub- und abisolierfrei. Die isolierte Ader wird einfach mit dem Werkzeug in die Klemme gedrückt und das überstehende Ende wird gleichzeitig abgeschnitten. Die Klemmen der Dosen sind meist farbcodiert (siehe Abbildung 5).



Abbildung 6: LSA-Werkzeug

Die immer höheren Übertragungsgeschwindigkeiten erfordern immer bessere Kabel und dazu natürlich auch Steckverbindungen, die mit der Kabelqualität Schritt halten. Der RJ-45 Stecker kann bis 500 MHz eingesetzt werden, höhere Anforderungen werden von GG45 oder Tera Steckern erfüllt.

2.4 Frame-Format

Die Daten, die über das Ethernet geschickt werden, bezeichnet man auf dem OSI Data Link Layer als Frame. Für den Physical Layer können noch weitere Felder, z.B. für die Synchronisation, notwendig sein, diese Felder und der Frame werden dann als Packet bezeichnet. Allerdings wird nicht überall streng zwischen Packet und Frame unterschieden.

Der Begriff „Byte“ ist ebenfalls nicht eindeutig definiert, obwohl man landläufig darunter 8 Bits versteht. Damit aber erst gar keine Verwechslungsgefahr besteht haben die Übertragungstechniker den Begriff „Oktett“ (engl. „Octet“) eingeführt, um eindeutig eine Datenmenge von 8 Bits zu bezeichnen.

Das Format eines Ethernet Frames (Abbildung 7) ist in allen Versionen gleich. Infolge des langen Zeitraums der Standardisierung haben sich einige begriffliche Verwirrungen eingeschlichen, die aber letztendlich zu demselben Format führen. Die unterschiedlichen Bezeichnungen, z.B. des Length/Type Feldes oder die zusätzlichen Daten in MAC Client Data Feld haben nur für die höheren Schichten eine Bedeutung.

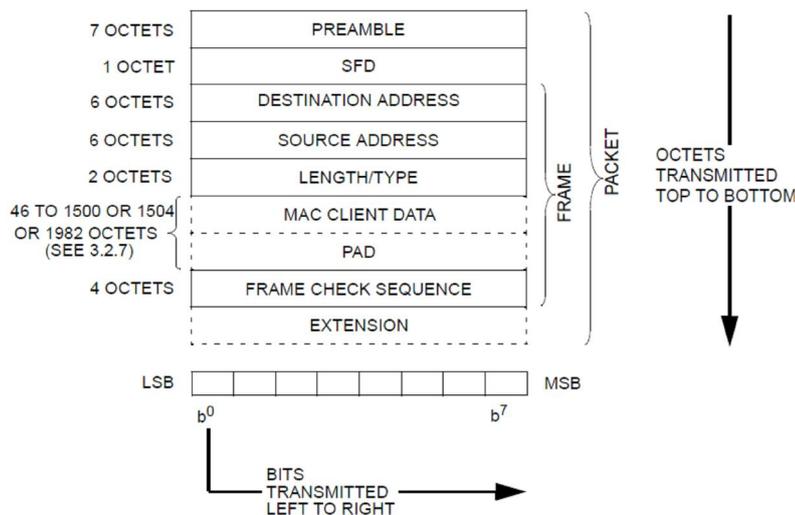


Abbildung 7: Format eines Ethernet Frames (IEEE 802.3)

Die Preamble besteht aus 7 Oktetts mit der Bitfolge 1010 1010 (hexadezimal x55). Der Start of Frame Delimiter (SFD) hat den Wert 1010 1011 (xD5). Die Oktetts werden mit dem niederwertigsten Bit ganz links dargestellt, und dieses Bit wird auch zuerst ausgesendet.

Eine Besonderheit weist die Länge des Nutzdatenfeldes auf. Es hat im „Basic MAC Frame“ eine Mindestlänge von 48 Oktetts, und eine Maximallänge von 1500 Oktetts. Allerdings kann dieses Feld für bestimmte Protokoll Daten verlängert werden, auf 1504 Oktetts bei VLANs („Q-tagged“) oder auf bis zu 1982 Oktetts als „Envelope“ mit zusätzlichen Feldern (als Prefix oder Suffix zu den Daten) für andere Protokolle, z.B. MPLS.

Die Extension wird benötigt um bei 1000BASE-T ein Frame künstlich zu verlängern, so dass auch bei der kurzen Sendezeit eines Packets Kollisionen erkannt werden können. Die Extension besteht aus besonderen Zeichen auf dem Medium, die nicht als Daten erkannt werden können.

2.5 CSMA/CD

Das Zugriffsverfahren Carrier Sense Multiple Access/Collision Detection ist bei einem „Shared Medium“, also einem Übertragungsmedium, das von mehreren Stationen benutzt wird, unverzichtbar. Dies war der Fall bei den frühen Ethernet-Varianten, die als Bussysteme mit Koaxialkabel ausgeführt waren.

Auf dem gemeinsamen Medium darf zu einer Zeit nur eine einzige Station senden, es ist grundsätzlich nur Halbduplex-Betrieb möglich. Eine Station prüft vor dem Senden, ob das Medium von einer anderen Station belegt ist (Carrier Sense). Ist es frei, beginnt sie zu senden, prüft aber laufend, ob ihre eigenen Daten tatsächlich unversehr auf dem Medium ankommen, oder mit der Aussendung einer anderen Station kollidieren und so verfälscht werden. Obwohl alle Stationen nur bei einem freien Medium senden, kann es infolge der endlichen Ausbreitungsgeschwindigkeit zu Kollisionen kommen. Im Falle der erkannten Kollision bricht jede Station ihre Sendung ab, und versucht es zu einem späteren, zufällig gewählten Zeitpunkt wieder.

Es ist wichtig, dass ein Sender erkennt, dass seine Daten infolge einer Kollision nicht erfolgreich gesendet werden konnten, weil dadurch eine schnellere Wiederholung möglich ist, als wenn man sich auf ein Quittungs- und Wiederholungsverfahren in den höheren Schichten (z.B.

TCP) verlassen würde. Damit aber Kollisionen zuverlässig erkannt werden, muss abhängig von der maximalen Ausdehnung des Mediums eine minimale Sendedauer eingehalten werden. So können auch bei zwei Stationen an den entfernten Enden des Mediums und bei ungünstigen Zeitpunkten (die zweite Station beginnt mit dem Senden kurz bevor die Daten der anderen Station bei ihr eintreffen) Kollisionen erkannt werden.

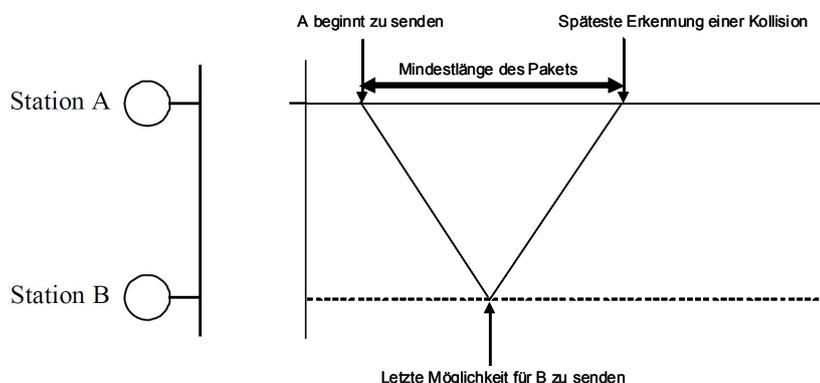


Abbildung 8: Kollisionserkennung bei CSMA/CD

Diese minimale Sendedauer wird durch die Mindestlänge der Nutzdaten von 46 Oktetts garantiert. Hat eine Station weniger Daten zu senden, müssen dies Daten auf 46 Oktetts durch „Padding“ (Wert: 00) erweitert werden.

Der Bereich, in dem sich eine Kollision ausbreiten kann – im einfachsten Fall eben das gesamte Koaxialkabel – nennt man Collision Domain. Sie ist begrenzt durch die maximal erlaubte Länge eines Segmentes. Die Zeit um das kürzeste Packet zu senden wird „slot time“ genannt und ist 512 Bitzeiten (64x8) lang. Nach dem Erkennen einer Kollision bricht der Sender das Aussenden der Daten ab, sofern die Preamble schon komplett ausgesendet ist, fügt aber noch 4 Oktetts mit den Bits (in vielen Fällen) 0101 0101 (xAA), das sogenannte „Jam-Signal“ an.

Bei den neueren Übertragungsverfahren werden ausschließlich Punkt-zu-Punkt-Verbindungen verwendet, und jede Übertragungsrichtung hat ihr eigenes exklusives Medium. Es kann also nicht mehr zu solchen Kollisionen kommen, bei denen Bits auf der Leitung durch Überlagerung verfälscht werden. Die Betriebsweise Halbduplex ist aber immer noch möglich, sei es, dass auf einem anschließenden Netz noch mit Koaxialkabel gearbeitet wird, oder dass End- oder Transitknoten einfach nicht in der Lage sind, gleichzeitig zu senden und zu empfangen. Erst mit dem 10 Gbit/s Ethernet wurde auf Halbduplex verzichtet.

Kollisionen werden bei 10BASE-T, 100BASE-TX und 1000BASE-T also nicht mehr durch Bitverfälschungen erkannt, sondern dadurch, dass während des Sendens Daten empfangen werden. Das Auflösen der Kollision erfolgt aber wie bisher: Abbrechen des Sendevorgangs, Jam-Signal und Wiederholung zu einem zufälligen Zeitpunkt.

2.6 10BASE-T

Dieser Standard beschreibt die Übertragung mit einer Geschwindigkeit von 10 Mbit/s über eine Kupferleitung – üblicherweise ein LAN-Kabel. Es werden nur zwei Adernpaare genutzt, eines für die Hin-, ein zweites für die Rückrichtung. Als Leitungscode wird der Manchester-Code verwendet, ein Return-to-Zero-Code. Das zu übertragende Bit ist in den Flanken des

Signals codiert, eine steigende Flanke bedeutet hier eine Eins, eine fallende eine Null. Ein Paket beginnt grundsätzlich mit einer fallenden Flanke aus dem Ruhezustand, also der Spannung 0 V. Die Spannung am Senderausgang beträgt ungefähr $\pm 2...3$ V. Bei einer Datenrate von 10 Mbit/s würde bei einer binären Übertragung nach Nyquist eine Bandbreite von 5 MHz ausreichen. Wegen des Manchester-Codes wird jedoch die doppelte Bandbreite benötigt.

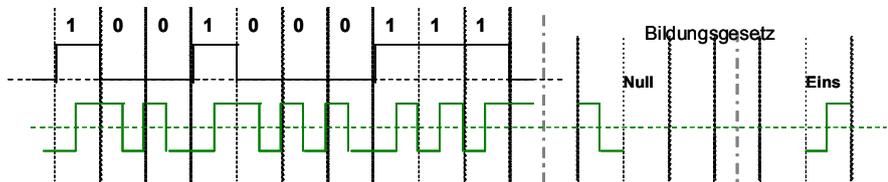


Abbildung 9: Manchester-Code nach IEEE 802.3

Der Manchester-Code ist gleichspannungsfrei und erlaubt eine leichte Ableitung des Taktes.

Bei 10BASE-T wird die Leitung nur belegt, wenn ein Paket zu senden ist, ansonsten liegt keine Spannung an. Damit aber die Verfügbarkeit der Verbindung überwacht werden kann, sendet jede Seite im Ruhezustand alle ca. 16 ms „Link Integrity Test (LIT)“-Pulse. Wenn ein Empfänger über eine längere Zeit (50 ms ... 150 ms) weder Daten noch Pulse enthält, nimmt er einen Fehler an. Dieser Puls wird auch „Normal Link Pulse (NLP)“ genannt.

2.7 100BASE-TX

Für die Übertragung mit einer Geschwindigkeit von 100 Mbit/s über Kupferkabel wird im Vergleich zu 10BASE-T nur die Leitungscodierung geändert. Es wird hier der MLT-3-Code verwendet, der jedoch weder gleichspannungsfrei ist, noch einen Taktgehalt hat.

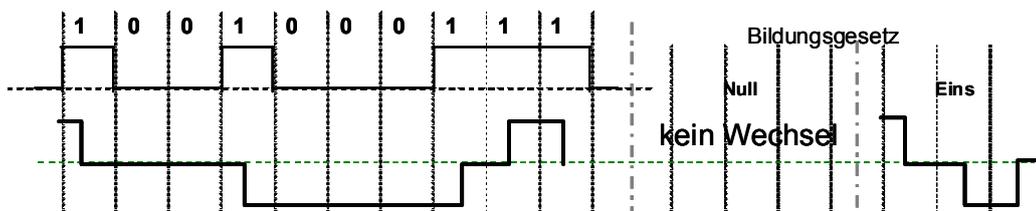


Abbildung 10: MLT-3-Code

Diese beiden Eigenschaften werden durch die vorherige 4B5B-Codierung erreicht. Dabei werden 4 Nutzbits in 5-Bit-Blöcke auf der Leitung codiert. Es gibt 32 verschiedene 5-Bit-Blöcke, davon werden 16 für die Daten benötigt. Es werden diejenigen ausgewählt, die keine langen Nullfolgen ergeben, so dass sich kein großer Gleichspannungsanteil aufbauen kann, und dass genügend Taktinformation enthalten ist.

Von den dann noch nicht benützten Blöcken werden 2 zur Kennzeichnung eines Paket-Anfangs verwendet, sie ersetzen das erste Byte der Preamble. Zwei weitere Blöcke kennzeichnen das Ende eines Pakets, sie werden nach der FCS geschickt.

Der Block mit lauter Einsen wird als IDLE-Signal gesendet, d.h. im Gegensatz zu 10BASE-T wird auch im Ruhezustand laufend ein Signal auf die Leitung gegeben, die LIT-Pulse werden nicht gebraucht. Außerdem gibt es noch ein Fehler-Signal.

	PCS code-group [4:0] 4 3 2 1 0	Name	MII (TXD/RXD) <3:0> 3 2 1 0	Interpretation
D A T A	1 1 1 1 0	0	0 0 0 0	Data 0
	0 1 0 0 1	1	0 0 0 1	Data 1
	1 0 1 0 0	2	0 0 1 0	Data 2
	1 0 1 0 1	3	0 0 1 1	Data 3
	0 1 0 1 0	4	0 1 0 0	Data 4
	0 1 0 1 1	5	0 1 0 1	Data 5
	0 1 1 1 0	6	0 1 1 0	Data 6
	0 1 1 1 1	7	0 1 1 1	Data 7
	1 0 0 1 0	8	1 0 0 0	Data 8
	1 0 0 1 1	9	1 0 0 1	Data 9
	1 0 1 1 0	A	1 0 1 0	Data A
	1 0 1 1 1	B	1 0 1 1	Data B
	1 1 0 1 0	C	1 1 0 0	Data C
	1 1 0 1 1	D	1 1 0 1	Data D
	1 1 1 0 0	E	1 1 1 0	Data E
	1 1 1 0 1	F	1 1 1 1	Data F
	1 1 1 1 1	I	undefined	IDLE, used as inter-stream fill code
C O N T R O L	1 1 0 0 0	J	0 1 0 1	Start-of-Stream Delimiter, Part 1 of 2; always used in pairs with K
	1 0 0 0 1	K	0 1 0 1	Start-of-Stream Delimiter, Part 2 of 2; always used in pairs with J
	0 1 1 0 1	T	undefined	End-of-Stream Delimiter, Part 1 of 2; always used in pairs with R
	0 0 1 1 1	R	undefined	End-of-Stream Delimiter, Part 2 of 2; always used in pairs with T
I N V A L I D	0 0 1 0 0	H	Undefined	Transmit Error; used to force signaling errors
	0 0 0 0 0	V	Undefined	Invalid code
	0 0 0 0 1	V	Undefined	Invalid code
	0 0 0 1 0	V	Undefined	Invalid code
	0 0 0 1 1	V	Undefined	Invalid code
	0 0 1 0 1	V	Undefined	Invalid code
	0 0 1 1 0	V	Undefined	Invalid code
	0 1 0 0 0	V	Undefined	Invalid code
	0 1 1 0 0	V	Undefined	Invalid code
	1 0 0 0 0	V	Undefined	Invalid code
	1 1 0 0 1	V	Undefined	Invalid code

Abbildung 11: Die 4B5B-Codierung (IEEE 802.3)

Nach der 4B5B-Codierung beträgt die Bitrate 125 Mbit/s und die entstandenen Bits werden noch gescrembled. Dadurch wird die spektrale Dichte auf der Leitung verteilt, es entsteht ein flaches Spektrum, unabhängig von den übertragenen Bitfolgen. Leider können dadurch keine Signale mehr auf der Leitung erkannt werden, man müsste zuerst descramblen und dabei Synchronisation mit dem Sender aufrecht erhalten. Trotz der Erhöhung der Bitrate auf 125 Mbit/s ergibt sich mit dem MLT-3 Code eine Bandbreite von nur 31,25 MHz.

2.8 Autonegotiation

Mit der Einführung von 100BASE-TX gab es schon Kompatibilitätsprobleme, denn die neuen Netzwerkkarten beherrschten sowohl 100BASE-TX als auch 10BASE-T. Also brauchte man eine Funktion, die bestimmen kann, welche Verfahren auf den beiden Enden einer Verbindung vorhanden sind, um das beste auszuwählen. Das leistet die Autonegotiation. Jede Seite gibt bekannt, welche Verfahren sie unterstützt, so dass das effizienteste Verfahren auf beiden Seiten genutzt wird. Dabei geht es hauptsächlich um die Übertragungsgeschwindigkeit und um den Halb- oder Voll duplexmodus.

Bei der Initialisierung einer Verbindung schickt jede Seite ihre Fähigkeiten in einem Paket, das mit Hilfe von Link Pulses übertragen wird. Das ist der „Fast Link Pulse (FLP)“. Im Gegensatz zum NLP geht es hier um eine ganze Pulsfolge, nämlich 17 Clock-Pulsen mit 16 dazwischen liegenden optionalen Datenpulsen. In diesen Datenpulsen wird das „Link-Codewort“ codiert, das die Fähigkeiten beschreibt.

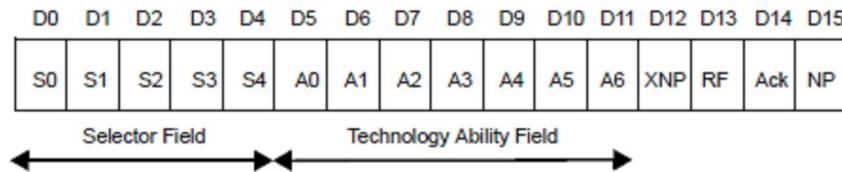


Abbildung 12: Die 16 Bits des Link-Codeworts (IEEE 802.3)

Mit den Selector-Bits wird der Standard der Übertragung angegeben, beim Ethernet ist das natürlich IEEE 802.3, also die Bits 00001.

S4	S3	S2	S1	S0	Selector description
0	0	0	0	0	Reserved for future Auto-Negotiation development
0	0	0	0	1	IEEE Std 802.3
0	0	0	1	0	IEEE Std 802.9a-1995 (withdrawn)
0	0	0	1	1	IEEE Std 802.5v-2001 (withdrawn)
0	0	1	0	0	IEEE Std 1394
0	0	1	0	1	INCITS
0	0	1	1	X	Reserved for future Auto-Negotiation development ^a
0	1	X	X	X	Reserved for future Auto-Negotiation development
1	X	X	X	X	Reserved for future Auto-Negotiation development

Abbildung 13: Die Selector-Bits des Link-Codeworts (IEEE 802.3)

In den darauffolgenden Technology-Bits werden die unterstützten Übertragungsgeschwindigkeiten und Modi angegeben.

Bit	Technology	Minimum cabling requirement
A0	10BASE-T	Two-pair Category 3
A1	10BASE-T full duplex	Two-pair Category 3
A2	100BASE-TX	Two-pair Category 5
A3	100BASE-TX full duplex	Two-pair Category 5
A4	100BASE-T4	Four-pair Category 3
A5	PAUSE operation for full duplex links	Not applicable
A6	Asymmetric PAUSE operation for full duplex Links	Not applicable

Abbildung 14: Die Technology-Bits des Link-Codeworts (IEEE 802.3)

Daneben gibt es noch Quittungen, Fehlermeldungen und herstellerspezifische Erweiterungen.

Ein Terminal, das nur 10BASE-T beherrscht betrachtet den FLP-Burst als NLP und ein Terminal mit Autonegotiation betrachtet den Empfang von NLPs als Anzeige von 10BASE-T.

Da Autonegotiation bei 10BASE-T und 10BASE-TX eine Option ist (nicht mehr bei Giga-bit-Ethernet), kann es auch vorkommen, dass eine 100BASE-TX Station mit Autonegotiation nur 100 Mbit/s Codes empfängt, und gar keine FLPs. In diesem Fall wird 100 Mbit/s gewählt, da allerdings der Duplex-Modus nicht bestimmt werden kann, geht man auf Halbduplex. Diese Art der Erkennung wird „Parallel Detection“ genannt. Die Autonegotiation kann auch deaktiviert werden, so dass mit festen Voreinstellungen gearbeitet werden muss. Falls allerdings eine Seite Autonegotiation deaktiviert und 100 Mbit/s bei Vollduplex eingestellt hat, die andere Seite

aber mit Autonegotiation arbeitet, fällt diese auf Halbduplex zurück. Es kommt zu einem „duplex mismatch“ mit Kollisionen und einem verringerten Durchsatz.

Die Autonegotiation ist nur auf Kupferkabeln, den Twisted Pairs, vorgesehen, da darauf die verschiedenen Verfahren überhaupt möglich sind. Für Glasfasern ist Autonegotiation nicht definiert.

2.9 MDI/MDI-X

Bei der Verbindung zweier Ethernet-Terminals muss das Adernpaar vom Sender auf der einen Seite mit dem Empfängereingang auf der anderen Seite verbunden werden. Bei einem Netz ist normalerweise ein Computer mit einem Switch oder Router verbunden. Diese haben an ihren Eingängen (die heißen dann „Uplink-Ports“) die Sende- und Empfangsanschlüsse im Vergleich zu einer Netzwerkkarte im Computer vertauscht - gekreuzt. Oft sind am Switch pro Port zwei Buchsen vorhanden, eine mit ungekreuztem und eine mit gekreuztem Eingang. Diese werden dann mit MDI und MDI-X bezeichnet.

Um dann zwei Computer direkt miteinander zu verbinden braucht man ein Crossover-Kabel, bei dem die Sende- und Empfangspaare gekreuzt sind. Dadurch gab es immer wieder Verwirrungen, so dass ein Verfahren eingeführt wurde, das die Sende- und Empfangsanschlüsse automatisch vertauschen kann. Dieses Verfahren wird als Auto-MDI-X bezeichnet, damit werden Crossover-Kabel überflüssig (IEEE 802.3-2012, 40.4.4 Automatic MDI/MDI-X Configuration).

2.10 Power over Ethernet

Zu den typischen Ethernet-Endgeräten – PCs – kommen jetzt immer mehr kleinere Geräte, wie Telefone oder Sensoren, an deren Standorten oft keine Stromversorgung möglich oder gewünscht ist. Daher sollte es auch möglich sein, solche Geräte direkt über das LAN-Kabel zu versorgen. Diese Versorgung, Power over Ethernet (PoE) war in IEEE 802.3af spezifiziert und wurde in IEEE 802.3at-2009 erweitert.

Zwei Aufgaben mussten dabei gelöst werden:

- Die Lösung muss mit allen bisherigen Endgeräten kompatibel sein und darf kein Endgerät zerstören.
- Unterschiedlicher Energiebedarf der Endgeräte soll erfüllt werden, ohne die Versorgung zu überlasten.

Das wird mit Hilfe eines Energiemanagements erreicht, indem bei der Initialisierung durch das „Power Sourcing Equipment (PSE)“, meist in einem Switch, festgestellt wird, ob ein Endgerät (ein „Powered Device (PD)“) tatsächlich versorgt werden möchte, und mit welcher Leistung. In dieser Phase wird mit geringen Spannungen gearbeitet, und die Endgeräte, die gespeist werden sollen, weisen einen Widerstand auf, der vom Leistungsbedarf abhängt. Erst wenn das PSE den Bedarf erkannt hat, wird die Versorgungsspannung, nominal 48 V, angelegt. Die verfügbare Leistung ist in Klassen aufgeteilt und kann bis zu 25 W betragen. Höhere Leistungen sind nicht erlaubt, weil sich die Kabel infolge der Verlustleistung zu stark erwärmen könnten.

Die Spannungsversorgung darf die Kommunikation natürlich nicht stören. Zwei Möglichkeiten der Einspeisung sind erlaubt, Endgeräte müssen beide Möglichkeiten unterstützen. Die Einspeisung für 10BASE-T/100BASE-TX ist in Abbildung 15 gezeigt.

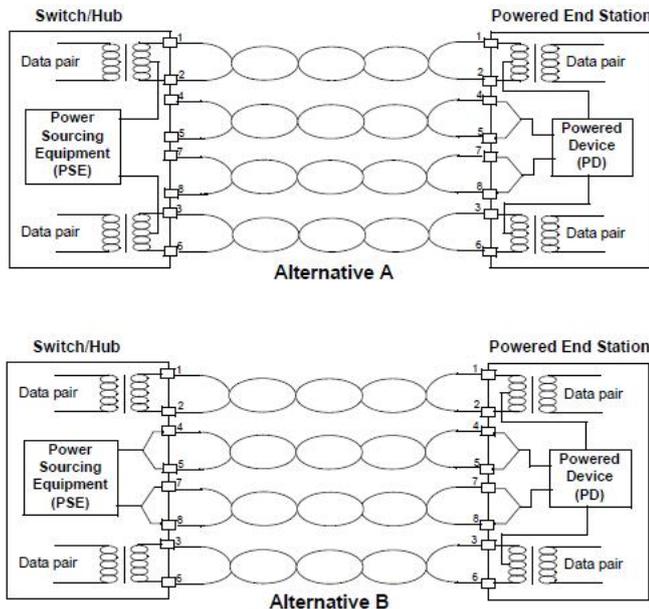


Abbildung 15: Einspeisemöglichkeiten für PoE (IEEE 802.3at)

Bei 1000BASE-T werden ja alle Adernpaare zur Kommunikation benutzt, so dass auch bei der Alternative B die Einspeisung durch die Übertrager erfolgt.

Fragen

Frage 1:

Wie lang müsste ein Frame mindestens sein, dass bei 1000BASE-T eine Collision Domain der Länge 100 m möglich wäre? Die Ausbreitungsgeschwindigkeit auf dem Kabel betrage 200 000 km/s.

Frage 2:

Welche Bandbreite ist bei der Übertragung von 10BASE-T mindestens erforderlich?

Frage 3:

Wie groß ist der Taktgehalt bei 10BASE-T?

Frage 4:

Was bedeutet SFD?

Frage 5:

Wozu werden einige der 4B5B-Blöcke verwendet, die keine Nutzdaten enthalten?

Frage 6:

Wie viel aufeinanderfolgende Nullen lässt der 4B5B-Code zu?

Frage 7:

Wie groß ist der Taktgehalt bei 100BASE-TX ohne Scrambling?

Frage 8:

Welche Bandbreite ist bei 100BASE-TX mindestens erforderlich?

Frage 9:

Was versteht man unter einer Kollision bei 10BASE-T?

Frage 10:

Wie stellen Sie die Verstärkung (y-Achse) und die Zeitbasis des Oszilloskops ein, wenn Sie ein 10BASE-T-Signal darstellen wollen, und ein paar Bits ausreichend sind?

Frage 11:

Wie stellen Sie den Oszilloskops für 100BASE-TX ein?

3 Das Versuchsmodell

Die Übertragungstechnik des Ethernets soll mit möglichst einfachen Mitteln untersucht werden, also ohne spezielle Protokoll-Testgeräte. Dafür soll der übliche Labor-Oszillograph verwendet werden, der mit den Signalleitungen des Ethernet verbunden wird. Für die Darstellung bestimmter Signalsequenzen auf dem Oszillographen ist es einmal notwendig, diese Sequenzen gezielt zu erzeugen, und zum andern auch, die passenden Triggersignale bereitzustellen, damit man diese Sequenzen auch innerhalb der großen Signalmenge auf den Leitungen finden und auf dem Oszillographen darstellen kann.

Die üblichen Netzwerkkarten für PCs und die Treiber der Betriebssysteme lassen sich meist nicht gezielt auf einzelne Betriebsarten einstellen, Fehlerfälle lassen sich nicht bewusst hervorrufen, und interne Signale sind nicht zugänglich. Daher viel die Wahl auf ein FPGA-Entwicklungssystem, das LX9-MicroBoard der Firma Avnet. Dieses Board enthält einen sehr leistungsfähigen FPGA, ein physikalisches Ethernet-Interface (PHY) und ein USB-Interface für ein Terminal-Programm auf einem PC. Durch passende Programmierung des FPGAs wird ein MAC-Layer, die Ansteuerung des PHY und ein Microcontroller – MicroBlaze – realisiert. Dieser wird so programmiert, dass über den PC alle notwendigen Einstellungen vorgenommen werden können und einzelne Testfälle gestartet und angezeigt werden können. Die notwendigen Triggersignale können aus dem MicroBoard ausgekoppelt werden.

3.1 Hardware

Das Versuchsmodell enthält alle notwendigen Baugruppen in einem Gehäuse: das MicroBoard, die Anpassungsschaltung für den Oszillographen an das Ethernet, die Anschlüsse für die Triggersignale und den USB-Anschluss zum PC für das serielle Interface und für die Spannungsversorgung des MicroBoards.



Abbildung 16: Die Frontplatte des Versuchsmodells

3.1.1 MicroBoard

Das MicroBoard ist ein Evaluation-Kit der Firma AVNET [8] für das Xilinx Spartan-6 FPGA. Zusätzlich enthält das Board einen Ethernet-Anschluss mit einem PHY-Layer für 10BASE-T und 100BASE-TX der Firma Texas Instruments (DP83848x) [2], einen USB-Anschluss zum PC mit serieller Schnittstelle, einen weiteren USB-Anschluss zur Programmierung, RAM- und Flash-Speicher, und Buchsen für externe Verbindungen, z.B. zu den LEDs auf der Frontplatte. Ein DIP-Schalter wird für die Auswahl von MAC-Adressen verwendet, die Tasten sind für Reset und Reload vorgesehen, sind aber nicht auf die Frontplatte herausgeführt.

In dem FPGA läuft ein MicroBlaze Microcontroller, der zusammen mit der Logik für den MAC-Layer und die eigenen Funktionen in VHDL für den FPGA programmiert ist. Die Logik und die Software für den Microcontroller – in C programmiert – sind im Flash-Speicher permanent abgelegt und werden beim Einschalten in den FPGA und in das RAM geladen.

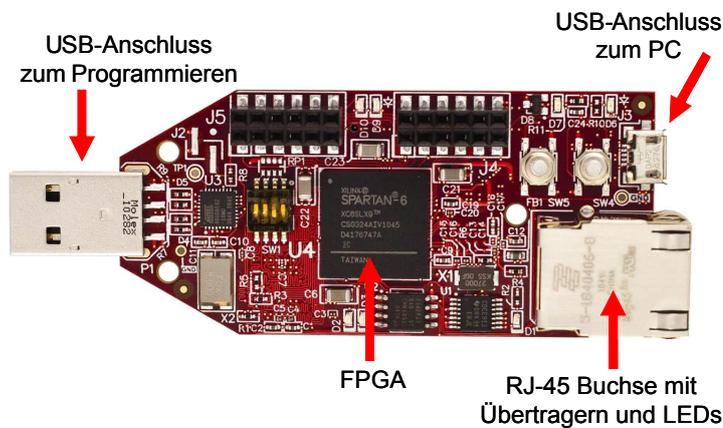


Abbildung 17: Die Oberseite des MicroBoards

Die Signale der LEDs, die auch auf dem Board sind, werden über die PMod-Buchsen (oben in Abbildung 17) herausgeführt, ebenso wie die wichtigen Signale zwischen PHY und MAC. Die Leuchtdauer der LEDs wird über Monoflops im FPGA verlängert, so dass auch sehr kurze Signale wahrgenommen werden können.

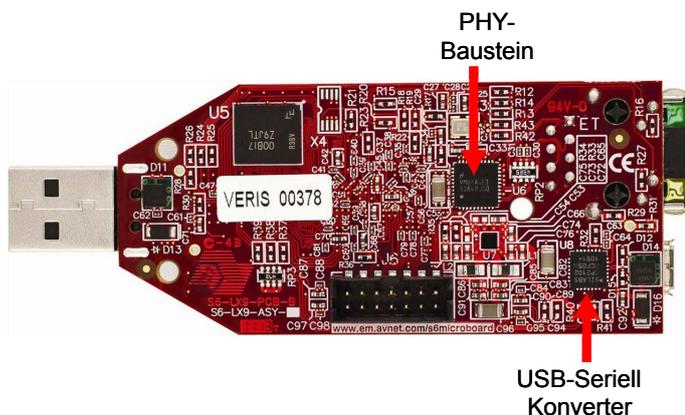


Abbildung 18: Die Unterseite des MicroBoards

Der USB-Programmier-Anschluss und die JTAG-Steckerleiste sind nur im Gehäuse zugänglich, die RJ-45-Buchse für das Ethernet und der USB-Anschluss für den PC sind nach außen geführt.



Abbildung 19: Das Innere des Versuchsmodells

3.1.2 PHY

Der PHY-Baustein (Texas Instruments DP83848x) ist in [2] ausführlich beschrieben. Er enthält einmal die Aufbereitung der elektrischen Signale von und zu den Übertragern, die in der RJ-45-Buchse integriert sind und die gesamte Sende- und Empfangslogik für die beiden Verfahren 10BASE-T und 100BASE-TX.

Der zu sendende Ethernet-Frame, von der Preamble bis einschließlich FCS, wird dem PHY über das MII in 4-Bit-Stücken (den „Nibbles“) übergeben, und vom PHY mit dem Leitungscodierung gesendet. In der Empfangsrichtung wird der Frame, ebenfalls von Preamble bis FCS in gleicher Weise an den MAC übergeben.

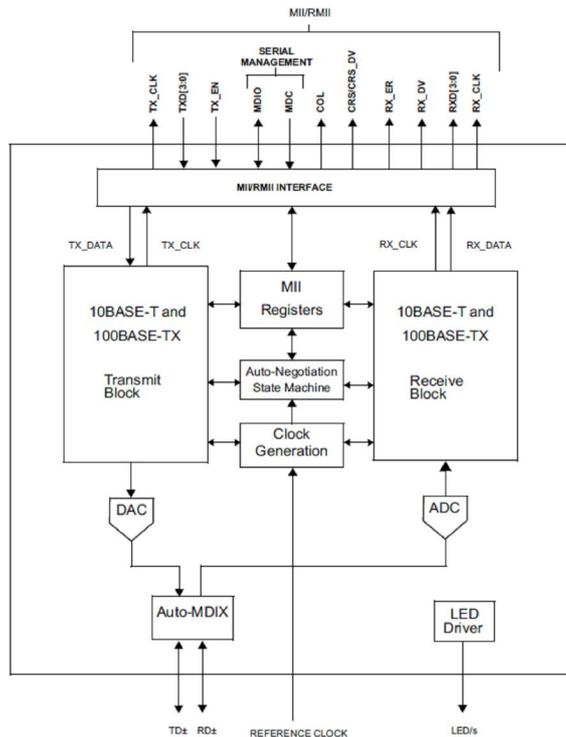


Abbildung 20: Blockschaltbild des 10/100Mbit/s Ethernet Transceivers

Der PHY kann über ein Management-Interface eingestellt werden, die wichtigsten Parameter sind:

- Geschwindigkeit
- Vollduplex/Halbduplex
- Autonegotiation
- MDI-X

Über dasselbe Interface können auch Zustände und Informationen abgefragt werden. Die Details dazu sind im Abschnitt 3.2.3 beschrieben.

3.1.3 Media Independent Interface (MII)

Der PHY-Baustein hat das Media Independent Interface implementiert. Zusätzlich wäre noch das Reduced Media Independent Interface (RMII) möglich, es wird aber hier nicht verwendet. Beim RMII sind weniger Signalleitungen nötig, die Daten werden aber mit einer höheren Rate zwischen PHY und MAC übertragen.

Das Management Interface ist ein serielles Interface mit zwei Signalen: ein Takt (Management Data Clock MDC) zum PHY und ein bidirektionales Datensignal (Management Data Input Output MDIO).

Die Daten vom MAC zum PHY werden über 4 Datenleitungen (TXD[0:3]), einem Takt vom PHY (TX_CLK) und einem Steuersignal (TX_EN) vom MAC übertragen. Das MAC zeigt mit TX_EN (TX Enable) an, dass ein Frame übergeben werden soll, dann werden mit dem Takt des PHY jeweils 4 Bit gleichzeitig vom MAC bereitgestellt. Die Taktrate hängt von der Ge-

schwindigkeit auf dem Ethernet ab, bei 10BASE-T beträgt sie 2,5 MHz, bei 100BASE-TX 25 MHz.

Die empfangenen Daten werden mit den Signalen RXD[0:3], RX_CLK und RX_DV vom PHY zum MAC auf gleiche Weise übertragen. Mit RX_DV (RX Data Valid) zeigt das PHY an, dass gültige Daten anliegen.

Zusätzlich erzeugt das PHY noch die Signale CRS (Carrier Sense), COL (Collision) und RX_ER (RX Error). Mit CRS zeigt das PHY an, dass bei Vollduplex die Empfangsleitung belegt ist (für die Dauer des ganzen Frames) und dass bei Halbduplex die Sende- oder die Empfangsleitung belegt ist. Mit COL wird eine Kollision signalisiert, wenn also im Halbduplex-Modus gleichzeitig beide Richtungen aktiv sind. Mit RX_ER wird ein Fehler beim Empfang von ungültigen Leitungscodes bei 100BASE-TX signalisiert.

Die Signalverläufe am MII sind für die Versuche deshalb wichtig, weil die Signale zum Triggern des Oszilloskops verwendet werden, um bestimmte Sequenzen auf den Leitungen zu finden.

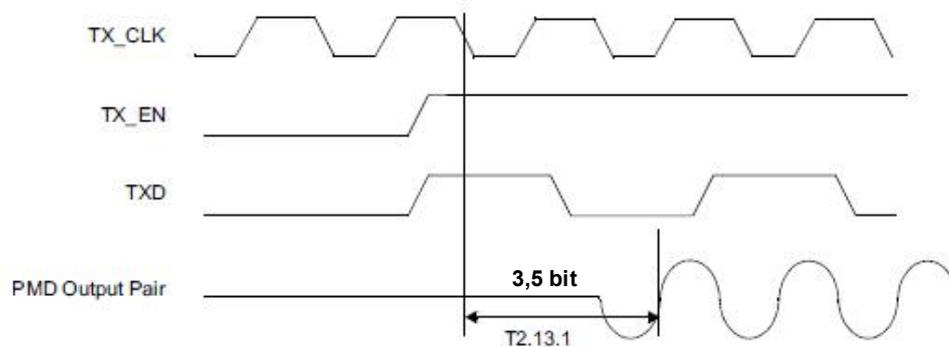


Abbildung 21: 10BASE-T Transmit Timing (Start of Packet)

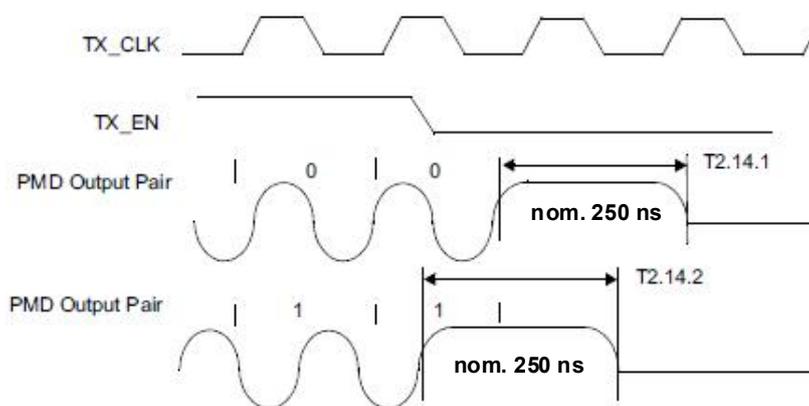


Abbildung 22: 10BASE-T Transmit Timing (End of Packet)

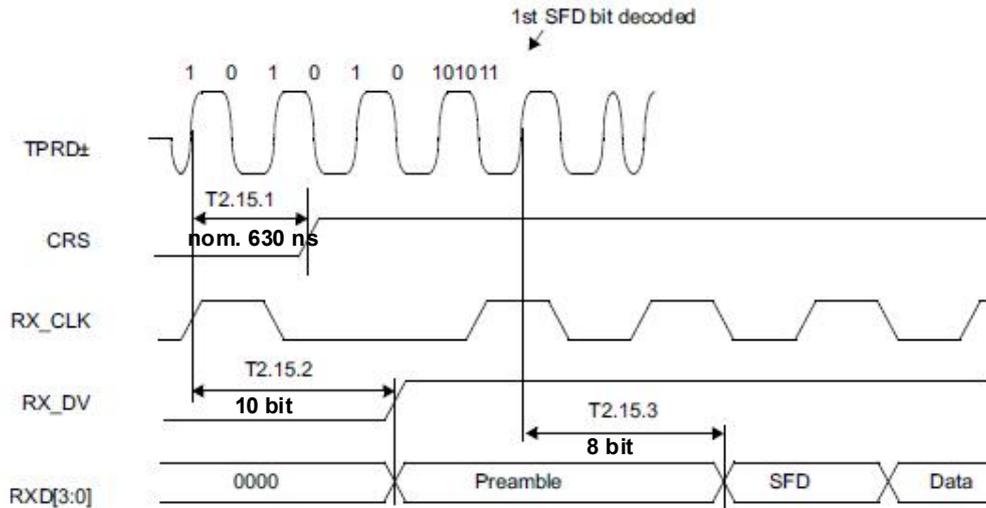


Abbildung 23: 10BASE-T Receive Timing (Start of Packet)

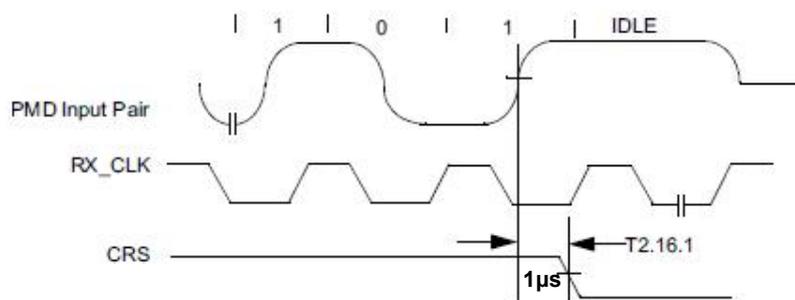


Abbildung 24: 10BASE-T Receive Timing (End of Packet)

Leider sind bei 100BASE-TX solche Diagramme nicht sehr aussagekräftig, weil an dem Signal auf den Leitungen infolge des Scramblings nichts mehr erkennbar ist.

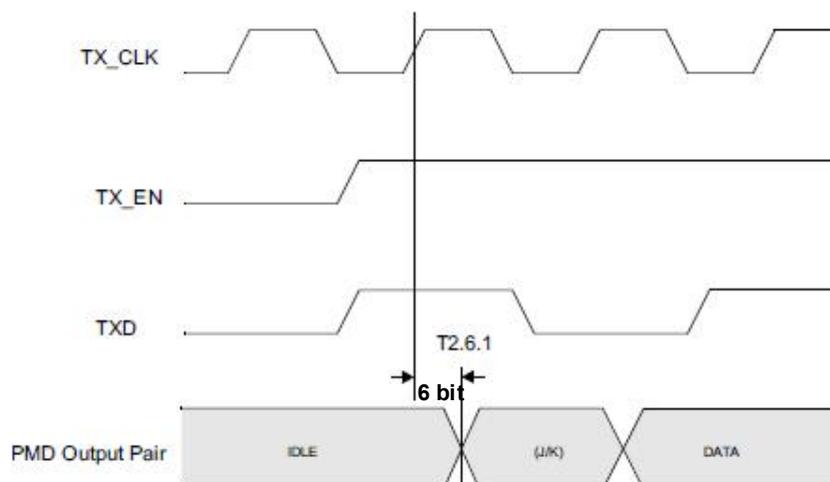


Abbildung 25: 100BASE-TX Transmit Packet Latency Timing

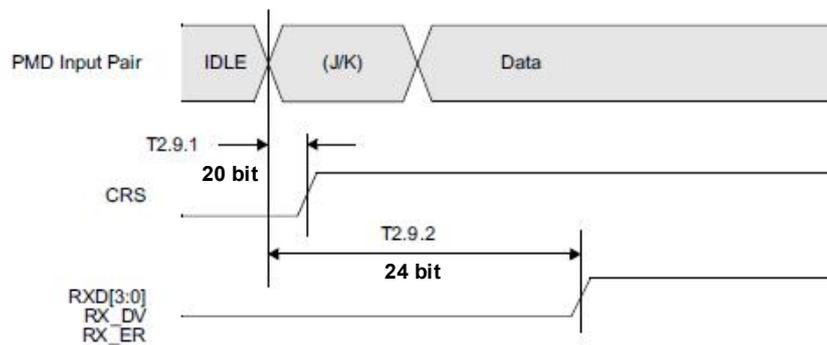


Abbildung 26: 100BASE-TX Receive Packet Latency Timing

Senden und Empfangen ist im PHY gleichzeitig möglich, das PHY arbeitet grundsätzlich im Vollduplex-Modus. Wird das PHY auf Halbduplex eingestellt berichtet es nur das Auftreten von Kollisionen zum MAC Layer. Die Behandlung einer Kollision ist aber die Aufgabe des MAC-Layers.

3.1.4 MAC

Der MAC-Layer ist als Logik im FPGA implementiert. Zum Senden eines Frames werden die Daten von den MAC-Adressen bis zum Ende der Nutzdaten, aber ohne FCS vom MicroBlaze bereitgestellt. Der MAC-Layer ergänzt Preamble und FCS und übergibt die Daten als Nibbles dem PHY.

In der Empfangsrichtung erhält der MAC-Layer die Daten vom PHY, setzt den Frame ohne Preamble zusammen und prüft die FCS und danach die Destination MAC Adresse (auf die eigene MAC oder die Broadcast Adresse) und die Länge der Daten. Erst wenn alle Prüfungen richtig sind, wird ein Interrupt an den MicroBlaze erzeugt. Die Interrupt-Prozedur im MicroBlaze muss die Daten auslesen und verarbeiten und danach dem MAC in einem Statusbit anzeigen, dass das Frame bearbeitet ist. Erst danach kann der MAC-Layer weitere Daten vom PHY akzeptieren.

Die Logik des MAC kann entweder für Voll- oder für Halbduplex ausgelegt werden. Ein Wechsel der Betriebsart per Software ist leider nicht möglich. Da es bei dem Versuch überhaupt nicht auf hohen Datendurchsatz ankommt, sondern auf die Demonstration von prinzipiellen Vorgängen, ist der MAC-Layer mit Halbduplex fest konfiguriert. Solange nicht explizit Kollisionen untersucht werden, wird der PHY-Baustein trotzdem auf Vollduplex eingestellt. Dadurch wird ein Abbruch einer Aussendung vermieden, wenn gleichzeitig ein Frame empfangen wird.

Der MicroBlaze versorgt den MAC-Layer bei der Initialisierung mit einer eigenen MAC-Adresse. Sie kann aber später auch geändert werden.

3.1.5 Die Aufzeichnung der Ethernet-Signale

Die Signale auf den beiden Adernpaaren des LAN-Kabels sollen auf einem Oszillografen dargestellt werden. Üblicherweise haben Oszillografen hochohmige, unsymmetrische Eingänge, die alle einen gemeinsamen Massepunkt haben. Damit der Anschluss eines Oszillografen die Potentiale auf dem LAN-Kabel nicht beeinflusst, wurde eine Trennung mit zwei Übertragern

gewählt. Allerdings werden dadurch auch die Signalpegel vermindert, was aber für die Reichweite im Labor unerheblich ist.

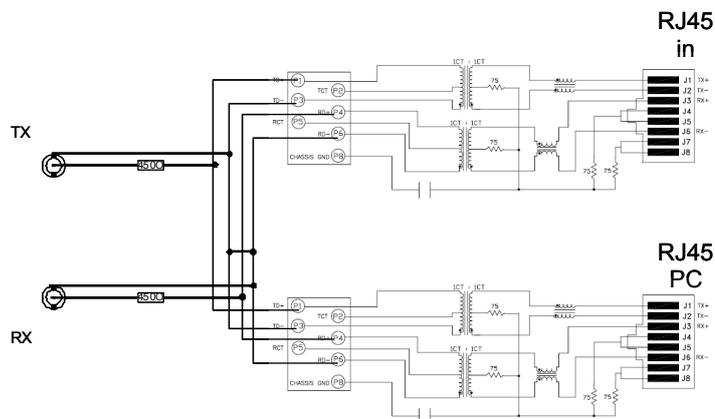


Abbildung 27: Anschluss des Oszillografen an das LAN_Kabel

Durch die Trennung des Oszillografen-Anschlusses können auch gefahrlos PCs angeschlossen werden.

Bei der hohen Bandbreite von über 30 MHz müssen die Meßleitungen zum Oszillografen angepasst werden. Deshalb ist hier ein Teiler 1:10 vorgesehen, der aus einem Widerstand von 450 Ω an den Übertragern und 50 Ω am Eingang des Oszillografen besteht. Damit ist auch das Koaxialkabel mit 50 Ω Wellenwiderstand angepasst. Am Eingang des Oszillografen muss auf jeden Fall ein 50 Ω Feedthrough oder ein T-Stück mit einem 50 Ω Abschlusswiderstand angeschlossen werden.

3.1.6 Triggersignale

Die Frames auf dem Ethernet können sehr lang sein, so dass es sehr schwierig sein kann, bestimmte Abschnitte auf dem Display des Oszillografen zu identifizieren. Daher sollen Trigger-Signale helfen z.B. Anfang oder Ende eines Frames aufzufinden, einzelne Bits zu identifizieren oder Kollisionen zu entdecken.

Dafür werden Signale am MII des PHY nach außen geführt und auch an den Oszillografen angeschlossen. Die folgenden Signale sind vorgesehen:

- TX_CLK
- TX_EN
- RX_CLK
- RX_DV
- RX_ER
- COL
- CRS
- TX_CLK and TX_EN
- RX_CLK and RX_DV

- TX_EN or RX_DV
- TXD(0)
- TXD(1)
- RXD(0)
- RXD(1)

Damit können leicht Anfang und Ende eines gesendeten (TX_EN) oder empfangenen Frames (RX_DV) erkannt werden, oder Kollisionen (COL). Mit Hilfe der Takte TX_CLK und RX_CLK können Bits auf dem Ethernet identifiziert werden. Dabei ist aber einmal ein Zeitversatz zu berücksichtigen (siehe 3.1.3) und die Taktrate von $\frac{1}{4}$ der Bitrate, da ja am MII mit einem Taktimpuls gleich 4 Bits übergeben werden. Die UND-Verknüpfung von Clock und Enable oder Data Valid dient zur Darstellung von Augendiagrammen. Mit der ODER-Verknüpfung von Enable und Data Valid kann auf ein empfangenes oder gesendetes Frame getriggert werden. Mit TXD und RXD können zwei Bits am MII dargestellt werden.

Drei Signale können gleichzeitig an die BNC-Buchsen herausgeführt werden, welche der 14 möglichen, kann durch Kommandos festgelegt werden. Auch hier ist wegen der hohen Frequenzen auf einen Abschluss der Meßleitung am Oszilloskop zu achten.

3.2 Software

Die Software für das Versuchsmodell läuft auf dem MicroBlaze Prozessor, sie ist in C geschrieben und mit dem Software Design Kit von Xilinx entwickelt worden. Die Software wurde in den Flash-Speicher des MicroBoards geladen und wird beim Anschluss der USB-Verbindung in den MicroBlaze geladen und startet.

Die Aufgaben der Software sind die Initialisierung des Ethernet-Interfaces, die Kommunikation über eine serielle Schnittstelle über USB mit einem Terminalprogramm auf dem PC, das Dekodieren und Ausführen der Befehle und das Annehmen und Speichern von Frames.

3.2.1 Initialisierung

Sobald das MicroBoard mit Strom versorgt wird, wird der FPGA aus dem Flash-Speicher geladen, zusammen mit einem Bootloader, der die Software ebenfalls aus dem Flash liest und in das externe RAM ablegt. Danach startet das eigentliche Programm. Es initialisiert den PHY-Baustein und den MAC-Layer und es lädt eine vorgegebene MAC-Adresse in den MAC-Layer, testet die Grundfunktionen und meldet sich über die serielle Schnittstelle mit Programmnamen, Version und lokaler MAC-Adresse. In Abhängigkeit des rechten DIP-Schalters SW1 auf dem Board wird entweder die normale Adresse 00-0A-35-01-02-03 (Schalter oben) oder die alternative Adresse 00-0A-35-01-02-05 (Schalter unten) geladen.

```

-----
|                               D H B W                               |
-----
| Ethernet Test Terminal V 9.0   3.5.2016                           |
-----

```

```

Local MAC 00-0A-35-01-02-03
Remote MAC 00-1D-72-00-09-E6

```

Cmd:

Oder:

```
-----
|                          D H B W                          |
-----
| Ethernet Test Terminal V 9.0   3.5.2016                   |
-----
```

Alternative Local MAC 00-0A-35-01-02-03

Remote MAC 00-1D-72-00-09-E6

Cmd:

Mit „Cmd:“ wird angezeigt, dass das Board zur Annahme von Kommandos bereit ist.

3.2.2 Kommando Interpreter

Der Interpreter nimmt ein Kommando, das mit RETURN abgeschlossen ist an, interpretiert es und führt es aus, wenn es syntaktisch richtig war. Die Kommandos bestehen aus 4 Buchstaben und können von einem oder zwei Parametern gefolgt sein. Das Ergebnis des Kommandos wird über die serielle Schnittstelle zurückgemeldet.

3.2.3 Einstellungen der seriellen Schnittstelle

Die serielle Schnittstelle arbeitet mit 115200 Baud, 8 Datenbits, kein Parity, 1 Stopbit und ohne Flußkontrolle.

3.2.4 PHY-Kommandos

Zur Einstellung des PHY-Bausteins gibt es eine Reihe von Kommandos, die Daten im PHY ändern oder auslesen. Die wichtigsten Kommandos dienen zum Einstellen der Übertragungsgeschwindigkeit, der unterstützten Funktionen bei der Autonegotiation, von MDX und der Ausgabe der PHY-Register, und damit dem Zustand der Verbindung.

3.2.5 Einstellen der MAC-Adresse

Die eigene MAC-Adresse wird bei der Initialisierung vorgegeben, entweder als normale oder als alternative Adresse. Sie kann aber auch geändert werden. Dabei werden der MAC- und der PHY-Layer zurückgesetzt.

Als MAC-Adresse des fernen Terminals wird nach der Initialisierung die Broadcast-Adresse (FF-FF-FF-FF-FF-FF) verwendet, nach dem Empfang des ersten Frames wird dessen Source-MAC-Adresse eingesetzt.

3.2.6 Senden

Zum Senden von Frames sind zwei Kommandos vorgesehen, es können einzelne Frames oder Folgen gleicher Frames gesendet werden. Frames verschiedener Länge und verschiedener Inhalte sind vordefiniert und können ausgewählt werden.

Befehl	Länge(o. Preamb)	Bytewert
SEND=0	64	0x00
SEND=1	64	0xFF
SEND=2	64	0xAA
SEND=3	64	Random
SEND=4	218	0x00
SEND=5	218	0xFF
SEND=6	218	0xAA
SEND=7	218	Random
SEND=8	1518	0x00
SEND=9	1518	0xFF
SEND=10	1518	0xAA
SEND=11	1518	Random
SEND=12	64	1 mal 0xFF

Tabelle 3: Die vordefinierten Frames

Es gibt Frames der minimalen und maximalen und einer mittleren Länge, jeweils mit den Inhalten 00, FF, AA oder zufälligen Werten in den Nutzdaten, und zusätzlich ein Frame mit nur einem Byte, das um Pad-Bytes ergänzt wird.

Für das Senden von Folgen können dieselben Frames eine vorgebbare Zahl gesendet werden.

3.2.7 Empfangen

Frames können vom MAC-Layer zu jeder Zeit entgegengenommen werden und werden mit einer Interrupt-Prozedur in einem Ringspeicher abgelegt. Die Frames werden auf die ersten 38 Bytes begrenzt, der Speicher kann 10 solche Frames fassen. Die Gesamtzahl der empfangenen Frames wird festgehalten. Der Inhalt des Speichers kann per Kommando ausgegeben werden, der Speicher wird danach gelöscht.

3.3 Versuchsaufbau

Für den Versuch werden das Versuchsmodell, ein Oszillograf und ein PC gebraucht. Der PC dient als Abschluss der Ethernet-Verbindung und als Bedienterminal für das Versuchsmodell, und gleichzeitig als dessen Stromversorgung.

3.3.1 Blockschaltbild

Bei dem Versuchsaufbau handelt es sich um eine Ethernet-Verbindung zwischen dem Versuchsmodell und einem PC. In diese Verbindung ist ein T-Stück eingeschleift, das die Ver-

bindung zum Oszillografen herstellt und für die Potentialtrennung sorgt. Am Oszillografen können so die Signale der beiden Richtungen dargestellt werden.

Das Versuchsmodell liefert eine Reihe von Signalen, die für die Triggerrung des Oszillografen verwendet werden können und direkt angeschlossen werden.

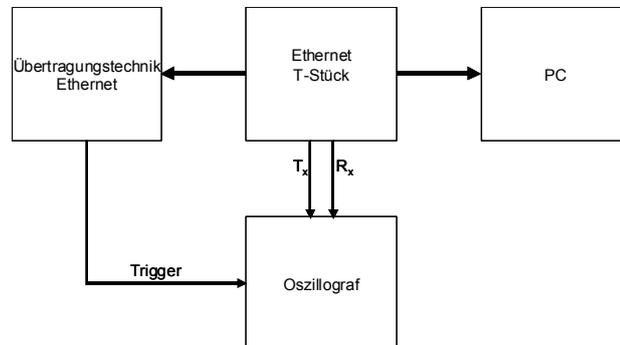


Abbildung 28: Das Blockschaltbild des Versuchs

Das T-Stück isoliert die unsymmetrischen Eingänge vom Oszillografen und vom Versuchsmodell.

3.3.2 Oszillograph

Dafür solle der übliche Labor-Oszillograf verwendet werden, ein Tektronix TDS 2014C. Mit seinen vier Kanälen ist die Darstellung der Ethernet-Signale und der Triggersignale gut möglich. Die Bandbreite von 100 MHz ist auch für 100 Mbit/s ausreichend.

3.3.3 PC

Der PC dient zum Abschluss der Ethernet-Verbindung und als Bedienterminal für das Versuchsmodell. Er braucht eine Netzwerkkarte, die 10 und 100 Mbit/s unterstützt. Die Netzwerkkarte soll auf Autonegotiation eingestellt werden („Speed&Duplex“).

Für die Kommunikation mit dem Versuchsmodell wird ein serielles Protokoll über USB gefahren. Das Versuchsmodell hat dazu einen Chip von Silicon Labs (CP210x USB-to-UART), dessen Treiber auf dem PC installiert sein muss [4]. Über die USB-Schnittstelle wird das Versuchsmodell auch mit Strom versorgt.

Das Terminal-Programm (z.B. HyperTerminal oder HTerm) muss ebenfalls installiert sein. Die Nummer des COM-Ports zum Versuchsmodell kann mit dem Gerätemanager bestimmt werden.

Für die Überwachung der Ethernet-Verbindung ist der Wireshark recht hilfreich, aber nicht unbedingt notwendig. Unter „Enabled Protocols“ sollte nur Ethernet ausgewählt werden, da sonst eine unnötige Interpretation der höheren Protokolle versucht wird.

Der PC wird versuchen, mit seinem IP-Protocol-Stack Verbindungen aufzubauen, die aber natürlich vom Versuchsmodell nicht bedient werden. Diese Versuche haben eher zufällige Ethernet-Frames zur Folge. Um gezielt Frames vom PC aus zu senden reicht es, z.B. eine Verbindung zu einem Netzlaufwerk zu versuchen. Dann werden direkt Frames gesendet, ihr Inhalt ist für den Versuch aber überhaupt nicht wichtig.

4 Die Bedienung des Versuchsmodells

Das Versuchsmodell wird über das Terminal-Programm des PCs bedient. Auf dem Versuchsmodell sind verschiedene LEDs, die neben der Spannungsversorgung verschiedene Ereignisse anzeigen:

RCV	Zeigt den Empfang eines Frames an
SND	Ein Frame wird gesendet
COL	Eine Kollision ist aufgetreten
Busy	Ein Kommando vom PC wird ausgeführt

Die Leuchtdauer der LEDs wird künstlich verlängert, damit sie bei den kurzen Ereignissen auch wahrnehmbar werden.

4.1 Befehlssyntax

Die Befehle folgen einer ganz einfachen Syntax:

Ein Kommando besteht aus vier Buchstaben und kann keinen, einen oder zwei Parameter haben. Das Kommando wird mit der RETURN-Taste („<ret>“) abgeschlossen.

Der erste Parameter wird von dem Kommando-Namen durch ein Gleichheitszeichen getrennt, vor dem zweiten Parameter steht ein Komma. Die Parameter können Dezimalzahlen oder Hexadezimalzahlen sein. Hexadezimalzahlen müssen den Prefix „0x“ oder „0X“ haben. Grundsätzlich können alle Buchstaben, auch in Kommando-Namen groß oder klein geschrieben werden.

Beispiele:

```
PPHY<ret>  
DUPL=1<ret>  
SREG=1,0x80fF<ret>
```

Zur Bedienungserleichterung bei immer wiederkehrenden Kommandos kann durch einfaches Drücken der RETURN-Taste das vorherige Kommando wiederholt werden.

4.2 Befehlsüberblick

Der Interpreter hat eine einfache Hilfe-Funktion die einen Überblick über alle Kommandos und deren Parameter gibt. Diese Funktion wird mit `HELP` aufgerufen.

Kommando	Bedeutung	1. Parameter		2. Parameter	
		Bedeutung	Werte	Bedeutung	Werte
ADFF	Advertise Fast Ethernet Full Duplex	De-/Aktivieren	0...1		
ADFH	Advertise Fast Ethernet Half Duplex	De-/Aktivieren	0...1		
ADSF	Advertise 10 Mbit/s Full Duplex	De-/Aktivieren	0...1		
ADSH	Advertise 10 Mbit/s Half Duplex	De-/Aktivieren	0...1		
AUTN	Auto Negotiation	De-/Aktivieren	0...1		
DUPL	Duplex Mode	De-/Aktivieren	0...1		
FAST	Fast Ethernet (100 Mbit/s)	De-/Aktivieren	0...1		
HELP	Print help text				
MDXE	Enable Auto MDIX	De-/Aktivieren	0...1		
MDXF	Force MDIX	De-/Aktivieren	0...1		
PMAC	Print MAC addresses				
PPHY	Print PHY Registers				
PRCV	Print Received Frames				
PSIG	Print Signal Selection				
RNEG	Restart Auto Negotiation				
SEND	Send a Message	Message Typ	0...9		
SENS	Send Sequence	Message Typ	0...9	Zahl der Messages	0...1000
SIGN	Signal Selection	Ausgang	0...2	Signal	0...7
VERS	Show Program Version				

Abbildung 29: Überblick über die wichtigsten Befehle

Neben diesen Befehlen gibt es noch eine Reihe weiterer, die aber für den Versuch nicht notwendig sind.

4.3 Beschreibung der Befehle

4.3.1 ADxx Advertise

Dieser Befehl ändert die unterstützten Funktionen in dem Link-Codewort, das mit den FLPs bei der Autonegotiation versendet wird. In diese Gruppe gehören Befehle

- ADFF Advertise Fast Ethernet Full-Duplex
- ADFH Advertise Fast Ethernet Half-Duplex
- ADSF Advertise 10 Mbit/s Full-Duplex
- ADSH Advertise 10 Mbit/s Half-Duplex

Diese Befehle haben einen Parameter, der entweder 1 für die Aktivierung oder 0 für die Deaktivierung sein kann. Nach der Ausführung des Befehls wird eine Meldung angezeigt. Diese Einstellungen haben keinen Einfluss auf die eigene Betriebsweise.

```
Cmd: adff=0
ADFF succesful
```

4.3.2 AUTN Auto Negotiation

Mit diesem Befehl kann die Autonegotiation ein- oder ausgeschaltet werden. Bei ausgeschalteter Autonegotiation reagiert das eigene Terminal nicht auf die von der anderen Seite angebotenen Funktionen, sondern nimmt nur die vordefinierten.

```
Cmd: autn=1
AUTN succesful
```

4.3.3 DUPL Duplex-Modus

Zulassen des Duplex-Modus. Nur wenn Autonegotiation deaktiviert ist, ist der damit eingestellte Modus zwingend.

```
Cmd: dupl=0
DUPL succesful
```

4.3.4 FAST Fast Ethernet (100 Mbit/s)

Umschalten zwischen 100 Mbit/s und 10 Mbit/s. Nur wenn Autonegotiation deaktiviert ist, ist der damit eingestellte Modus zwingend.

```
Cmd: fast=0
FAST succesful
```

4.3.5 HELP Print help text

Druckt den Hilfetext aus.

```
Cmd: HELP
Commands in upper or lower case
Equal sign between command and parameters, comma between parameters
Parameters as decimal or hexadecimal numbers, the latter with prefix 0X or 0x
Send command by pressing the return key <ret>
Examples: PPHY<ret>  DUPL=1<ret>  SREG=1,0x80fF<ret>
```

```
ADFF  Advertise Fast Ethernet Full Duplex   Par1 Range 0...1
ADFH  Advertise Fast Ethernet Half Duplex  Par1 Range 0...1
ADSF  Advertise 10 Mbit/s Full Duplex      Par1 Range 0...1
ADSH  Advertise 10 Mbit/s Half Duplex      Par1 Range 0...1
AUTN  Auto Negotiation                    Par1 Range 0...1
COLT  Collision Test                      Par1 Range 0...1
DUPL  Duplex Mode                         Par1 Range 0...1
FAST  Fast Ethernet (100 Mbit/s)         Par1 Range 0...1
HELP  Print help text                    No Parameters
LOOP  Loopback in PHY                    Par1 Range 0...1
MDXE  Enable Auto MDIX                    Par1 Range 0...1
MDXF  Force MDIX                          Par1 Range 0...1
PMAC  Print MAC addresses                 No Parameters
PPHY  Print PHY Registers                 No Parameters
PRCV  Print Received Frames               No Parameters
PREG  Print all Registers                 No Parameters
PSIG  Print Signal Selection              No Parameters
REST  Restart PHY                         No Parameters
RNEG  Restart Auto Negotiation            No Parameters
SEND  Send a Message                      Par1 Range 0...12
SENS  Send Sequence                      Par1 Range 0...12  Par2 Range 0...1000
SIGN  Signal Selection                    Par1 Range 0...2   Par2 Range 0...14
SMAC  Set local MAC                       Par1 Range 0...16777215  Par2 Range 0...16777215
SREG  Set Register                        Par1 Range 0...29  Par2 Range 0...65535
VERS  Show Program Version                No Parameters
```

Signal values and names for SIGN Par2

```
0: No Signal      1: MAC_TX_CLK    2: MAC_TX_EN     3: MAC_RX_CLK
4: MAC_RX_DV     5: MAC_RX_ER    6: MAC_COL       7: MAC_CR5
8: TX_EN_CLK     9: RX_DV_CLK   10: TX_ENRX_DV  11: TXD(0)
12: TXD(1)      13: RXD(0)     14: RXD(1)
```

4.3.6 MDXE Enable Auto MDIX

Bestimmt, ob die Auto-MDI-X-Funktion genutzt wird, ob bei vertauschten Sende- und Empfangs-Paaren eine Umschaltung vorgenommen wird. Wenn diese Funktion nicht aktiviert ist, wird nur auf den vorgegebenen Leitungen gearbeitet.

```
Cmd: mdxe=0
MDXE succesful
```

4.3.7 MDXF Force MDIX

Damit werden Sende- und Empfangsanschlüsse manuell vertauscht.

```
Cmd: mdxf=1
MDXF succesful
```

4.3.8 PMAC Print MAC addresses

Gibt die lokale MAC-Adresse aus und, wenn bekannt auch die der Gegenstation.

```
Cmd: pmac
Local MAC 00-0A-35-01-02-03
Remote MAC 00-1D-72-00-09-E6
```

4.3.9 PPHY Print PHY Registers

Gibt die Inhalte der wichtigen PHY-Register aus.

```
Cmd: PPHY
Read all PHY registers
-----
NEW
BMCR: LOOP=0 S100=0 AUTN=0 DUPL=0 COLT=0
BMSR: T4 =0 100F=1 100H=1 10BF=1 10BH=1          AUCP=0
      REMF=0 AUTN=1 LNKS=1
ANAR:      REMF=0 ASPA=0 PAUS=0 T4 =0 100F=0 100H=0
      10BF=0 10BH=0
ANLP: ACKR=1 REMF=0 ASPA=0 PAUS=1 T4 =0 100F=1 100H=1
      10BF=1 10BH=1
PHYS: MDIX=0 RCVE=0 POLA=0 FCS =0 SIGD=0 DESL=0 PRCV=0
      REMF=0 JABD=0 AUCP=0 LOOP=0 DUPL=0 S10B=1 LNKS=1
COUN: FCSR= 0 RECR= 0
PHYC: MDXE=0 MDXF=0
```

Reg	Bit	Bedeutung
BMCR		Basic Mode Control Register
	LOOP	
	S100	Speed 100 Mbit/s
	AUTN	Auto Negotiation
	DUPL	Duplex
	COLT	Collision Test

BMSR	Basic Mode Status Register
100F	100BASE-TX Full-Duplex
100H	100BASE-TX Half-Duplex
10BF	10BASE-T Full-Duplex
10BH	10BASE-T Half-Duplex
AUCP	AUtonegotiation ComPlete.
REMF	REMOte Fault, entweder von der anderen Seite gemeldet oder selbst erkannt. Ist ein Fehler aufgetreten, wird hier eine 1 angezeigt, bis das Register gelesen wird
AUTN	AUTO Negotiation ability
LNKS	LiNK Status. Mit 1 ist der Link aktiv. Hier bleibt der letzte Status bis zum Lesen stehen.
ANAR	AutoNegotiation Advertisement Register
REMF	REmote Fault meldet der anderen Seite einen Fehler
ASPA	ASsymetric PAuse dient zur Flusskontrolle, vom PHY nicht unterstützt
PAUS	PAuse dient zur Flusskontrolle, vom PHY nicht unterstützt
T4	
100F	100BASE-TX Full-Duplex
100H	100BASE-TX Half-Duplex
10BF	10BASE-T Full-Duplex
10BH	10BASE-T Half-Duplex
ANLP	AutoNegotiation Link Partner ability register
ACKR	ACKnowledgement Received wird der korrekte Empfang der Autonegotiation Messages bestätigt.
REMF	REmote Fault meldet der anderen Seite einen Fehler
ASPA	ASsymetric PAuse dient zur Flusskontrolle, vom PHY nicht unterstützt
PAUS	PAUS dient zur Flusskontrolle, vom PHY nicht unterstützt
T4	
100F	100BASE-TX Full-Duplex
100H	100BASE-TX Half-Duplex
10BF	10BASE-T Full-Duplex
10BH	10BASE-T Half-Duplex
PHYS	PHYSical Status register und gibt den aktuellen Status an. Viele Werte sind identisch mit dem BMSR
MDIX	gibt an, ob Sende- und Empfangspaar vertauscht sind (entweder durch Einstellung oder durch Erkennung)

RCVE	ReCeIve Error, wird nach dem Lesen zurückgesetzt
POLA	gibt an, ob die beiden Adern der Empfangsrichtung vorschriftsmäßig abgeschlossen (0) oder vertauscht sind (1).
FCS	False Carrier Sense wurde erkannt, wird beim Lesen zurückgesetzt.
SIGD	unconditional SIGnal Detect bei 100BASE-TX
DESL	DEScrambler Lock bei 100BASE-TX
PRCV	PageReCeIved eine weitere Seite während der Autonegotiation
REMF	REMOte Fault
JABD	JABer Detected
AUCP	AUtonegotiation Complete
LOOP	Loop
DUPL	Duplex
S10B	bedeutet Speed ist gleich 10 Mbit/s (1) oder 100 Mbit/s (0)
LNKS	gibt den aktuellen LiNK Status an.
COUN	
FCSR	False Carrier Sense event counter Register, wird beim Lesen zurückgesetzt.
RECR	Receiver Error Counter Register, wird beim Lesen zurückgesetzt.
PHYC	PHYSical Control register
MDXE	Auto-MDI-X enabled
MDXF	MDI-X forced

4.3.10 PRCV Print Received Frames

Gibt die im Ringspeicher abgelegten empfangenen Frames mit den ersten 38 Byte aus. Die Frames werden mit der tatsächlichen Länge angegeben, die MAC-Adressen werden strukturiert angezeigt.

```
cmd: prcv
Total frames received: 75
Len+ Dest MAC + Orig MAC + TL +
260 FFFFFFFF001D720009E6 0800 45 00 00 F2 13 DC 00 00 80 11 A1 C0 C0 A8 01 0F C0 A8 01 FF 00 8A 00 8A
253 FFFFFFFF001D720009E6 0800 45 00 00 EB 13 DD 00 00 80 11 A1 C6 C0 A8 01 0F C0 A8 01 FF 00 8A 00 8A
260 FFFFFFFF001D720009E6 0800 45 00 00 F2 13 DE 00 00 80 11 A1 BE C0 A8 01 0F C0 A8 01 FF 00 8A 00 8A
```

Die Gesamtzahl der seit dem Start des Programms empfangenen Frames wird ebenfalls angezeigt. Nach dem Auslesen wird der Ringspeicher gelöscht.

4.3.11 PSIG Print Signal Selection

Mit diesem Befehl werden die Signale ausgegeben, die mit dem Befehl SIGN den BNC-Buchsen zugeordnet wurden.

```
cmd: psig
Signal assignments:
```

```
MAC_TX_CLK assigned to output S0
MAC_TX_EN assigned to output S1
MAC_RX_CLK assigned to output S2
```

4.3.12 RNEG Restart Auto Negotiation

Die Autonegotiation wird nochmals durchgeführt.

```
Cmd: rneg
RNEG succesful
```

4.3.13 SEND Send a Message

Ein Frame von zehn vordefinierten wird gesendet. Die vordefinierten Frames haben die Nummern von 0 bis 10. Die Länge der Nutzdatenfelder ist 46, 200 und 1500, die Bytes in diesen Feldern können die hexadezimalen Werte 00, FF oder AA haben. Der Frame 9 hat eine Nutzdatenlänge von einem Byte des Wertes FF, das Feld wird mit Pad-Bytes (00) auf 46 erweitert.

Preamble	Dest MAC	Orig MAC	TL	Payload	FCS	ohne PreAmb	Gesamt Bytes	Gesamt Bits	Ts[μs] 10 Mbit/s	Ts[μs] 100 Mbit/s
8	6	6	2	46	4	64	72	576	57,6	5,76
8	6	6	2	200	4	218	226	1808	180,8	18,08
8	6	6	2	1500	4	1518	1526	12208	1220,8	122,08

```
SEND=0
Cmd: Frames sent: 75 length: 46 value: 00

Cmd: SEND=1
Frames sent: 76 length: 46 value: FF

Cmd: SEND=2
Frames sent: 77 length: 46 value: AA

Cmd: SEND=3
Frames sent: 78 length: 46 value: Random

Cmd: SEND=4
Frames sent: 79 length: 200 value: 00

Cmd: SEND=5
Frames sent: 80 length: 200 value: FF

Cmd: SEND=6
Frames sent: 81 length: 200 value: AA

Cmd: SEND=7
Frames sent: 82 length: 200 value: Random

Cmd: SEND=8
Frames sent: 83 length: 1500 value: 00

Cmd: SEND=9
Frames sent: 84 length: 1500 value: FF

Cmd: SEND=10
Frames sent: 85 length: 1500 value: AA

Cmd: SEND=11
Frames sent: 86 length: 1500 value: Random
```

```
Cmd: SEND=12
Frames sent: 87 length: 1 value: FF
```

4.3.14 SENS Send Sequence

Mit diesem Befehl kann ein vordefinierter Frame mehrmals gesendet werden.

```
Cmd: sens=1,5
Sequence of 5 sent, Total frames sent: 15
SENS succesful
```

```
Cmd: sens=5,3
Sequence of 3 sent, Total frames sent: 18
SENS succesful
```

```
Cmd: sens=7,500
Sequence of 500 sent, Total frames sent: 518
SENS succesful
```

4.3.15 SIGN Signal Selection

Die in Kapitel 3.1.6 beschriebenen Triggersignale können über einen Multiplexer einer der drei BNC-Buchsen beliebig zugeordnet werden. Die BNC-Buchsen werden von 0 bis 2 durchnummeriert, die Signale von 0 bis 14 entsprechend der Tabelle 4.

Parval	Signal
0	No Signal
1	MAC_TX_CLK
2	MAC_TX_EN
3	MAC_RX_CLK
4	MAC_RX_DV
5	MAC_RX_ER
6	MAC_COL
7	MAC_CRS
8	TX_CLK and TX_EN
9	RX_CLK and RX_DV
10	TX_EN or RX_DV
11	TXD(0)
12	TXD(1)
13	RXD(0)
14	RXD(1)

Tabelle 4: Die Nummern der Triggersignale

```
Cmd: sign=0,4
MAC_RX_DV assigned to output S0
```

```
Cmd: sign=1,5
MAC_RX_ER assigned to output S1
```

5 Versuchsdurchführung

5.1 Vorbereitung des Versuchs

Für den Versuch brauchen Sie die folgenden Teile:

- Versuchsmodell „Übertragungstechnik Ethernet“
- 1 kurzes LAN-Kabel
- 1 längeres LAN-Kabel
- Labor-Oszillograf
- Labor-PC
- USB-Kabel
- 4 BNC-Kabel
- 4 Feed-Throughs

Die Verbindungen werden nach Abbildung 30 hergestellt. Schalten Sie die Geräte ein. Die Power-LED des Versuchsmodells soll leuchten.

Alle Verbindungen zum Oszillografen erfolgen mit BNC-Messkabeln, und 50 Ω -Feedthroughs am Oszillograf-Eingang. Damit werden Reflexionen bei falschen Leitungsabschlüssen vermieden. Achten Sie bei der Einstellung der Kanäle mit den beiden Ethernet-Leitungen, dass im Versuchsmodell schon ein Spannungsteiler 1:10 eingebaut ist. Dies ist bei den Signalausgängen S_0 , S_1 und S_2 nicht der Fall.

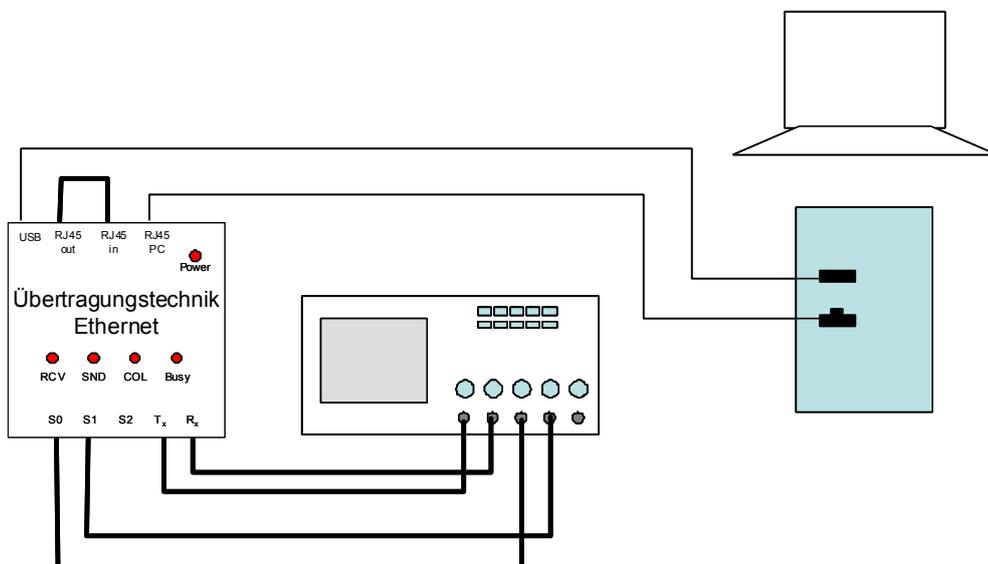


Abbildung 30: Die Verbindungen für den Versuch

Suchen Sie im Gerätemanager des PCs nach dem COM-Port, der der USB to UART Bridge CP210x von Silicon Labs zugeordnet ist. Starten Sie Das Terminalprogramm und öffnen

Sie eine Verbindung mit diesem COM-Port. Die Einstellungen müssen 115200 Baud, 8 Datenbits, kein Parity, 1 Stopbit und ohne Flußkontrolle sein.

Der PC soll jetzt mit dem Versuchsmodell kommunizieren können. Da sich das Versuchsmodell schon initialisiert hat, sehen Sie die Meldung mit der Versionsnummer nicht. Drücken Sie die Return-Taste, dann werden Sie die Busy-LED leuchten sehen, der Interpreter führt den gespeicherten Befehl „VERS“ und meldet sich mit dem Startbild und der Versionsanzeige. Geben Sie jetzt einen Befehl ein, z.B. PREG, dann werden die Register des PHY ausgegeben.

Auf dem PC können Sie jetzt den Wireshark starten und eine Aufzeichnung („Capture“) am Ethernet-Interface beginnen. Wenn Sie unter „Analyse“ bei „Enabled Protocols“ nur „Ethernet“ auswählen, dann wird gar nicht versucht höhere Protokolle, die ja nicht vorhanden sind, zu dekodieren. Überprüfen Sie gelegentlich den Empfang im Wireshark, wenn Sie Frames gesendet haben.

5.2 Versuche mit 10BASE-T

Das Übertragungsverfahren bei 10BASE-T kann sehr einfach auf dem Oszillografen dargestellt werden, weil – abgesehen von den Link Pulses – nur während der Aussendung eines Frames die Spannung auf dem Link ungleich 0 ist. Und mit dem Manchester-Code wird ein leicht zu dekodierender Leitungscode eingesetzt.

5.2.1 Anfang eines Frames bei 10BASE-T

Das Versuchsmodell ist nach dem Start auf die Default-Werte eingestellt, so dass Auto-MDI-X und Autonegotiation unterdrückt sind, und mit 10 Mbit/s gearbeitet wird. Später können Sie diesen Zustand mit den folgenden Befehlen einstellen, beachten Sie dabei die Einstellung auf Vollduplex (siehe Kapitel 3.1.4):

```
AUTN=0
MDXE=0
DUPL=1
FAST=0
```

Es sollte jetzt ein Link mit 10 Mbit/s aufgebaut werden. Sie können dies mit dem Befehl PPHY feststellen: Unter BMSR sollte der vorherige Linkstatus mit 0 angegeben sein (LNKS=0), unter PHYS der aktuelle Linkstatus mit 1 (LNKS=1).

Damit der Frame gut auf dem Oszillografen interpretiert werden kann sollen jetzt noch zwei Signale dargestellt werden, der Takt des Senders (MAC_TX_CLK) und der Beginn des Frames (MAC_TX_EN). Diese beiden Signale werden mit den BNC-Buchsen S₀ und S₁ verbunden:

```
Cmd: sign=0,1
MAC_TX_CLK assigned to output S0

Cmd: sign=1,2
MAC_TX_EN assigned to output S1
```

Wenn Sie auf das Signal MAC_TX_EN im Kanal 4 triggern, wird der Anfang eines Frames aufgezeichnet. Bei den beiden Triggersignalen reicht eine Einstellung von 5 V/div. Bei dem Takt handelt es sich um den Takt am MDI, der ¼ der Übertragungsgeschwindigkeit beträgt. Der Zusammenhang zwischen Takt und Signal auf der Leitung ist in Abbildung 21 dargestellt. Stellen Sie „Acquire“ wieder auf „Sample“.

Senden Sie jetzt mit `SEND=1` einen Frame und nehmen Sie das Oszillogramm auf. Beschreiben Sie das Signal und dekodieren Sie die ersten 4 Symbole. Versuchen Sie die gesamte Preamble des Frames zu erkennen und den Anfang der MAC-Adresse.

5.2.2 Ende eines Frames bei 10BASE-T

Triggern Sie jetzt auf die fallende Flanke von Kanal 4 um das Ende eines Frames zu sehen. Senden Sie wieder ein Frame. Beschreiben Sie das Ende des Frames und vergleichen sie es mit Abbildung 22.

5.2.3 Gesamter Frame bei 10BASE-T

Jetzt sollen Sie die Länge der verschiedenen Frames messen, sie müssen also jeweils einen ganzen Frame auf dem Oszillografen darstellen. Senden Sie dabei drei Frames mit verschiedenen Längen. Bei sehr langen Zeiten ist die Darstellung des sich schnell ändernden Signals auf dem Ethernet (und auch die des Taktes) nicht mehr aussagekräftig. Auf jeden Fall können Sie die Länge des TX_EN-Signals sehen. Mit „Acquire“ auf „Peak“ gestellt, wird nur noch der Umriss des Ethernet-Signals gezeigt.

Alle gesendeten Frames sollten Sie mit Wireshark auf dem PC sehen können.

5.2.4 Empfangener Frame bei 10BASE-T

Mit diesem Versuch soll ein Frame vom PC betrachtet werden und mit den Frames vom Versuchsmodell verglichen werden. Jetzt sind andere Triggersignale erforderlich:

```
Cmd: sign=0,3
MAC_RX_CLK assigned to output S0
```

```
Cmd: sign=1,4
MAC_RX_DV assigned to output S1
```

Der PC muss jetzt veranlasst werden, ein Frame zu senden. Das kann man erreichen, indem versucht wird ein Netzlaufwerk zu verbinden. Welches Frame gesendet wird, ist für unsere Versuche nicht wichtig. Triggern Sie auf das Signal MAC_RX_DV. Sie sehen, wie sich der Takt ändert. Vergleichen Sie dazu die Abbildung 23. Sie können auch das Signal MAC_RX_CRG statt des Takts auf dem Oszillografen darstellen. Dann kann man erkennen, wann der PHY nach dem Erkennen der Belegung den Anfang eines Frames ableitet. Dazu geben Sie ein:

```
Cmd: sign=0,7
MAC_CRG assigned to output S0
```

Beschreiben Sie das empfangene Signal und vergleichen es mit dem gesendeten.

5.2.5 Darstellung von Augendiagrammen

Augendiagramme werden erzeugt, indem die übertragenen Bits – möglichst mit zufälligen Werten auf dem Oszillografen übereinander geschrieben werden. Getriggert wird dabei mit dem

Bittakt. Das Versuchsmodell stellt einen Takt für jeweils 4 Bits zur Verfügung. Allerdings ist es nicht möglich, die Leitung kontinuierlich zu belegen. Senden Sie Folgen von Frames, am besten mit zufälligen Bytes. Zum Triggern bei 10 Mbit/s stellt das Versuchsmodell ein Ausgangssignal bereit, eine Und-Verknüpfung des Taktes mit TX_EN. Damit wird das laufende Triggern des Oszillografen, auch wenn keine Frames gesendet werden, vermieden.

Triggern Sie auf die fallende Flanke des Taktes, stellen Sie den Kanal 1 dar und stellen Sie die „Persistence“ unter „Display“ auf „Infinite“. Dann senden Sie mehrmals hintereinander Folgen von 1000 Frames mit zufälligem Inhalt. Das Augendiagramm baut sich langsam auf. Erklären Sie es.

```
Cmd: SIGN=0,8
TX_EN_CLK assigned to output S0
sens=11,1000
```

5.2.6 Untersuchung des Spektrums

Die Signale auf der Leitung sollen jetzt im Frequenzbereich untersucht werden. Der Oszillograf wird mit „Math“ auf Spektrumsanzeige eingestellt. Wählen Sie Kanal 1, „FFT“ und „Hanning“.

Senden Sie jetzt die langen Frames mit verschiedenen Inhalten. Beschreiben Sie die Spektren.

5.3 Versuche mit 100BASE-TX

Bei diesem Verfahren ist die Leitungscodierung sehr viel aufwendiger als bei 10BASE-T, aber vor allem: es wird gescrambled! Dadurch können Inhalte auf der Leitung nicht mehr erkannt werden, es sei denn man descrambled wieder, aber das ist mit den einfachen Mitteln eines Oszillografen nicht möglich. Deshalb werden wir uns auf den MLT-3 Codes beschränken.

Die Erzeugung des Sendesignals bei 100BASE-TX in dem PHY-Baustein ist in Abbildung 31 gezeigt.

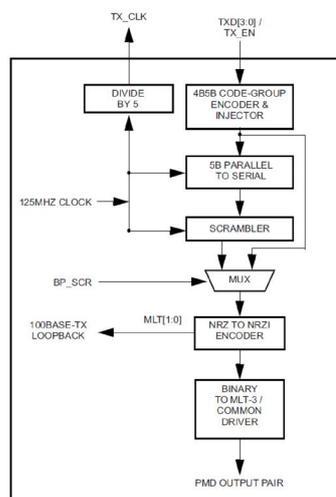


Abbildung 31: Die Leitungscodierung bei 100BASE-TX im PHY-Baustein

5.3.1 Gesendete Signale

Im Gegensatz zu 10BASE-T wird bei 100BASE-TX laufend gesendet, nämlich das IDLE-Pattern „I“ (1111 1111). Allerdings wird auch dieses gescrambled und kann auf der Leitung nicht gesehen werden.

Stellen sie jetzt den 100 Mbit/s- Modus ein:

```
Cmd: FAST=1
FAST successful
```

Der Sendetakt soll ebenfalls auf dem Oszilloskop dargestellt werden:

```
Cmd: sign=0,1
MAC_TX_CLK assigned to output S0
```

Der Oszilloskop muss so eingestellt werden, dass sie die vom Versuchsmodell gesendeten Signale erkennen können. Beschreiben Sie die Signale.

5.3.2 Dekodieren der gesendeten MLT-3-Zustände

Versuchen Sie, das im vorhergegangenen Versuch aufgenommene MLT-3-Signal zu dekodieren. Am einfachsten geht das am Oszilloskop, da Sie das aufgenommene Oszillogramm auf der Zeitachse verschieben und strecken oder stauchen können. Nehmen Sie die Cursor zu Hilfe.

5.3.3 Augendiagramm

Nehmen Sie jetzt ein Augendiagramm der 100BASE-TX-Signale auf. Stellen Sie den Oszilloskop entsprechend ein. Im Gegensatz zu 10 BASE-T brauchen Sie jetzt keine Frames zu senden, da ja laufend das IDLE-Pattern übertragen wird. Und durch das Scramblen erhalten wir ja schon eine pseudo-zufällige Bitfolge. Interpretieren Sie das Diagramm.

5.3.4 Empfangene Signale

Jetzt können Sie noch die Empfangsseite betrachten, also das Signal vom PC. Für die Taktreferenz legen Sie den Empfangstakt auf den Kanal 3 des Oszilloskop.

```
Cmd: sign=0,3
MAC_RX_CLK assigned to output S0
Cmd: SIGN=1,4
MAC_RX_DV assigned to output S1
```

Beschreiben Sie das Signal. Sehen Sie irgendwelche Abweichungen zu dem gesendeten Signal?

5.3.5 Spektrum

Stellen Sie den Oszilloskop jetzt wieder auf die Darstellung des Spektrums ein und nehmen das Spektrum auf. Auch dieses Mal brauchen Sie nicht explizit Frames senden, sondern können das IDLE-Pattern verwenden. Durch das Scrambling unterscheidet sich dessen Spektrum nicht von dem der gesendeten Frames. Beschreiben Sie das Ergebnis.

5.4 Autonegotiation

In diesem Versuch soll die Übertragung der Fähigkeiten des eigenen Systems zur Gegenstelle mit Hilfe der Normal Link Pulses und der Fast Link Pulses untersucht werden. Die daraus abgeleitete Auswahl der Betriebsart wird aber hier nicht betrachtet.

5.4.1 Normal Link Puls

Zuerst sollen die Normal Link Pulses (auch Link Integrity Test Pulses genannt) untersucht werden, also die Signale, die im Ruhezustand eines Links gesendet werden. Entfernen Sie die Verbindung zum PC, dann schickt das Versuchsmodell NLPs und wartet auf den Empfang der NLPs von der Gegenstelle.

Das Versuchsmodell stellen Sie wieder mit den folgenden Befehlen ein, so dass Auto-MDI-X und Autonegotiation unterdrückt sind, und nur mit 10 Mbit/s halbduplex gearbeitet wird.

```
AUTN=0  
MDXE=0  
DUPL=0  
FAST=0
```

Die NLPs sind positive Impulse, triggern Sie also den Oszilloskopen so, dass Sie einen einzelnen NLP sehen können. Beschreiben Sie diesen Puls.

Die NLPs werden laufend wiederholt. Verstellen Sie Zeitbasis, dass Sie eine Folge von NLPs sehen können. Indem Sie die „Acquire“ des Oszilloskopen auf „Peak“ statt „Sample“ verstellen, lässt sich die Pulsfolge besser erkennen. Mit welcher Rate werden die NLPs gesendet?

5.4.2 Normal Link Pulse vom PC

Trennen Sie jetzt die Verbindung auf dem Versuchsmodell zwischen RJ45 out und RJ45 in und verbinden Sie RJ45 PC mit dem PC. Auf dem Kanal 2 sehen Sie jetzt die NLPs vom PC.

5.4.3 Einzelner Fast Link Pulse

Unterbrechen Sie die Verbindung zum PC, so dass kein Link aufgebaut ist, und das Versuchsmodell nur die Link Pulses schickt.

Nehmen Sie einen einzelnen Fast Link Pulse auf und vergleichen Sie ihn mit dem NLP.

5.4.4 FLP Burst

Jetzt soll ein ganzer Burst von FLPs aufgenommen und anschließend nach Abbildung 12, Abbildung 13 und Abbildung 14 dekodiert werden. Dafür arbeiten Sie am Besten wieder mit der

„Acquire“-Einstellung „Peak“. Sie können jetzt einige Einstellungen des Link Codeworts in den eigenen Advertise Messages ändern und die Auswirkung in dem FLP Burst sehen.

Schalten Sie die Autonegotiation aus, ändern Sie die Einstellungen, starten den Oszilloskop und schalten erst dann Autonegotiation wieder ein. Die Einstellungen können Sie mit PPHY unter ANAR überprüfen.

```
Cmd: autn=0
AUTN succesful
```

```
Cmd: adff=0
ADFF succesful
```

```
Cmd: autn=1
AUTN succesful
```

5.5 Kollisionen

Da der MAC-Layer des Versuchsmodells – unabhängig von den Einstellungen des PHY – grundsätzlich in Halbduplex arbeitet, könnten bei den bisherigen Untersuchungen Kollisionen schon aufgetreten sein. Da aber die Rate der gesendeten Frames auf beiden Seiten sehr klein ist, ist die Wahrscheinlichkeit dafür gering. Da aber das PHY auf Vollduplex eingestellt war, wurden der Empfang eines Frames während des Sendevorgangs gar nicht an den MAC-Layer gemeldet und somit nicht behandelt. Alle Frames wurden gesendet und empfangen. Der einzige Unterschied zur Vollduplex-Implementierung besteht darin, dass der MAC-Layer ein zu sendendes Frame solange zurückhält, bis ein gleichzeitiger Empfang abgeschlossen ist.

Um sicher Kollisionen herbeizuführen, werden Sende- und Empfangsrichtung des Versuchsmodells direkt miteinander verbunden. Sobald also ein Frame gesendet wird, kommt auch eines am Empfänger an. Wie der Sender sich dann verhält wird hier untersucht.

5.5.1 Kollision in einem Frame

Entfernen Sie die Verbindung zum PC und stecken Sie einen Schleifenstecker ein. Stellen Sie das PHY auf 10BASE-T und Halbduplex ein:

```
AUTN=0
MDXE=0
DUPL=0
FAST=0
```

Für die Darstellung auf dem Oszilloskop legen Sie jetzt noch die Signale TX_EN und COL an die BNC-Buchsen.

```
Cmd: sign=0,2
MAC_TX_EN assigned to output S0
```

```
Cmd: sign=1,6
MAC_COL assigned to output S1
```

Auf dem Oszilloskop sollen jetzt alle Kanäle dargestellt werden: Kanal1: T_x, Kanal2: R_x, Kanal3: TX_EN, Kanal4: COL. Die Zeitbasis wird so eingestellt, dass ein kurzes Frame dargestellt wird. Triggern Sie auf TX_EN und senden ein Frame. Was sehen Sie auf dem Oszilloskop?

5.5.2 Wiederholung nach Kollisionen

Wird vom Sender eine Kollision erkannt, so wird der Frame nach einer Wartezeit wiederholt. Die Wartezeit wird nach einem Algorithmus aus einer Zufallszahl bestimmt. Da in unserem Fall ja alle Sendeveruche in Kollisionen enden, können die Wiederholungen beobachtet werden. Stellen Sie den Oszillografen so ein, dass Sie mehrere Sendeveruche darstellen können.

Wie groß sind die Abstände? Wie viele Wiederholungen finden statt?

6 Literatur

- [1] Rech, J.; Ethernet; Heise 2014
- [2] Texas Instruments DP83848x PHYTER Mini / LS Single Port 10/100 MB/s Ethernet Transceiver (SNLS250E –MAY 2008–REVISED APRIL 2015)
- [3] Reichardt, J. Schwarz, B. VHDL-Synthese De Gruyter Oldenburg 2015
- [4] <http://www.silabs.com/products/mcu/pages/usbtouartbridgevcpcdrivers.aspx>
- [5] em.avnet.com/s6microboard
- [6] IEEE <http://www.ieee.org>
- [7] IEEE 802.3at-2009 (Power over Ethernet Erweiterungen)
- [8] AVNET MicroBoard
<http://www.em.avnet.com/en-us/design/drc/Pages/Xilinx-Spartan-6-FPGA-LX9-MicroBoard.aspx>

7 Abkürzungen

4B5T	4 Binary to 5 Binary Code
ATM	Asynchronous Transfer Mode
AUI	Attachment Unit Interface
AXTLK	Alien Cross Talk
CLK	Clock
COL	Collision
CRS	Carrier Sense
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
DEC	Digital Equipment Corporation
DIP	Dual In Line
DV	Data Valid
EIA	Electronic Industry Alliance
ER	Error
F/FTP	Foiled/Foiled Twisted Pair
FCS	Frame Check Sequence
FDDI	Fiber Distributed Data Interface
FEXT	Far End Cross Talk
FLP	Fast Link Pulse
FPGA	Field Programmable Gate Array
FTP	Foiled Twisted Pair
HP	Hewlett-Packard
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
JTAG	Joint Test Action Group
LAN	Local Area Network
LED	Light Emitting Diode
LIT	Link Integrity Test
LSA	Löt-, Schneid-, Abisolierfrei
LSB	Least Significant Bit
MAC	Media Access Control
MAU	Medium Attachment Unit
MDC	Management Data Clock
MDI	Media Dependant Interface
MDIO	Management Data Input Output MDIO
MDI-X	Media-Dependant Interface Crossover
MII	Media Independent Interface
MLT-3	Multi Level Transmission 3 States
MPLS	Multi Protocol Label Switching
MSB	Most Significant Bit

NEXT	Near End Cross Talk
NLP	Normal Link Pulse
NRZI	Non Return to Zero Inverted
OSI	Open System Interconnection
PC	Personal Computer
PCS	Physical Coding Sublayer
PD	Powered Device
PLS	Physical Layer Signaling
PMA	Physical Medium Attachment
PMD	Medium Dependant Sublayer
PoE	Power over Ethernet
PSE	Power Sourcing Equipment
RJ-45	Registered Jack 45 (8-polige Steckverbindung für LAN-Kabel)
RMII	Reduced MII
RX	Receiver
RXD	Received Data
S/UTP	Screened Unshielded Twisted Pair
SDH	Synchronous Digital Hierarchy
SF/FTP	Screened Foiled/Foiled Twisted Pair
SFD	Start of Frame Delimiter
STP	Shielded Twisted Pair
TCP	Transmission Control Protocol
TIA	Telecommunication Industry Association
TP	Twisted Pair
TX	Transmitter
TXD	Transmission Data
USB	Universal Serial Bus
UTP	Unshielded Twisted Pair
VLAN	Virtual LAN