



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Anwendungen 1 Ausarbeitung

Jan Jasper Salathé

Dynamische Konfiguration mit AVB im TTEthernet

Jan Jasper Salathé

Thema dieser Ausarbeitung

Dynamische Konfiguration mit AVB im TTEthernet

Stichworte

TTEthernet, AVB, Real Time Ethernet, Automotive-Echtzeitsysteme

Kurzzusammenfassung

Diese Ausarbeitung behandelt die Anforderungen an echtzeitfähige Kommunikation sowie die am weitesten verbreiteten Kommunikationsprotokolle und Bussysteme im Automotive Bereich. Es wird gezeigt, dass es in Zukunft einen Bedarf an zur Laufzeit dynamisch allozierbaren Netzwerkre-sourcen geben wird. Weiterhin wird ein Konzept zur dynamischen Allokation von Netzwerkre-sourcen zur Laufzeit im TTE mit AVB vorgestellt. Diese Ausarbeitung ist im Rahmen der Ver-anstaltung Anwendungen 1 des Master of Science Informatik entstanden und steht im Kontext der Vorbereitung auf die Masterthesis.

Jan Jasper Salathé

Title of the paper

Dynamic Configuration with AVB in TTEthernet

Keywords

TTEthernet, AVB, Real Time Ethernet, Automotive-Realtimesystems

Abstract

This paper treats the requirements of communication in realsystems and discusses the most common communicationprotocols and bussystems in the automotive field. It is shown, that dy-namically allocated networking resources will be required by some applications in the future and a concept of dynamically allocation in TTE with AVB is introduced. This paper has been written within the lecture Anwendungen 1 of the Master of Science in Computer Science and is a prepara-tion for the master thesis.

Inhaltsverzeichnis

1	Einleitung	4
2	Anforderungen an Echtzeitkommunikation	5
3	Feldbussysteme im Automobil	6
4	Ethernet im Automobil	8
5	Ziele	12
6	Risiken	13
7	Zusammenfassung	13
	Literatur	14

1 Einleitung

Automobile benötigen seit ihrer Entwicklung elektrische Systeme für den Betrieb. Anfangs handelte es sich lediglich um das Zündsystem des Motors. Später kamen elektrische Anlasser, Blinker, Scheinwerfer und Radios hinzu. Die Entwicklung, dass in Automobilen immer mehr elektrische Systeme integriert werden, setzt sich bis heute fort [5]. So befinden sich aktuell in Fahrzeugen der Oberklasse bis zu 70 elektrische Steuergeräte (Electronic Control Systems, ECUs). [22]

Zwischen den ECUs findet ein Informationsaustausch statt, der bei bestimmten Systemen Echtzeitanforderungen unterliegt. Echtzeitanforderungen spielen besonders bei Systemen im Antriebstrang und Bremssystem eine Rolle. Abhängig von der benötigten Bandbreite und Anforderungen an die Echtzeitfähigkeit kommen unterschiedliche Bussysteme zum Einsatz. Zum Einhalten der Echtzeitanforderungen muss a priori ein Timing für die Bussysteme erstellt werden. Da die einzelnen Bussysteme des Bordnetzes auch untereinander über Gateways Informationen austauschen, stellt dies mittlerweile eine umfangreiche Aufgabe dar.

Ein weiterer Aspekt des Informationsaustausches stellen Fahrassistenz- und Entertainmentsysteme dar. Fahrassistenzsysteme warnen den Fahrer beispielsweise vor dem Verlassen der Fahrbahn oder informieren ihn über Geschwindigkeitsbegrenzungen. Für diese Anwendungen werden Bilddaten mit einer hohen Übertragungsrate von einem Bildsensor zu einer ECU übertragen. Entertainmentsysteme unterhalten die Fahrgäste mit multimedialen Inhalten, wodurch ebenfalls hohe Anforderungen an die Bandbreite entstehen. Zusätzlich unterliegen auch der Informationsaustausch dieser Anwendungen Echtzeitanforderungen. Diese steigenden Anforderungen an die Bandbreite der Bussysteme werden aktuell in Fahrzeugen durch serielle Punkt-zu-Punkt Verbindungen (Low Voltage Differential Signaling, LVDS) bedient [26]. Durch den Einsatz von Ethernet lässt sich die benötigte Bandbreite bereitstellen und die Kosten solcher Systeme reduzieren. [26]

Von Jones [14] werden Entwicklungen skizziert, die auf Applikationen im Automobil hinauslaufen, wie sie auf Smartphones verfügbar sind. Daraus resultieren durch den Nutzer dynamisch anpassbare Systemkomponenten, auf denen die Software während des Betriebs erweitert werden kann. Eine Integration von solchen erweiterbaren Softwarekomponenten in das Bordnetz erfordert eine dynamische Allokation von Netzwerkressourcen zur Laufzeit. Weiterhin unterliegt der Informationsaustausch dieser Softwarekomponenten gegebenenfalls Echtzeitanforderungen. Protokolle wie Time Triggered Ethernet (TTE) mit a priori bestimmtem Timing lassen eine dynamische Allokation von Netzwerkressourcen jedoch nicht zu. Dies lässt jedoch mit dem Audio/Video Bridgin (AVB) Standard erreichen.

Diese Ausarbeitung stellt einerseits eine Übersicht über die vorhandenen Protokolle zur Echtzeitkommunikation im Automotiv Bereich dar. Andererseits wird darauf eingegangen, warum eine Kombination von TTE und AVB besonders interessant ist. Im folgenden werden erst die Anforderungen an echtzeitfähige Kommunikation dargelegt. Danach wird in Abschnitt 3 ein Überblick

über die vorhandenen Feldbussysteme gegeben. Bussysteme auf Basis von Ethernet werden im Abschnitt 4 betrachtet. Danach werden die Ziele bis zur Masterarbeit in Abschnitt 5 erläutert und in Abschnitt 6 auf mögliche Risiken und Probleme eingegangen. Zuletzt wird diese Ausarbeitung in Abschnitt 7 zusammengefasst.

2 Anforderungen an Echtzeitkommunikation

Echtzeitsysteme sind immer Teil eines Gesamtsystems, dem physikalische Prozesse zu Grunde liegen. Aus dieser Tatsache ergeben sich spezielle Anforderungen. Werden diese Anforderungen nicht eingehalten, so kann das korrekte Funktionieren des Gesamtsystems nicht garantiert werden. Dies wirkt sich auch auf die Anforderungen an die Kommunikation eines Echtzeitsystems aus [17]. Auf die unterschiedlichen Anforderungen wird im Folgenden eingegangen.

Ende-zu-Ende Laufzeit: Ein echtzeitfähiges Kommunikationssystem muss eine in einem bestimmten Rahmen definierte Laufzeit der Pakete zwischen zwei Endpunkten garantieren. Dabei darf die Laufzeit lediglich in einem begrenzten Umfang schwanken (Jitter). Das nicht Einhalten der a priori definierten Laufzeit gilt als Fehler. Die Laufzeit von Paketen muss auch bei Multicast Übertragungen eingehalten werden, alle angesprochenen Endpunkte müssen das Paket zu innerhalb eines definierten Zeitrahmens erhalten. Zusätzlich muss die Laufzeit den durch das zugrunde liegende System definierten Reaktionszeiten angepasst sein. [17]

Flexibilität: Ein echtzeitfähiges Kommunikationssystem muss flexibel in unterschiedlichen Anwendungsszenarien einsetzbar sein. Dies bedeutet, dass es einer Anwendung gewisse Schnittstellen zur Verfügung stellen muss. Durch diese Abstraktion muss bei einer Änderung der Anwendung nicht das Kommunikationsprotokoll erneut getestet werden [17]. Gegebenenfalls wird auch ein dynamisches Allokieren von Kommunikationsressourcen benötigt.

deterministisches Verhalten: Das Verhalten eines echtzeitfähiges Kommunikationssystem muss deterministisch sein. Dies beinhaltet einerseits eine definierte Laufzeit, aber auch, dass Pakete genau ein mal zugestellt werden [16]. Außerdem darf sich die Reihenfolge der Pakete zwischen Sender und Empfänger nicht verändern [15].

Fehlererkennung und -toleranz: Ein echtzeitfähiges Kommunikationssystem muss in der Lage sein, Kommunikations- und Knotenfehler zu erkennen. In einem Fehlerfall müssen alle Komponenten des Echtzeitsystems darüber informiert werden, damit eine entsprechende Fehlerbehandlung stattfinden kann [17]. Das Feststellen eines Fehlerfalles sollte auf allen verteilten Knoten des

Echtzeitsystems gleich sein, da es sonst zu unkoordinierter Fehlerbehandlung kommt. Weiterhin muss ein Stören der Kommunikation durch defekte Knoten verhindert werden [16].

Nachrichtenkategorien: Werden über ein echtzeitfähiges Kommunikationssystem Nachrichten mit unterschiedlichen Anforderungen übermittelt, so muss eine Kategorisierung der Nachrichten geschehen. Die unterschiedlichen Kategorien erfüllen die jeweiligen Anforderungen. So lassen sich die Nachrichten beispielsweise in periodisch auftretende und unregelmäßig auftretende Nachrichten sowie Nachrichten ohne Echtzeitanforderungen unterteilen [15]. Die Verwendung der jeweiligen Kategorie hängt von den internen Triggern der Events ab. Wird ein Wert periodisch ermittelt und übertragen (Time Triggered, TT), so ist die Anzahl und der Zeitpunkt des Auftretens der Nachrichten klar definiert. Soll beim Auftreten eines bestimmten Wertes eine Komponente benachrichtigt werden, so können die Nachrichten zu beliebigen Zeitpunkten auftreten (Event Triggered, ET). Dabei muss die Rate der Events begrenzt werden, da sonst Teile des Echtzeitsystems durch die Anzahl der Events überlastet werden. [17]

Zertifizierbarkeit: Echtzeitfähige Kommunikationssysteme, die in für den Betrieb kritischen Systemen eingesetzt werden sollen, müssen zertifiziert werden. Dabei muss ein korrektes Funktionieren der Kommunikation bei jeder Auslastung und unter definierten Fehlerzuständen sichergestellt werden. Ist ein Protokoll darauf ausgelegt, zertifiziert zu werden, so erleichtert dies den Zertifizierungsprozess. [16]

Bereitstellung einer globalen Zeit: Verteilte Echtzeitsysteme benötigen eine globale Zeit. Sollen periodisch Messwerte, die den Zustand eines Systems zu einem bestimmten Zeitpunkt darstellen, durch mehrere verteilte Knoten erfasst werden, so muss dies zeitlich koordiniert werden. Da die Zeitgeber in den einzelnen Knoten einem Drift unterliegen, müssen die Uhren der Knoten regelmäßig synchronisiert werden. Dies kann zwar durch externe Systeme wie eine gemeinsame Taktquelle realisiert werden, jedoch lässt sich dieses System durch die Verwendung eines Protokolls auf Basis des Kommunikationssystems zur Zeitsynchronisation einsparen. [17]

3 Feldebussysteme im Automobil

Im Automobil kommt eine Vielzahl von Feldebussystemen zum Einsatz. Sie sind jeweils auf Anwendungszwecke spezialisiert. Hier wird ein Überblick über die am meisten verwendeten Feldebussysteme gegeben.

CAN: Das Controller Area Network (CAN) Protokoll, 1983 vom Unternehmen Bosch spezifiziert, stellt pro Bus eine Kollisionsdomäne dar. Alle an den Bus angeschlossenen Knoten empfangen eine

Nachricht und entscheiden, ob sie diese verarbeiten. Jede Nachricht ist mit einer Arbitrierungs-ID markiert, die gleichzeitig die Priorität festlegt und bei der Busarbitrierung zur Konfliktlösung verwendet wird. Die Nachrichtenübertragung über den Bus geschieht Event Triggered, eine Erweiterung spezifiziert Time Triggered CAN. CAN erkennt Übertragungsfehler, Kollisionen und garantiert definierte Übertragungszeiten. Das Busmedium für CAN ist nicht standardisiert, die Übertragungsrate liegt bei bis zu 1 MBit/s. [3, 17, 1]

LIN: Das Local Interconnect Network (LIN) Protokoll stellt pro Bus ebenfalls eine Kollisionsdomäne dar, allerdings ist ein Master definiert, der den Zugriff koordiniert. Ein Buszugriff besteht immer aus dem Paket vom Master zu einem Slave und dem Antwortpaket. Die Pakete enthalten Prüfsummen zum Feststellen von Übertragungsfehlern. LIN ist für die kostengünstige Anbindung von Sensoren und Aktoren konzipiert und nutzt eine vorhandene UART-Schnittstelle. Die Übertragungsrate beträgt 20 kBit/s. [19]

FlexRay: FlexRay wurde von einem Konsortium für den Einsatz im Automobil entwickelt. Der Zugriff auf die gemeinsame Kollisionsdomäne wird durch ein TDMA-Verfahren (Time Division Multiple Access) geregelt, das einen statischen Bereich für TT-Nachrichten und einen dynamischen Bereich für ET-Nachrichten beinhaltet. Nur im statischen Bereich können deterministisch Nachrichten übermittelt werden, da für jede Nachricht ein fester Zeitbereich reserviert ist. Im dynamischen Bereich existiert für jede Nachricht ein minimaler Zeitbereich, der bei Bedarf verlängert wird. Durch die Position des Paketes im dynamischen Bereich wird gleichzeitig die Priorität festgelegt. FlexRay unterstützt eine Linien- und Sterntopologie (passiver Bus/aktiver Stern) sowie einen redundanten Netzaufbau. Eine Zeitsynchronisation wird durch mehrere redundante Zeitsynchronisationsmaster am Anfang jedes Buszyklusses durchgeführt. Die Übertragungsrate beträgt maximal 10 MBit/s. [31, 3]

MOST: Media Oriented System Transport (MOST) ist ein 1998 durch ein Konsortium spezifizierter Bus für Multimediaanwendungen im Automobil. Durch das Konsortium sind sowohl ein optisches als auch ein elektrisches Busmedium spezifiziert. Meist wird eine Ringtopologie verwendet, die bei Bedarf redundant ausgeführt wird. Auch Sterntopologien sind mit einem Hub realisierbar. Im MOST Ring werden von einem Master Frames mit einer definierte Samplingrate eingespeist, die Übertragungsrate ist direkt von der Samplingrate abhängig. Die Frames werden von den Slaves zur Zeitsynchronisation verwendet und werden von ihnen im Ring weitergeleitet. MOST stellt verbindungsorientierte Datenübertragungen mit garantierten Bandbreiten bereit, für jeden Datenstrom ist in jedem Frame ein Datenbereich allokiert. Je nach verwendeten Standard erlaubt MOST bis zu 15 (MOST25), 29 (MOST50) oder 93 (MOST150) Datenströme mit jeweils 4 Byte Daten pro Frame. Daraus ergibt sich eine Übertragungsrate von 25, 50 beziehungsweise 150 MBit/s. [3, 20, 21, 6]

LVDS: Low Voltage Differential Signaling (LVDS) stellt lediglich eine Methode zur Datenübertragung dar. Eine Datenverbindung besteht aus einem oder mehreren Paaren von Datenleitungen und einer Taktleitung. Die Daten werden mit einem anwendungsabhängigen Protokoll takt-synchron übermittelt. Mit LVDS sind abhängig von Leitungslänge, äußeren Störeinflüssen sowie verwendeter Hardware mehrere GBit/s Übertragungsgeschwindigkeiten pro Datenleitung zu erreichen. Die Datenübertragung findet unidirektional statt.

4 Ethernet im Automobil

Ethernet wird nicht nur in zeitlich unkritischen Systemen verwendet. So wird Ethernet zum Datenlogging und -monitoring der Sensordaten unter Echtzeitbedingungen im Wendelstein 7-X Fusionsreaktor verwendet [28]. Auch in Fertigungsanlagen finden auf Ethernet basierende Technologien wie PROFINET IRT und EtherCAT unter Echtzeitbedingungen Verwendung [24]. In der Luftfahrt ist Ethernet mit AFDX im A380 vertreten [17]. Time Triggered Ethernet (TTE) zielt auf eine Verwendung im Automobil ab. Mit dem IEEE 802.1BA Standard für Audio/Video Bridging (AVB) [9] wird Ethernet um ein Protokoll für zeitsensitive Applikationen erweitert. Im Folgenden wird ein Überblick über Ethernet in Echtzeitsystemen gegeben.

Echtzeitsysteme mit normalem Ethernet: Da Ethernet mittlerweile weit verbreitet ist, verfügen SoCs (System on Chip) oftmals über eine integrierte Ethernet Anbindung. Dies führt zu einer Kostenreduktion bei einer Verwendung von Ethernet im Vergleich zu Bussystemen mit spezielle Hardwarekomponenten zur Anbindung. Hinzu kommt, dass für Ethernet unterschiedliche Übertragungsmedien mit Bandbreiten von 10 MBit/s bis 10 GBit/s verfügbar sind. Weiterhin erlaubt Ethernet einen flexiblen strukturierten Netzaufbau. [15, 25]

Ethernet nach dem IEEE 802.3 Standard ist auf Grund des verwendeten CSMA/CD- (Carrie Sense Multiple Access/Collision Detection) und Backoff-Verfahrens nur begrenzt Echtzeitfähig. Bei Kollisionen führen diese Verfahren besonders bei hoher Netzwerkauslastung zu einem großen Jitter und gegebenenfalls Paketverlusten. Von Kweon und Shin [18] wurde ein Verfahren zur statistischen Reduktion der Kollisionen und somit des Jitters entwickelt. Es basiert auf einem Traffic-Smoother im Protokollstack zwischen Link- und Protokoll-Layer, der burstartiges Paketvorkommen verhindert. Der Traffic-Smoother gleicht dem Credit-Based Traffic-Shaper von AVB. Durch dieses Verfahren kann zwar eine statistische Wahrscheinlichkeit für das Einhalten einer vorgegebenen Laufzeit unter definierter Netzwerkauslastung bestimmt werden, jedoch kann dem Jitter keine enge Grenze gesetzt werden. Ein Erkennen von Paketverlusten findet ebenfalls nicht statt. Da keine deterministische Paketlaufzeit erreicht werden kann, eignet sich dieses Verfahren nicht für kritische Echtzeitsysteme.

Dabei wurden ein Verlust von einem Paket je zehntausend akzeptiert. Durch das Verwenden von geschwitchten Netzwerken wird eine gemeinsame Kollisionsdomäne verhindert, da zwischen jedem Knoten und dem Switch eine bidirektionale Verbindung besteht [2, 30]. Insbesondere können dabei unterschiedliche Knoten ungestört voneinander kommunizieren. Da die Daten bei hoher Auslastung der verfügbare Bandbreite im Switch gepuffert werden, kommt es zu einem hohem Jitter [26]. Zusätzlich können Pakete bei Pufferüberläufen im Switch verloren gehen. Auch mit Prioritäten berücksichtigendem Forwarding nach IEEE 802.1Q [10] ist keine deterministische Laufzeit zu erreichen [2].

Geswitchtes echtzeitfähiges Ethernet: Von Carvajal u. a. [2] wurden unterschiedliche Kombinationen von COTS-Hardware (Commercial of the Shelf) und modifizierte Hardware evaluiert. Die Echtzeitkommunikation wird mit Time Triggered Nachrichten abgewickelt, während unkritische Kommunikation parallel über die selben Netzwerkressourcen übertragen wird. Die modifizierte Hardware wurde dabei um zusätzliche Arbitrierungsmechanismen zur Kontrolle des Netzwerkzugriffes erweitert. Eine Verwendung von COTS-Hardware führt auf Grund der weiten Verfügbarkeit zu einer Kostenreduktion. Für die Evaluation wurde in Betracht gezogen, entweder den Switch zu modifizieren, die Netzwerkinterfaces der Knoten zu modifizieren oder beides zu modifizieren.

Werden nur die Netzwerkinterfaces modifiziert, so stellen die Netzwerkinterfaces das Einhalten eines kollisionsfreien Timings sicher. Somit muss der gesamte Netzwerkverkehr koordiniert werden und nur modifizierte Netzwerkinterfaces können verwendet werden. Wird nur der Switch modifiziert, so zieht der Switch dabei Echtzeitnachrichten beim Forwarding den unkritischen Nachrichten vor. Zusätzlich werden die Echtzeitnachrichten in Puffern gehalten und zu definierten Zeitpunkten weitergeleitet. Durch den Switch kann so ein unkontrolliertes Fluten durch einen defekten Knoten verhindert werden. Es können auch Netzwerkknoten eingebunden werden, die über keine echtzeitfähigen Netzwerkinterfaces verfügen. Das modifizieren der Netzwerkinterfaces und des Switches stellt die aufwändigste Option dar. Die Netzwerkinterfaces stellen dabei das Einhalten eines kollisionsfreien Timings der Echtzeitnachrichten sicher, während der Switch die Echtzeitnachrichten vorrangig forwarded.

Mit Messungen wurde in der Evaluation belegt, dass der modifizierte Switch besonders bei hoher Auslastung der Bandbreite einen wesentlich geringeren Jitter als der COTS-Switch aufwies. Beim Erreichen der maximalen Bandbreite wurden zudem Paketverluste beim modifizierten Switch nur bei unkritischen Nachrichten festgestellt. [2]

EtherCAT: EtherCAT ist ein auf Ethernet basierendes Protokoll, das in der industriellen Automatisierung Anwendung findet. Es wurde vom Unternehmen Beckhoff initiiert und wird durch die EtherCAT Technology Group weiterentwickelt. Als Topologie wird meist eine Linie oder ein Ring verwendet, prinzipiell ist aber jede Topologie umzusetzen. Durch einen Master wird ein Frame an

den ersten der miteinander Verbundenen Slaves übertragen. Während des Empfanges des Frames wird durch jeden Slave ein definierter Datenbereich geändert, die benötigten Daten extrahiert und das Frame ohne weitere Pufferung an den Nächsten Slave weitergeleitet. Am Ende erreicht das Frame wieder den Master. Die Frames werden durch den Master periodisch versandt, wodurch eine Zeitsynchronisation der Knoten stattfindet. Für die Cut-Through Weiterleitung benötigen die Slaveknoten spezielle Hardware, die aktuell nur mit Übertragungsraten von 100 MBit/s zur Verfügung stehen. Allerdings beträgt die Latenz beim Forwarding lediglich 500 ns. Normaler Ethernetverkehr, beispielsweise auf Basis von TCP/IP, wird eingebettet weitergeleitet. Dementsprechend können keine normalen Netzwerkknoten verwendet werden. [24, 4]

PROFINET IRT: PROFINET IRT (PROcess Field NETwork Isochronous Real Time) ist ein von der Firma Siemens entwickeltes, auf Ethernet basierendes Protokoll. Es findet ebenfalls in der industriellen Automatisierung Anwendung. Ein Master sendet über eine Ring-, Linien- oder Sterntopologie periodisch eine Nachricht an jeden Slave. PROFINET IRT erlaubt eine Zeitsynchronisation der Knoten. Netzwerkinterfaces und speziell für PROFINET IRT optimierte Cut-Through Switches werden lediglich von Siemens vertrieben, die Übertragungsrate der Komponenten beträgt 100 MBit/s. Das Protokoll verwendet zum Einhalten der Echtzeitanforderungen einen nicht veröffentlichten Scheduling, das TT- und ET-Nachrichten zulässt. [24, 23]

AFDX: Avionics Full Duplex Switched Ethernet (AFDX) ist ein IEEE 802.3 konformes Protokoll, das für Anwendungen in der Luftfahrt vorgesehen ist. Das Protokoll ist durch den ARINC-Standard 664 standardisiert. AFDX garantiert eine Übertragung von Nachrichten mit einer definierten Übertragungsrate zwischen einem Sender und einem oder mehreren Empfängern. Das Einhalten der Ratenlimitierung wird durch spezielle Switches umgesetzt. Zudem werden eine unveränderte Reihenfolge der Nachrichten beim Sender und den Empfängern sowie eine maximale Übertragungszeit und ein definierter Jitter garantiert. Nachrichtenverluste auf Grund von Pufferüberläufen in den Switches können ausgeschlossen werden. [17]

TTE: Time Triggered Ethernet (TTE) ist ein auf Ethernet basierendes und damit kompatibles Protokoll. Es wurde an der Technischen Universität Wien entwickelt und wird durch das Unternehmen TTTech vertrieben. Der Anwendungsbereich von TTE zielt auf kritische Echtzeitsysteme in der Luftfahrt und in Automobilen ab. TTE Netzwerkkomponenten sind aktuell mit Bandbreiten von 10/100/1000 MBit/s verfügbar, das Netzwerk kann redundant aufgebaut werden. Das Protokoll unterstützt die Paketklassen TT, Rate Constrained (RC) und Best Effort (BE). TT-Pakete garantieren ein deterministisches Verhalten durch einen a priori bestimmten Schedule. Für RC-Pakete wird a priori eine maximale Rate reserviert, sie haben eine höhere Priorität als BE-Pakete, die für den normalen Ethernet Verkehr ohne spezielle Echtzeitanforderungen verwendet werden.

Die TTE Switches des Netzes werden mit dem errechneten Scedule statisch konfiguriert. Im Scedule ist für jedes TT-Paket ein TDMA Zeitschlitz vorgesehen. TT-Pakete werden durch einen definierten Wert im Type Feld der MAC-Frames identifiziert. Die TTE Switches sorgen dafür, dass gescedulte TT-Pakete nur zu den definierten Zeitpunkten übermittelt werden. Blockiert dabei ein anderes RC- oder BE-Paket die Leitung, so wird dieses durch den Switch unterbrochen und nach Übermitteln des TT-Pakets erneut übertragen. Eine Zeitsynchronisation der Knoten und Switches wird über den IEEE 1588 Standard [7] durchgeführt. [15, 16, 26, 29]

Von Steinbach u. a. [26] wird ein Applikation auf Basis von TTE analysiert. Dabei handelt es sich um Rundum-Bilderfassung durch vier Kameras an einem Automobil. Die Kameras erfassen mit VGA-Auflösung 30 Bilder pro Sekunde, wodurch pro Kamera ein Datenstrom von 150 MBit/s erzeugt wird. Eine oder mehrere ECUs verarbeiten die Datenströme und stellen die aufbereiteten Daten dem Nutzer bereit. Die maximale Laufzeit beträgt 33,3 ms und der maximale Jitter 3,33 ms. Diese Vorgaben wurden von der Ingenieursgesellschaft Auto und Verkehr (IAV GmbH) bereitgestellt und basieren auf realistischen Annahmen. Der Anwendungsfall wurde mit OMNET++, einer Simulationsumgebung für Ethernet, simuliert. Dabei wurden die Daten jeweils mittels TT-, RC- und BE-Paketen übertragen. Bei der Übertragung mit BE-Paketen kam es dabei zu einem inakzeptabel hohen Jitter von ca. 14700 μ s, wobei die maximale Laufzeit unter den Vorgaben lag. Bei der Übertragung mit TT- beziehungsweise RC-Paketen wurde dabei ein signifikant geringerer Jitter von ca. 30 μ s erreicht. Wurde die Übertragung der TT-Pakete der einzelnen Kameras im voraus kollisionsfrei gesceduled, so kam ein Jitter von ca. 10 μ s zustande. Diese Analyse zeigt, dass TTE auch bei hohen Übertragungsraten eine deterministische Paketübertragung garantieren kann.

Mit TTE kann also entsprechend eine deterministische, echtzeitfähige Kommunikation umgesetzt werden. Auch eine meist für kritische Echtzeitsysteme benötigte Größenordnung der Laufzeit wird garantiert. Dabei muss a priori ein Scedule berechnet werden, das statisch in den TTE-Netzwerkkomponenten konfiguriert wird. Ein Ändern des Scedules zur Laufzeit ist nicht vorgesehen, die Ressourcen sind statisch zugeordnet.

AVB: Audio/Video Bridging (AVB) ist eine durch die IEEE standardisierte Erweiterung des Ethernet Standards für zeitsensitive Anwendungen. Der Standard wurde ursprünglich für die latenzarme Übertragung von Audio- und Videostreamen konzipiert, eignet sich aber generell für zeitsensitive Anwendungen. AVB besteht aus vier Erweiterungen zum IEEE 802.1Q Standard, die Protokolle zur Zeitsynchronisation, Streamreservierung und Streamübertragung sowie Forwarding Regeln für Datenströme definieren [13]. Durch AVB werden zwei QoS-Klassen (Quality of Service) mit unterschiedlichen garantierten Laufzeiten bereitgestellt. Neben zeitsensitiven Datenströmen erlaubt AVB das Übertragen von normalem Ethernetverkehr.

Der IEEE 802.1AS Standard [8] spezifiziert eine kontinuierliche Zeitsynchronisation mit dem Precision Time Protocol (PTP) IEEE 1588 Standard [7]. Durch den Best Master Clock Algorithm

(BCMA) wird der beste Zeitsynchronisationsmaster bestimmt. Er wird als zeitliche Referenz zur Synchronisation verwendet [31]. Durch das im IEEE 802.1Qat Standard definierte Stream Reservation Protocol (SRP) [11] können Datenströme zur Laufzeit durch Endknoten bekanntgemacht und angefordert werden. Die einzelnen Switches berechnen bei der Anforderung eines Datenstroms, ob noch genügend Bandbreite zur Übertragung garantiert werden kann, reservieren diese Bandbreite gegebenenfalls und melden das Ergebnis der Anforderung an die betroffenen Endknoten. Ein Switch darf eine Anfrage nur positiv beantworten, wenn die Übermittlung des Datenstroms mit der angeforderten Bandbreite im Rahmen der durch die QoS-Klasse vorgegebene Paketlaufzeit garantiert werden kann. Der IEEE 802.1Qat Standard [12] definiert einen Credit-Based Traffic-Shaper zum Garantieren der reservierten Bandbreite. Durch den IEEE 802.1BA Standard [9] wird AVB allgemein beschrieben und der Zusammenhang der einzelnen Standards hergestellt.

Von Steinbach u. a. [27] wurden TTE und AVB im Bezug auf Jitter und Laufzeit durch eine OMNET++ Simulation verglichen. Die Simulation zeigte, dass AVB insbesondere bei breitbandigen Multimediadatenströmen ähnliche und auch bessere Laufzeiten und Jitter wie TTE-RC Nachrichten aufweisen. AVB lässt außerdem eine dynamische Allokation von Ressourcen zur Laufzeit zu.

5 Ziele

Die vorherigen Kapitel haben gezeigt, dass ein breites Spektrum von echtzeitfähige Kommunikationssystemen im Automotive Bereich vorhanden ist. Allerdings sind die meisten Kommunikationssysteme nicht für multimediale Anwendungen mit hohen Anforderungen an die Übertragungsbandbreite geeignet. Lediglich MOST und die auf Ethernet basierenden Kommunikationssysteme stellen eine entsprechende Bandbreite bereit. Allerdings ist MOST nicht für das Übertragen von systemkritischen Echtzeitdaten konzipiert und basiert auf nur für MOST verwendeten Hardware-schnittstellen. Mit AFDX lassen sich die für bestimmte Systeme benötigte minimalen Latenzen nicht erreichen. EtherCAT und PROFINET IRT haben zwar eine entsprechend hohe Bandbreite, jedoch wird diese bei EtherCAT von allen Knoten genutzt und ein Verwenden von Knoten ohne entsprechende Hardware ist nicht erlaubt. Zudem erlaubt PROFINET IRT lediglich eine Kommunikation zwischen einem Master und mehreren Slaves und ist somit nicht flexibel genug.

TTE umgeht diese Nachteile, lässt allerdings keine dynamische Allokation von Netzwerkressourcen zur Laufzeit zu. Dabei kann TTE sowohl für systemkritische Echtzeitkommunikation, als auch für unkritische Kommunikation und normalen Ethernetverkehr verwendet werden. Da es aber, wie in der Einleitung erläutert, abzusehen ist, dass bestimmte Anwendungen diese dynamische Allokation benötigen, werde ich mich in den kommenden Semestern mit einer dynamischen Allokation von Netzwerkressourcen im TTE durch AVB auseinandersetzen. Im Rahmen der kommenden Semester ist auch die Entwicklung eines Switch-Prototypens, der die Prinzipien der beiden Protokolle vereinigt, vorgesehen. Der Prototyp wird auf Basis der NetFPGA Plattform der Stanford Univer-

sity entstehen und die damit erreichten Eigenschaften durch Messungen verifiziert werden. Die Entwicklung des Prototyps sollte auf Grund des umfangreichen Aufwandes auf bestehenden Projekten aufbauen. Entsprechende Projekte auf Basis des NetFPGA sind als Open Source verfügbar, deren Verwendbarkeit wurde jedoch noch nicht ausreichend betrachtet.

6 Risiken

Während der Umsetzung dieses Konzeptes besteht das Risiko, dass unterschiedliche Schwierigkeiten auftreten. Diese werden im Folgenden beschrieben. Die Entwicklung wird auf Basis einer Plattform stattfinden, mit der im Umfeld der CoRE Gruppe noch keine Erfahrungen gesammelt wurden. Somit ist es insbesondere schwierig, den genauen Aufwand abzuschätzen, in Folge dessen kann er unterschätzt werden. Die NetFPGA Plattform wird und wurde jedoch für viele Wissenschaftliche Projekte verwendet. Weiterhin besteht eine Community, die über Erfahrungen mit dieser Plattform verfügt. Eine Simulation des Konzepts mit OMNET++ wird aktuell von einem Kommilitonen durchgeführt. Dementsprechend ist noch nicht abzusehen, ob es wesentliche Probleme bei der Umsetzung des Konzeptes gibt. Im kommenden Semester muss deshalb das Konzept mit den zugrunde liegenden Protokollen und Funktionsweisen analysiert werden. Weiterhin ist verfügt die gewählte Hardwareplattform eventuell nicht über ausreichend Hardwareressourcen. Dies ist zwar im Betracht der auf dieser Plattform entwickelten Projekte nicht zu erwarten, kann aber auch nicht ausgeschlossen werden. Tritt dies ein, so muss eine Plattform mit entsprechend größeren Ressourcen gewählt werden. Der Umstieg auf eine andere Hardwareplattform muss durch eine Hardwareabstraktion mit entsprechende Schnittstellen erleichtert werden. Weiterhin muss abgeschätzt werden, welche Auswirkungen eine Rekonfigurierbarkeit von Netzwerkressourcen im laufenden Betrieb auf die Funktionssicherheit von systemkritischer Kommunikation auf dem selben Medium hat. So darf das installieren einer fehlerhaften Applikation auf keinen Fall zu einer Beeinflussung der kritischen Systeme eines Automobils führen. Da es sich bei dem Switch jedoch lediglich um einen Prototypen für das vorgestellte Konzept handelt, wird dieser Aspekt nicht weiter behandelt.

7 Zusammenfassung

In dieser Ausarbeitung wurden die Voraussetzungen für Kommunikation im Rahmen von Echtzeitsystemen beschrieben. Es wurde ein Übersicht über die Eigenschaften unterschiedlicher echtzeitfähiger Kommunikationssysteme gegeben. Dabei wurde insbesondere gezeigt, welche Vorteile TTE hat. Mit einem Ausblick auf zukünftige Entwicklungen im Automobil wurde der Bedarf für eine dynamische Allokation von Netzwerkressourcen für zeitkritische Anwendungen gezeigt. Ein entsprechendes Konzept mit einer statischen Allokation durch TTE für systemkritische Anwendungen sowie dynamische Allokation von Netzwerkressourcen durch AVB wurde vorgestellt.

Literatur

- [1] *CAN Specification Version 2.0*. Bosch, 1991. URL: http://www.bosch-semiconductors.de/media/pdf_1/canliteratur/can2spec.pdf.
- [2] Gonzalo Carvajal, Chun W. Wallace Wu und Sebastian Fischmeister. „Evaluation of Communication Architectures for Switched Real-time Ethernet“. In: *IEEE Transactions on Computers* (2012). DOI: [10.1109/TC.2012.177](https://doi.org/10.1109/TC.2012.177). URL: <http://www.computer.org/portal/web/csdl/doi/10.1109/TC.2012.177>.
- [3] Thomas Dohmke. *Bussysteme im Automobil: CAN, FlexRay und MOST*. 2002. URL: http://swt.cs.tu-berlin.de/lehre/eks/ws0102/Bussysteme_Vortrag.pdf.
- [4] *EtherCAT*. URL: www.ethercat.de.
- [5] IEEE 802.3 Ethernet Working Group. *Reduced Twisted Pair Gigabit Ethernet PHY - Call for Interest*. März 2012. URL: http://www.ieee802.org/3/RTPGE/public/mar12/CFI_01_0312.pdf.
- [6] Andreas Grzempa. *MOST - The Automotive Multimedia Network*. Franzis, 2011.
- [7] *IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*. IEEE, 2008.
- [8] *IEEE 802.1AS Standard for Local and Metropolitan Area Networks - Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks*. IEEE, 2011.
- [9] *IEEE 802.1BA Standard for Local and metropolitan area networks—Audio Video Bridging (AVB) Systems*. IEEE, Sep. 2011. DOI: [10.1109/IEEESTD.2011.6032690](https://doi.org/10.1109/IEEESTD.2011.6032690).
- [10] *IEEE 802.1Q Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks*. IEEE, 1999. DOI: [10.1109/IEEESTD.1999.89204](https://doi.org/10.1109/IEEESTD.1999.89204).
- [11] *IEEE 802.1Qat Amendment 14: Stream Reservation Protocol (SRP) - 802.1Qat*. IEEE, Mai 2010.
- [12] *IEEE 802.1Qav Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams - 8.2.1Qav*. IEEE, Dez. 2010.
- [13] *IEEE Audio/Video Bridging Task Group*. Online, abgerufen am 15.02.2013. URL: <http://www.ieee802.org/1/pages/avbridges.html>.
- [14] Willie D. Jones. „Smarter Cars? There’s an App for That“. In: *IEEE Spectrum* (Feb. 2011). URL: <http://spectrum.ieee.org/green-tech/advanced-cars/smarter-cars-theres-an-app-for-that>.
- [15] H. Kopetz. „The Rationale for Time-Triggered Ethernet“. In: *Real-Time Systems Symposium*, 2008. Nov. 2008, S. 3 –11. DOI: [10.1109/RTSS.2008.33](https://doi.org/10.1109/RTSS.2008.33).

- [16] H. Kopetz u. a. „The time-triggered Ethernet (TTE) design“. In: *Object-Oriented Real-Time Distributed Computing, 2005. ISORC 2005. Eighth IEEE International Symposium on*. Mai 2005, S. 22 –33. DOI: [10.1109/ISORC.2005.56](https://doi.org/10.1109/ISORC.2005.56).
- [17] Hermann Kopetz. *Real-Time Systems: Design Principles for Distributed Embedded Applications*. 2. Aufl. Springer, 2011.
- [18] Seok-Kyu Kweon und K.G. Shin. „Statistical real-time communication over Ethernet“. In: *Parallel and Distributed Systems, IEEE Transactions on* 14.3 (März 2003), S. 322 –335. ISSN: 1045-9219. DOI: [10.1109/TPDS.2003.1189588](https://doi.org/10.1109/TPDS.2003.1189588).
- [19] LIN Konsortium. *Local Interconnect Network*. Online, abgerufen am 13.02.2013. URL: <http://www.lin-subbus.org/>.
- [20] MOST Cooperation. *Media Oriented System Transport*. Online, abgerufen am 13.02.2013. URL: <http://www.mostcooperation.com>.
- [21] *MOST Specification 3.0 V2*. MOST Cooperation, Juli 2010.
- [22] N. Navet u. a. „Trends in Automotive Communication Systems“. In: *Proceedings of the IEEE* 93.6 (Juni 2005), S. 1204 –1223. ISSN: 0018-9219. DOI: [10.1109/JPROC.2005.849725](https://doi.org/10.1109/JPROC.2005.849725).
- [23] PROFINET. URL: <http://www.profibus.com/technology/profinet/>.
- [24] G. Prytz. „A performance analysis of EtherCAT and PROFINET IRT“. In: *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on*. Sep. 2008, S. 408 –415. DOI: [10.1109/ETFA.2008.4638425](https://doi.org/10.1109/ETFA.2008.4638425).
- [25] SAE. *Deterministic Ethernet: SAE AS6802 "Time-Triggered Ethernet"*. Online, abgerufen am 28.01.2013. 2011. URL: http://de.slideshare.net/slideshow/embed_code/10331509.
- [26] Till Steinbach, Franz Korf und Thomas C. Schmidt. „Real-time Ethernet for Automotive Applications: A Solution for Future In-Car Networks“. In: *2011 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*. Berlin: IEEE Press, 2011, S. 216–220. ISBN: 978-1-4577-0233-4. DOI: [10.1109/ICCE-Berlin.2011.6031843](https://doi.org/10.1109/ICCE-Berlin.2011.6031843).
- [27] Till Steinbach u. a. „Tomorrow’s In-Car Interconnect? A Competitive Evaluation of IEEE 802.1 AVB and Time-Triggered Ethernet (AS6802)“. In: (Sep. 2012).
- [28] Steffen Tanbach und Christine Henning. „Wie zuverlässig ist der Empfang von Netzwerkpaketen bei der Verwendung des Transportprotokolls UDP?“ In: *Informatik Spektrum* 30 (Apr. 2007), S. 84–90. DOI: [10.1007/s00287-007-0138-8](https://doi.org/10.1007/s00287-007-0138-8).
- [29] *TTEthernet Specification*. TTTech, 2008.

- [30] J. Wang und Binoy Ravindran. „Time-utility function-driven switched Ethernet: packet scheduling algorithm, implementation, and feasibility analysis“. In: *Parallel and Distributed Systems, IEEE Transactions on* 15.2 (Feb. 2004), S. 119 –133. ISSN: 1045-9219. DOI: [10 . 1109 / TPDS . 2004 . 1264796](https://doi.org/10.1109/TPDS.2004.1264796).
- [31] Helge Zinner u. a. „A comparison of time synchronization in AVB and FlexRay in-vehicle networks“. In: *Intelligent Solutions in Embedded Systems (WISES), 2011 Proceedings of the Ninth Workshop on*. Juli 2011, S. 67 –72.