



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

AW1 Ausarbeitung

Christian Strahl

Realtime Ethernet aus Sicht der Flugzeug- und
Automobilindustrie

Inhaltsverzeichnis

1 Einführung	3
2 Aktuelle Situation in der Automobilindustrie	4
2.1 Vorteile durch Ethernet	4
3 Bussystemkonzepte	6
3.1 Timetriggered	6
3.2 Tokenbus	6
3.3 Trafficshaping	6
4 AFDX	8
4.1 Virtual Link	10
4.2 AFDX-Message	11
4.3 Netzwerklast	11
4.3.1 Virtual Link	12
4.3.2 AFDX-Message	12
4.4 AFDX-Protokoll	13
4.4.1 Frame Aufbau	13
5 Anforderungen der Automobilindustrie	14
5.1 Nachteile von AFDX	14
5.2 Ausblick	14
Literaturverzeichnis	15

1 Einführung

In der frühen Luftfahrt steuerte ein Pilot sein Flugzeug rein mechanisch. Über Stangen und Bowtenzüge wurden seine Bewegungen an die Leitwerke weitergegeben. Diese mechanischen Konstruktionen waren sehr wartungsaufwändig und schwer, und wurden später durch elektronische Systeme (z. B. Fly-by-Wire, Fly-by-Light) ersetzt. In einem modernen Flugzeug gibt es viele elektronische Komponenten, die untereinander kommunizieren müssen. Zu diesem Zweck wurden die Komponenten in der Vergangenheit mit aufwändigen Kabelbäumen verbunden. Da die Menge und Komplexität der zu verbindenden Komponenten in einem modernen Flugzeug immer größer wird, ist die Flugzeugindustrie dazu übergegangen, Bussysteme für ihre Vernetzung zu verwenden.

Die Fahrzeugindustrie steht zur Zeit vor dem selben Problem. Allein der Kabelbaum des neuen Audi Q7 besteht aus ungefähr 4 km Kupferkabel und wiegt etwa 30 kg, dazu kommt dass das Erstellen, Verlegen und Testen eines solchen Kabelbaums ein arbeitsintensiver und damit teurerer Vorgang ist. Der Einsatz von Bussystemen in Fahrzeugen ist auch wegen dem steigendem Bedarf an Bandbreite nötig. Außerdem macht ein Bussystem den Austausch bzw. Einbau neuer Komponenten leichter. So ließe sich z. B. eine Kamera in die Heckklappe des Fahrzeuges einbauen, die beim Rückwärtsfahren dem Fahrer den Bereich hinter dem Fahrzeug auf dem Display des Navigationssystems anzeigt. Hierfür müssen keine neue Kabel verlegt werden.

2 Aktuelle Situation in der Automobilindustrie

Die Automobilindustrie hat bereits vor einigen Jahren erkannt, dass der Einsatz von Bussystemen in Fahrzeugen verschiedene Vorzüge bietet. Jedoch kamen mit der Zeit immer mehr Bussysteme für unterschiedliche Aufgaben hinzu. So werden zur Zeit in einigen Fahrzeugen bis zu fünf verschiedene Bussysteme parallel eingesetzt. Darunter finden sich z. B.:

- **LIN** Local Interconnect Network
- **CAN** Controller Area Network
- **MOST** Media Oriented Systems Transport
- **FlexRay**

Die Verwendung von so vielen unterschiedlichen Bussystemen in einem Fahrzeug bringt mehrere Probleme mit sich:

- Die einzelnen Bussysteme sind zu einander inkompatibel, da sie unterschiedliche Übertragungsverfahren, Signalpegel, Stecker und Leitungen verwenden.
- Jedes Bussystem besitzt seine eigene Sprache und benötigt spezielle Software für die Planung der Datenübertragung.
- Für jedes Bussystem werden speziell geschulte Mitarbeiter benötigt.

Durch den Einsatz von Ethernet als Bussystem würden einige dieser Nachteile entfallen.

2.1 Vorteile durch Ethernet

Ethernet ist standardisiert und wird in vielen Bereichen bereits seit Jahren verwendet. Dadurch ergeben sich eine Reihe von Vorteilen für die Verwendung als Bussystem.

- Verwendung von vorhandener Software und Analysetools, die zum Teil sogar kostenlos verfügbar sind. Wie z. B. der OpenSource Netzwerksniffer Wireshark.

- Die vorhandenen Tools werden bereits seit Jahren verwendet und weiter entwickelt.
- Da viele Personen sie verwenden, können Fehler leichter entdeckt und schneller behoben werden.
- Mitarbeiter die sich mit Ethernet auskennen sind leichter zu finden als Mitarbeiter für proprietäre Bussysteme.
- Die Bandbreite von Ethernet ist höher als die von bisher eingesetzten Bussystemen.

Durch die Verwendung von Ethernet als Bussystem in Fahrzeugen würde zudem die Anbindung des Fahrzeugs an das Internet erleichtert werden. Dadurch hätte man neue Möglichkeiten wie z.B. das Nachladen von Kartenmaterial für das Navigationssystem aus dem Internet.

3 Bussystemkonzepte

Es gibt unterschiedliche Konzepte um Bussysteme zu realisieren. Einige verbreitete Bussystemkonzepte sollen im folgenden kurz vorgestellt werden.

3.1 Timetriggered

Bei diesem Prinzip wird die Kommunikation zu vorher genau festgelegten Zeitpunkten durchgeführt. Diese Methode benötigt eine sehr sorgfältige Planung, da es sonst zu Überschneidungen in der Kommunikation führen kann.

Bekannte Bussysteme die auf dem Timetriggered Prinzip beruhen sind z.B. FlexRay oder TTCAN¹.

3.2 Tokenbus

Bei dieser Methode wird ein sog. Token über das Bussystem versendet. Anders als beim Tokenring wird der Token hierbei an alle Busteilnehmer versendet. Der Teilnehmer für dessen ID der Token bestimmt ist, kann nun seine Daten versenden und anschließend den Token mit der ID seines nächsten Nachbarn auf den Bus senden.

Bekannte Bussysteme die das Tokenbus Prinzip einsetzen sind z.B. ARCNET oder das Feldbussystem Profibus.

3.3 Trafficshaping

Beim Trafficshaping werden verschiedene Verfahren (z.B. LeakyBucket, TokenBucket, ...) eingesetzt um die Netzwerklast gleichmäßig zu verteilen. Oft wird zusätzlich die zur Verfügung stehende Netzwerkkapazität deutlich überdimensioniert (overprovisioning). Durch die

¹Timetriggered Controller Area Network

Analyse des Netzwerkverkehrs und die Verteilung der Netzwerklast werden Staus in den Switchen des Netzwerks vermieden. Dadurch lassen sich grobe Aussagen über die Latenz von Paketen machen.

Bussysteme die Trafficshaping einsetzen sind z. B. AFDX.

4 AFDX

Die Abkürzung AFDX steht für Avionics Full Duplex Switched Ethernet. Dabei handelt es sich um ein zweifach redundantes, echtzeitfähiges geschaltetes Ethernet, in dem ein primäres Netzwerk (Netzwerk A: rot) und ein sekundäres Netzwerk (Netzwerk B: blau) existieren, die stets die gleichen Daten führen. Die verwendeten AFDX-Switche enthalten statische Tabellen, die von den Systementwicklern in NCD-Dateien (Network Control Document) definiert werden. Durch die statische Konfiguration und die Begrenzung des Übertragungsvolumens sind die Laufzeiten von Nachrichten im AFDX-Netzwerk deterministisch. Des Weiteren kontrolliert ein AFDX-Switch auch AFDX-spezifische Parameter (z. B. BAG, maxFrameSize, etc.) und verwirft Pakete, die gegen diese verstoßen. Der Einsatz von AFDX bringt viele Vorteile mit sich. So sinkt der Verdrahtungsaufwand erheblich und erleichtert die Erweiterung des Systems. Auch das Gewicht ist bei einem Bussystem wie AFDX deutlich geringer als bei großen Kabelbäumen. AFDX wird heute bereits im A380 von Airbus eingesetzt und wird auch in vielen weiteren neuen Flugzeugen wie der Boeing-787, dem Airbus A400M und dem Airbus A350 Verwendung finden.

Für die Endsysteme sind die Übertragungsparameter in ICD-Dateien (Interface Control Document) definiert. Sie enthalten neben den Informationen über verschiedene AFDX-Message- und Virtual Link Parameter auch noch Steckerbelegungen.

Abbildung 4.1 zeigt eine schematische Darstellung des AFDX-Netzwerkes in einem Flugzeug. Die roten Rechtecke symbolisieren die AFDX-Switche. Die abgehenden Linien stellen die Verbindungen zu den Endsystemen dar. Sowohl die Switche als auch die Verbindungen sind ein zweites Mal blau im Hintergrund dargestellt (Redundanz). Die weißen Rechtecke stehen für die Controller¹ der einzelnen Systeme. An diese Controller werden beispielsweise Sensoren oder diskrete Signale angeschlossen. Die Grafik zeigt im oberen Bereich für den Betrieb des Flugzeugs kritischere Systeme und nach unten hin weniger kritische Systeme.

¹ Rechnereinheiten im Flugzeug

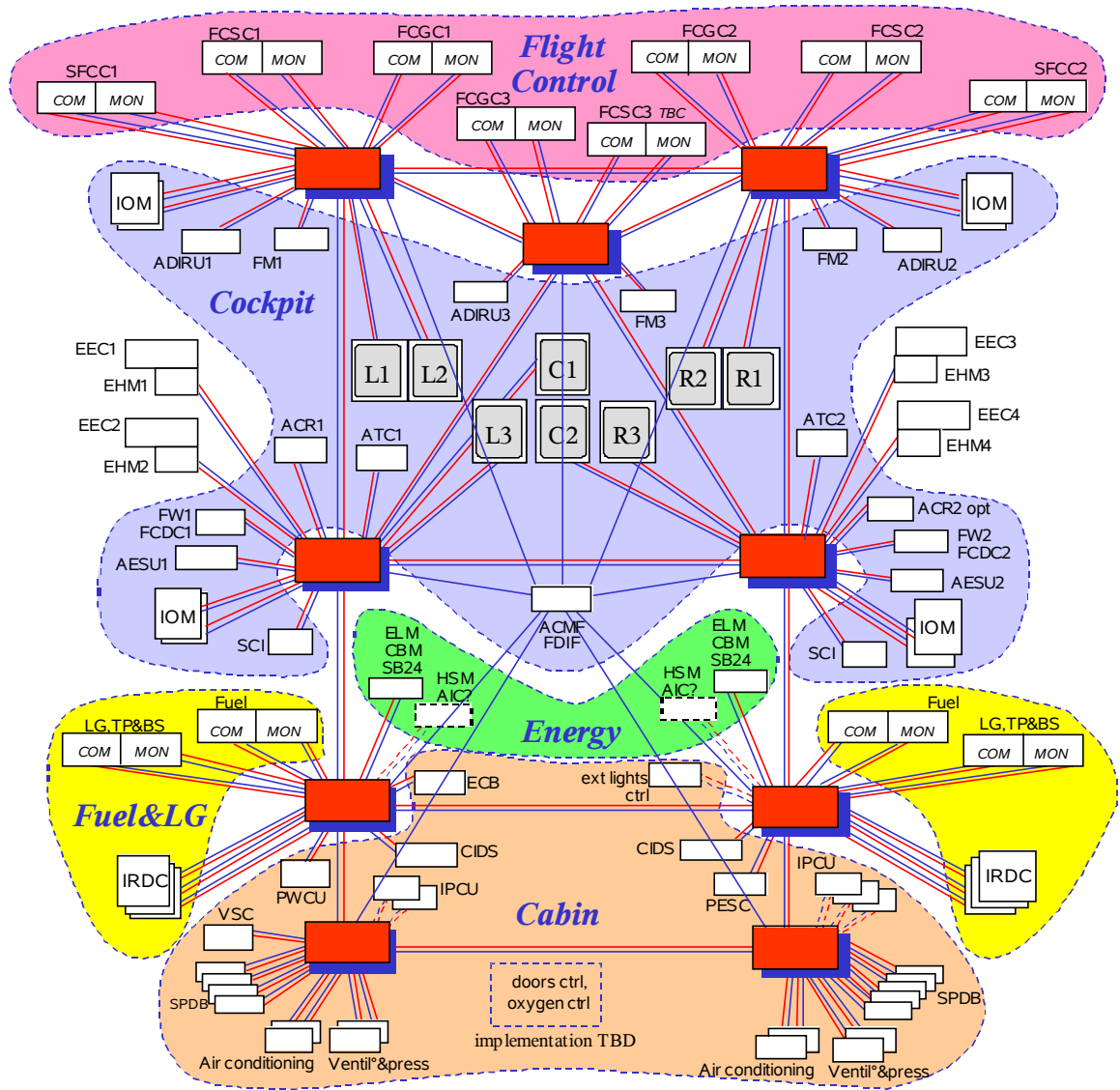


Abbildung 4.1: AFDX-Netzwerk eines Flugzeugs (Miller und Collins (2004))

4.1 Virtual Link

Bevor es AFDX gab, wurden die meisten Controller, gemäß dem ARINC-429 Standard, mittels umfangreicher Kabelbäume direkt miteinander verdrahtet. Bei AFDX werden die früheren Kabelbäume mit Hilfe von Virtual Links auf ein Netzwerk abgebildet. Ein Virtual Link (VL) beschreibt eine unidirektionale 1 zu 1 oder 1 zu n Beziehung zwischen Endsystemen, von denen jeweils immer eines Informations-Sender und der bzw. die anderen Empfänger sind. Über diese virtuellen Verbindungen werden definierte Nachrichtenpakete (AFDX-Messages) zwischen den Controllern ausgetauscht. Abbildung 4.2 zeigt verschiedene Möglichkeiten der Verbindung von Endsystemen.

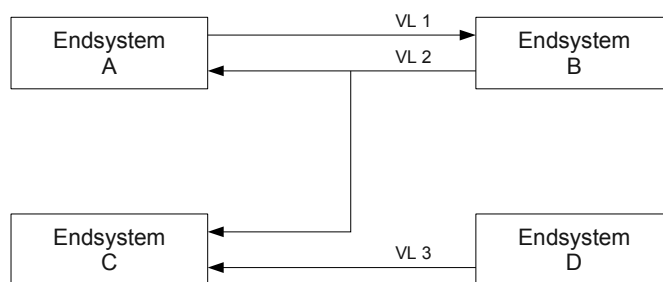


Abbildung 4.2: Mit Virtual Links verbundene Endsysteme

Um die Netzwerklast gleichmäßig zu verteilen, ist für jeden VL eine BAG (Bandwidth Allocation Gap) definiert. Die BAG definiert einen Mindestabstand zwischen zwei AFDX-Frames (siehe Abschnitt 4.4.1) in einem Virtual Link und wird auch als IPT (Inter Packet Time) bezeichnet. Des Weiteren wird für jeden Virtual Link eine maximale Framegröße (maxFrameSize) definiert. Ist die AFDX-Message größer als die maximale Framegröße, so wird das Paket fragmentiert.

Ein VL wird über einen Identifier, genannt Virtual Link Identifier (VLID), identifiziert. Genaues ist dem Abschnitt 4.4.1 über den Aufbau von AFDX-Frames zu entnehmen.

Die für den Nachrichtenversand wichtigsten Virtual Link Parameter sind:

- VLID
- BAG
- maxFrameSize

4.2 AFDX-Message

Für jeden Virtual Link sind AFDX-Messages definiert. AFDX-Messages enthalten die Daten, die zwischen den Controllern ausgetauscht werden sollen. Sie werden mit Hilfe eines oder mehrerer AFDX-Frames (siehe Abschnitt 4.4.1) über das Netzwerk übertragen.

Eine AFDX-Message kann als Unicast an nur eine bestimmte Komponente oder als Multicast an mehrere Controller versendet werden, solange die Controller als empfangende Endsysteme des Virtual Links definiert sind.

Identifiziert wird eine AFDX-Message über die UDP-Zieladresse (im folgenden MessageID genannt). Eine genauere Aufteilung der verschiedenen Header folgt im Abschnitt 4.4.1. Für eine AFDX-Message sind, ähnlich wie bei den Virtual Links, eine Reihe von Übertragungsparametern definiert.

Die für die Kommunikation der Endsysteme wichtigsten Parameter sind:

- **UDP-Zieladresse:** Wird bei AFDX auch MessageID genannt. Ermöglicht es, die AFDX-Message in Verbindung mit der Virtual Link ID eindeutig zu identifizieren.
- **Transmissionsrate:** Definiert eine periodische Zeit nach der die AFDX-Message verschickt wird. Der Wert der Transmissionsrate muss größer oder gleich der BAG des zugehörigen Virtual Links sein.
- **BufferSize:** Gibt die maximale Größe der AFDX-Message an. Ist die BufferSize größer als die maxFrameSize des dazugehörigen VLs wird die AFDX-Message fragmentiert.

4.3 Netzwerklast

Das AFDX-Netzwerk besteht aus einem geschichteten Ethernet Netzwerk mit 100 MBit/s, bzw. 10 MBit/s. Damit diese physikalische Bandbreite nicht überschritten wird, sind allen VLs feste logische Bandbreiten zugeteilt. Der in Abbildung 4.3 dargestellte physikalische Link stellt die Gesamtbandbreite der Ethernet-Verbindung dar. Im Inneren sind die virtuellen Links mit einer ihrer maximalen Bandbreite entsprechenden Größe dargestellt. Die Summe der maximalen Bandbreiten der Virtual Links darf die physikalisch zur Verfügung stehende Bandbreite nicht überschreiten.

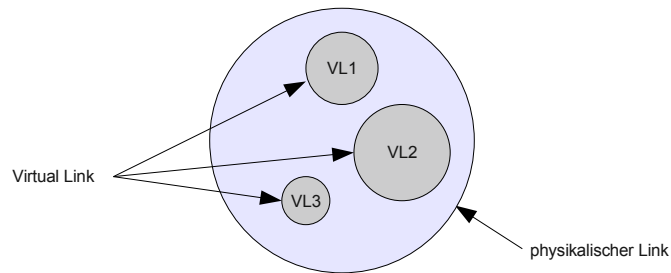


Abbildung 4.3: Aufteilung der Bandbreite auf mehrere Virtual Links

4.3.1 Virtual Link

Mit Hilfe der BAG und der maximalen Framegröße (*maxFrameSize*) lässt sich die für einen Virtual Link maximal benötigte Bandbreite wie folgt berechnen:

$$\text{maximaleVLBandbreite} = \frac{\text{maxFrameSize}}{\text{BAG}}$$

Mögliche Werte für die BAG sind 1, 2, 4, 8, 16, 32, 64 und 128. Für eine *maxFrameSize* von 320 Byte und eine BAG von 64ms würde sich so z.B. folgende maximale Bandbreite ergeben:

$$\frac{320\text{Byte}}{64\text{ms}} = 5000 \frac{\text{Byte}}{\text{s}} = 40 \frac{\text{kBit}}{\text{s}}$$

4.3.2 AFDX-Message

Aufgrund der maximalen Paketgröße (*BufferSize*) und Transmissionsrate lässt sich die für eine AFDX-Message benötigte Bandbreite wie folgt berechnen:

$$\text{MsgBandbreite} = \frac{\text{BufferSize}}{\text{Transmissionsrate}}$$

Eine AFDX-Message mit 300 Byte und einer Transmissionsrate von 100ms benötigt z.B. eine Bandbreite von:

$$\frac{300\text{Byte}}{100\text{ms}} = 3000 \frac{\text{Byte}}{\text{s}} = 24 \frac{\text{kBit}}{\text{s}}$$

Mit dieser Bandbreite ließen sich bei einem 100 MBit/s Netzwerk mehr als 4000 AFDX-Messages definieren. Die Summe der AFDX-Message-Bandbreiten eines Virtual Links darf nicht größer sein als die maximale Virtual Link Bandbreite.

4.4 AFDX-Protokoll

Bei AFDX erfolgt die Datenübertragung mit dem UDP/IP Protokoll. Die einzelnen Frames unterscheiden sich bei AFDX von herkömmlichen Netzwerkframes in einigen Merkmalen.

4.4.1 Frame Aufbau

AFDX-Frames sind die Pakete, die im AFDX-Netzwerk versendet werden. Sie enthalten AFDX-Messages bzw. ihre Fragmente. Ein AFDX-Frame besteht aus einem MAC-Header, einem IP-Header, einem UDP-Header, der UDP-Payload und einer Sequenznummer (SN). Der Aufbau eines AFDX-Frames ist in Abbildung 4.4 skizziert.

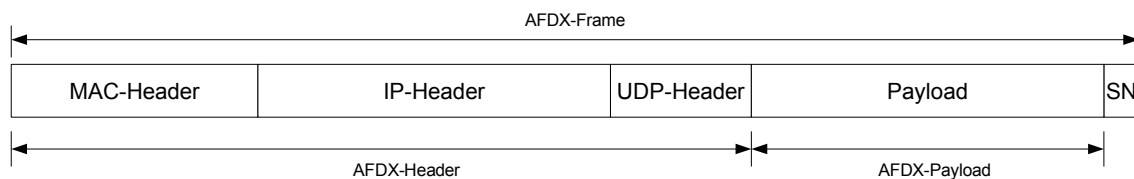


Abbildung 4.4: Aufbau eines AFDX-Frames

Da ein Paket im AFDX-Netzwerk statisch geroutet wird und alle Identifier-Zuordnungen von Flugzeug zu Flugzeug gleich sind, werden einige Einträge in den Protokoll-Headern etwas anders verwendet als im normalen UDP- / Ethernet-Netzwerken. So ist die Ziel-Adresse im MAC-Header in den ersten vier Bytes statisch gesetzt, die hinteren zwei Bytes enthalten die VLID. Die MessageID der AFDX-Messages wird durch den UDP-Ziel Port angegeben. Einige weitere statische Merkmale der AFDX-Header sind nachfolgend als Beispiele aufgeführt:

- Die UDP Checksum ist immer 0.
- Die TTL (Time To Live) im IP-Header ist immer auf 1 gesetzt.
- Diverse Adressteile im MAC- und IP-Header sind fest definiert.

Diese Merkmale erleichtern die Implementierung einer Paketfilterung für AFDX-Frames.

5 Anforderungen der Automobilindustrie

Für viele zeitkritische Anwendungen in der Automobilindustrie sind die Latenz und der Jitter der Nachrichten die wichtigsten Parameter. So muss z. B. der Airbag 10 ms nach einem Aufprall zünden, da bereits nach 20 ms die Relativbewegung der Insassen erfolgt. Ähnliche Zeitanforderungen ergeben sich bei der Steuerung der Gurtstraffer. Bei Eingriffen in die Motorsteuerung (z. B. für ASR¹) müssen sogar noch kürzere Latenzen eingehalten werden.

5.1 Nachteile von AFDX

Einer der größten Nachteile von AFDX ist jedoch, dass die Latenz eines Paketes nicht genau vorherzusagen ist, da diese von der Planung des Paketweges im AFDX-Netzwerk abhängig ist. Ebenso ist der Jitter, also die Abweichung von der erwarteten Laufzeit der Pakete nicht genau vorhersagbar, was in anderen Echtzeit-Bussystemen auf der Basis von Ethernet möglich ist (siehe hierzu [Steinbach \(2008\)](#)).

Da bei AFDX wie in Abschnitt 4.4.1 beschrieben einige Felder des MAC- und des IP-Headers fest definiert sind, also anders als im Ethernet- bzw. dem IP-Standard beschrieben, kann man bei AFDX nicht jedes beliebige Netzwerkpaket switchen. Es ist also nicht ohne weiteres möglich ein Ethernetfähiges Gerät an einem AFDX-Netzwerk zu betreiben. Außerdem benötigt man spezielle AFDX-Netzwerkswitche, die das Virtual Link Konzept verwenden um Pakete im Netzwerk zu verteilen.

5.2 Ausblick

Da AFDX die in Abschnitt 5 beschriebenen Anforderungen nicht erfüllt, ist AFDX für Automobile Anwendungen eher ungeeignet. In meinem Projekt werde ich mich aus diesem Grund mit anderen Echtzeitbussystemen auf Ethernetbasis beschäftigen. Welche das genau sein werden steht zu diesem Zeitpunkt noch nicht fest. Jedoch werden die Untersuchungen von Till Steinbach ([Steinbach \(2008\)](#)) in die Wahl des Bussystemes mit einfließen.

¹Anti Schlupf Regelung

Literaturverzeichnis

- [Miller und Collins 2004] MILLER, David ; COLLINS, Rockwell: *AFDX Determinism*. 2004.
– URL http://www.arinc.com/aeec/general_session/gs_reports/2004/presentations/afdx_determinism_david_miller.pdf
- [P. Holleczeck 2007] P. HOLLECZEK, B. Vogel-Heuser: *Mobilität und Echtzeit*. Springer, 2007. – ISBN 978-3-540-74836-6
- [Paul Pop 2004] PAUL POP, Zebo P.: *Analysis and Synthesis of Distributed Real-Time Embedded Systems*. Kluwer Academic Publishers, 2004. – ISBN 1-4020-2872-5
- [Richter 2008] RICHTER, H.: *Projektbeschreibung CarRing II, TUCar V0 und V1*. TU Claustal. 2008. – URL http://www.in.tu-clausthal.de/uploads/media/Projektbeschreibung_TUCar_V5.pdf
- [Rokosch 2002] ROKOSCH, Uwe: *Airbag und Gurtstraffer*. Vogel Fachbuch, 2002. – ISBN 3-8023-1883-8
- [Steinbach 2008] STEINBACH, Till: *Ethernet als Bus für Echtzeitanwendungen im Automobil*, Hochschule für Angewandte Wissenschaften Hamburg, Studienarbeit, 2008
- [Strahl 2007] STRAHL, Christian: *Design und Entwicklung eines Realtimesystems für den Flugzeugsystembus AFDX*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2007
- [Systems 2007] SYSTEMS, GE Fanuc E.: *AFDX / Arinc 664 Protocol Tutorial*. GE Fanuc Embedded Systems. 2007. – URL <http://www.gefanucembedded.com/news-events/whitepapers/1955>