

Abschlussbericht zum Förderprogramm KMU-innovativ:
Informations- und Kommunikationstechnologie

SACHBERICHT ZUM VERWENDUNGSNACHWEIS TEIL I UND II

Förderkennzeichen: 16KIS0937K

Distributed, Extendable, Lightweight, Open, Reliable, service-oriented Architecture for next-gen mobility DELIA

Projektleitung	Dr. Leonid Lichtenstein
Administrative Ansprechpartner	Stephan Nießen (Leiter Administration)
Bevollmächtigter Unterzeichner	Roland Gerhards

Projektlaufzeit 01.02.2019 – 31.07.2022

ZAL Zentrum für Angewandte Luftfahrtforschung GmbH
Hein-Saß-Weg 22
21129 Hamburg

Projektträger:



Dimitar Kroushkov

Fabienne Stavropoulos (Administratives Projektmanagement)

VDI/VDE Innovation + Technik GmbH

Steinplatz 1
10623 Berlin

Verbundpartner:



Cornelia Brühlhart (Kordinatorin)

**Zentrum für Angewandte
Luftfahrtforschung GmbH**

Hein-Saß-Weg 22
21129 Hamburg



Leonardo Govoni

AED Engineering GmbH

Taunusstr. 51
80807 München



Jürgen Steinert

Solectrix GmbH

Fürther Str 244b
90429 Nürnberg



Universität Stuttgart

Jun. Prof. Björn Annighöfer

Universität Stuttgart

Pfaffenwaldring 27
70569 Stuttgart



Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Prof. Dr. Mathias Fischer

Universität Hamburg

Mittelweg 177
20148 Hamburg

Assoziierte Partner:



Oliver Döbertin

Airbus Operations GmbH

Kreetslag 10
21129 Hamburg



Michael Paschold

**Tyco Electronics Raychem
GmbH** a TE Connectivity Ltd.
company

Finsinger Feld 1;
85221 Ottobrunn

Versionsverzeichnis

Version	Datum	Bearbeiter	Beschreibung
0.9	28.10.22	Cornelia Brühlhart	Finaler Entwurf
1.0	14.01.23	Leonid Lichtenstein, Constantin Deneke	Ergänzungen, Qualitätskontrolle und Freigabe

Abkürzungsverzeichnis

ARP	Address Resolution Protocol
EVB	Converter Evaluation Board
FRER	Frame Replication and Elimination for Reliability
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
MAC	Media Access Control
OPC UA	Open Platform Communications Unified Architecture
OSI	Open Systems Interconnection
PHY	Physical Layer
TSN	Time-Sensitive Networking

Inhaltsverzeichnis

Versionsverzeichnis	3
Abkürzungsverzeichnis	3
1 Sachbericht Teil I: Kurzbericht	6
1.1 Ursprüngliche Aufgabenstellung und technischer Stand	6
1.2 Ablauf des Vorhabens	6
1.3 Wesentliche Ergebnisse	7
2 Aufgabenstellung und Ziele	8
2.1 Gesamtziel und Einordnung in die förderpolitischen Ziele	8
2.2 Kurzdarstellung der Projektpartner	8
3 Voraussetzungen und Ausgangssituation	10
4 Planung und Ablauf des Vorhabens	10
4.1 Projektstrukturplan	11
4.2 Arbeitspakete und Meilensteine	11
5 Stand der Wissenschaft und Technik	16
6 Kühlungskonzept	16
7 ZAL Endpoint	20
8 Erweiterung der DELIA Softwareplattform	22
9 Evaluierung von Kommunikationsprotokollen	23
10 Aufbau des DELIA Demonstrators	25
10.1 Erster Tischaufbau der DELIA-Komponenten	25
10.2 Konzeptionierung und Auswahl von Use-Cases	26
10.3 Entwicklung des DELIA Demonstrators	26
10.4 Softwarearchitektur	28
10.5 Datenweitergabe über das Netzwerk	29
10.6 Time-Sensitive Networking	30
11 Die wichtigsten Positionen des zahlenmäßigen Nachweises	33
11.1 Material	33
11.2 Personalkosten	33
11.3 Reisekosten	33
11.4 Abschreibungen auf vorhabenspezifische Anlagen	33
11.5 Sonstige unmittelbare Vorhabenkosten	33
12 Notwendigkeit und Angemessenheit der geleisteten Arbeit	33
13 Verwertbarkeit und voraussichtlicher Nutzen	34
14 Fortschritt auf dem Gebiet des Vorhabens bei anderen Stellen	35

15	Erfolgte Veröffentlichungen des Ergebnisses.....	35
	Literaturverzeichnis.....	36

1 Sachbericht Teil I: Kurzbericht

1.1 Ursprüngliche Aufgabenstellung und technischer Stand

Die technologische Zielsetzung des DELIA-Forschungsvorhabens war die Erforschung und Erprobung einer modularen verteilten Kommunikations- und Prozessorplattform für mobile Anwendungen und die Entwicklung möglicher Anwendungsfälle in der Flugkabine. Die heutige Bordelektronik mobiler Anwendungen wird in Teilen von verschiedenen Zulieferern aufgebaut und besteht somit häufig aus abgeschotteten Systemen, die unterschiedliche Kommunikationskanäle nutzen. Diese Kommunikationskanäle sind häufig nicht miteinander kompatibel und genügen den zukünftigen Anforderungen an Übertragungskapazitäten, Echtzeitfähigkeit und Interoperabilität nicht. Diese Struktur bremst Neuentwicklungen, denn Informationsflüsse zwischen den Hardware-Komponenten lassen sich nur schwer oder gar nicht realisieren und verhindern so eine einheitliche Softwareentwicklung.

Eine maßgebliche Fragestellung in DELIA richtet sich auf die Entwicklung einer innovativen Kommunikations- und Processing-Plattform. Moderne Mobilitätskonzepte erfordern dynamische Anpassungen der Funktionsverteilung und der Datenströme an die jeweilige Umgebung sowie den aktuellen Zustand des Netzwerkes. Im Verbundprojekt wurden daher verschiedene neuartige Netzwerkknoten entwickelt, welche diese Funktionalität abbilden können und darüber hinaus höhere Datenraten und Ausfallsicherheiten ermöglichen, bei gleichzeitiger Reduzierung des Systemgewichts

Für autonomes Fahren werden immer mehr Kameras, Sensoren und Steuergeräte eingesetzt, die eine große Datenmenge erzeugen. Auch in der Luftfahrtindustrie führt die zunehmende Digitalisierung aller Geräte und Prozesse an Bord zu steigenden Anforderungen an die Avionik- und Kabinennetzwerke. Um diese technologischen Herausforderungen bei sicherheits- und zeitkritischen Anwendungen in zukünftigen Mobilitätskonzepten angehen zu können, müssen Lösungen gefunden werden, die höhere Kapazitäten aufweisen, frei konfigurierbar sind und auf gängige offene Industriestandards setzen. Zur Erreichung der angestrebten Flexibilität und Zuverlässigkeit ist nach dem neusten Stand der Forschung eine modulare und verteilte Kommunikations- und Prozessorplattform, wie sie im DELIA-Projekt erforscht wurde, notwendig.

1.2 Ablauf des Vorhabens

Das DELIA-Projekt wurden im Zeitraum vom 01.02.2019 bis 31.07.2022 bearbeitet und war in drei Hauptarbeitspakete (Anforderungsentwicklung, Entwurf & Realisierung, Demonstration & Erprobung) unterteilt. Des Weiteren war das Projekt zeitlich gesehen in drei Blöcke gegliedert, welche jeweils durch einen Meilenstein abgeschlossen wurden.

Zum ersten Meilenstein wurde die Grundlage für die Vorentwicklung der Systemeinkomponenten aufgestellt. Im Rahmen der Festlegung der funktionalen Systemanforderungen wurde ein beispielhafter Nutzungsablauf der DELIA-Plattform und ihrer Shareholder beschrieben. Weiterhin wurden für jeden Shareholder User Stories abgeleitet und dokumentiert, bevor sie unterschiedlichen Anwendungsgebieten (z.B. Flugtaxi, Auto, Kabine, Industrie 4.0) zugeordnet wurden.

Mit Meilenstein M1 als Grundlage, konnte die Architektur des Gesamtsystems ausgelegt werden. Das Projektkonsortium hat auf Basis der Use Cases eine Schnittstellenspezifikation für das Gesamtsystem definiert und eine Topologie für das aufzubauende Netzwerk spezifiziert. Zum Meilenstein M2 folgte die Realisierung der DELIA-Plattform in Hard- und Software, die Auswahl der Sensoren und Aktoren, sowie deren Anbindung und Implementierung über das Netzwerk.

Zum Meilenstein M3 erfolgte die Anbindung der Sensoren und Aktoren an das Netzwerk, die Evaluierung möglicher Kommunikationsprotokolle und die Implementierung der Use-Cases. Hierbei erstellte die ZAL GmbH federführend ein Konzept für den Demonstrator.

1.3 Wesentliche Ergebnisse

Im Rahmen des DELIA-Projektes hat die ZAL GmbH unterschiedliche Aspekte einer modularen verteilten Kommunikations- und Prozessorplattform untersucht.

Zum einen wurden neue Kühlkonzepte für die in DELIA angestrebte Dezentralisierung der Einzelkomponenten analysiert und konzeptioniert. Unter Berücksichtigung der Einbauorte in der mobilen Plattform, der thermalen Bedingungen und der Use-Cases wurde eine Kühlung für die DELIA-Module entworfen. Insbesondere das Konzept einer Naturumlaufkühlung in der Flugzeugkabine wurde betrachtet. Es konnte gezeigt werden, dass ein stabiler Naturumlauf mit einer angenommenen Abwärme des DELIA-Moduls in Höhe von 30W möglich ist, aufgrund der ausreichenden Luftkühlung aber nicht zwingend notwendig ist.

Das durch die Partner entwickelte DELIA-Modul, welches als dezentrale Recheneinheit in der Netzwerkarchitektur dient, sieht kaum Schnittstellen für Sensoren und Aktoren vor. Aus diesem Grund hat die ZAL GmbH in diesem Projekt eine Elektronikkomponente entwickelt, welche diverse Schnittstellen zur Anbindung von Sensoren und Aktoren bietet – den sog. „ZAL Endpoint“. Das Konzept des Endpoints sieht die Erweiterung für spezifische Anwendungsfälle über Aufsteckplatinen vor, z.B. für die Anbindung von LED-Modulen.

Auf den Modulen selbst wurden Virtualisierungslösungen durch den Hypervisor XEN umgesetzt. Dies bedeutet, dass auf der gleichen Hardware mehrere simulierte „Computer“ gleichzeitig ausgeführt werden, ohne dass diese in Konflikt geraten oder sich Rechenressourcen „stehlen“, da diese fest zugewiesen sind. So können sicherheitskritische Funktionen, wie z.B. Rauchmelder, neben unkritischen, wie Inflight-Entertainment, auf den gleichen Modulen abgebildet werden. Die DELIA-Module nutzen das Open-Source Betriebssystem Alpine Linux, das eine hohe Ressourceneffizienz bietet.

Des Weiteren hat die ZAL GmbH drei Übertragungsprotokolle evaluiert: MQTT, OPC UA PubSub und Zenoh. Ergebnis der Evaluierung war, dass durch den gezielten Datenverkehr sowie den schlanken und simplen Aufbau das MQTT-Protokoll am besten für das DELIA-Einsatzszenario geeignet ist.

Die gesammelten Erkenntnisse wurden in einem realitätsnahen Testaufbau zusammengeführt und erfolgreich für die Anwendung der Kabinenbeleuchtung getestet. Die Steuerungsmodule übernehmen hierbei die Lichtsteuerung einzelner LED-Leisten, deren LEDs in dem Demonstrationsszenario dynamisch angesteuert wurden. Der ZAL Endpoint wurde zur Anbindung der LED-Leisten an die Module eingesetzt. Für die Kommunikation zwischen den einzelnen Modulen und Endpunkten ist Time-Sensitive Networking (TSN) ein integraler Bestandteil der Architektur des DELIA-Demonstratoraufbaus der ZAL GmbH. Es konnte gezeigt werden, dass bei der Trennung eines Moduls ein anderes Modul dessen Funktionen zuverlässig übernimmt und dass auch bei einer erhöhten Netzwerkauslastung keine Störungen wie der Ausfall der Beleuchtung oder eine verzögerte Ansteuerung der LED-Leisten auftreten.

Das Forschungsprojekt DELIA hat wichtige Grundlagen für die Entwicklung eines innovativen Kabinennetzwerkes geschaffen. Die Forschungsergebnisse zeigen, dass verteilte Systeme schnelle und zuverlässige Datenübertragungswege ermöglichen. Durch den Einsatz von TSN-Technologien kann die verzögerungsfreie Übertragung auch bei hoher Datenlast oder Ausfall einer Systemkomponente sichergestellt werden. Mit dem ZAL Endpoint ist außerdem ein vielseitiges Endgerät entstanden, das auch in weiteren Industrie- und Forschungsprojekten weiterverwendet werden kann. Durch konsequente Verwendung offener Industriestandards und einheitlicher Softwareentwicklung kann zukünftig an die gewonnen Erkenntnisse nahtlos angeknüpft werden.

2 Aufgabenstellung und Ziele

Technologische Innovationen, Standardisierung und interdisziplinäre Forschung ermöglichen die kontinuierliche Entwicklung neuer Plattformen und Werkzeuge, für den Einsatz in der Kabine von Morgen. Kooperationen zwischen verschiedenen Unternehmen und Partnern beschleunigen hierbei das Erforschen neuer technologischen Möglichkeiten und bilden die Brücke zwischen Industrie und Forschung. Die Ergebnisse in dem Projekt DELIA spiegeln diese Arbeit dar, da hier die Expertise der ZAL GmbH Ingenieure mit Teilnehmern aus dem akademischen und industriellen Sektor vereint werden. Gemeinsam wurde bei dem Projekt DELIA eine innovative Kommunikations- und Verarbeitungsplattform geschaffen, welche sowohl eine dynamische System-Rekonfiguration und Datenstrom-Umverteilung ermöglicht. Dank dieser kann die Resilienz eines Kabinennetzwerkes gestärkt und eine Treibstoffersparnis erreicht werden, da verschiedene Datenströme sich Pfade teilen können.

Hierbei wurden sowohl verschiedene neue Technologien wie TSN und Virtualisierung auf einem eingebetteten System untersucht, als auch ein finaler Demonstrator für das Gesamtsystem erstellt, der das geschaffene Wissen vereint.

2.1 Gesamtziel und Einordnung in die förderpolitischen Ziele

Eine maßgebliche Fragestellung in DELIA richtet sich auf die Entwicklung einer innovativen Kommunikations- und Processing-Plattform. Moderne Mobilitätskonzepte erfordern dynamische Anpassungen der Funktionsverteilung und der Datenströme an die jeweilige Umgebung sowie den aktuellen Zustand des Netzwerkes. Im Verbundprojekt wurden daher verschiedene neuartige Netzwerkknoten entwickelt, welche diese Funktionalität abbilden können und darüber hinaus höhere Datenraten und Ausfallsicherheiten ermöglichen, bei gleichzeitiger Reduzierung des Systemgewichts. Um die gewonnenen technologischen Fähigkeiten im Anschluss an dieses Forschungsprojekt industriell verwerten zu können, wurde eine weitere Komponente entwickelt: Der ZAL Endpoint, welcher auch einen essenziellen Bestandteil des Demonstrator-Aufbaus im Rahmen dieses Projektes repräsentiert.

Um diese technologischen Herausforderungen bei sicherheits- und zeitkritischen Anwendungen in zukünftigen Mobilitätskonzepten angehen zu können, müssen Lösungen gefunden werden, die höhere Kapazitäten aufweisen, frei konfigurierbar sind und auf gängige offene Industriestandards setzen. Zur Erreichung der angestrebten Flexibilität und Zuverlässigkeit ist nach dem neusten Stand der Forschung eine modulare und verteilte Kommunikations- und Prozessorplattform sehr gut geeignet. Vorteile hierbei sind, zum einem die hohe Flexibilität beim Aufbau und Re-Konfigurieren des Netzwerkes, als auch die Möglichkeit von Pfadredundanz und Ausfallsicherheit im Falle einer Notfallsituation in der Flugkabine. Vor einem industriellen Einsatz, besonders in der Luftfahrt, müssen diese Technologien jedoch zunächst zu einem höheren Technologiereifegrad geführt werden.

Die technologische Zielsetzung dieses risikoreichen, vorwettbewerblichen Forschungsvorhabens war daher die Erforschung und Erprobung einer modularen verteilten Kommunikations- und Prozessorplattform für mobile Anwendung, und die Entwicklung möglicher Anwendungsfälle in der Flugkabine.

2.2 Kurzdarstellung der Projektpartner

Das **Zentrum für Angewandte Luftfahrtforschung (ZAL)** ist eines der modernsten Forschungszentren für die zivile Luftfahrt. Die ZAL GmbH fungiert hierbei mit mehr als 60 Mitarbeiter:innen als technologische Basis für das Technologiezentrum. Die ZAL GmbH unterstützt seine Kunden bei der schnellen Realisierung und Erprobung von neuen Technologien für die Luftfahrt. So zielen zum Beispiel diverse Industrieprojekte auf die Vorentwicklung und

konzeptionelle Integration der physikalischen Schicht (OSI Level 1) eines neuartigen Avionik-Glasfasernetzwerks ab. Darüber hinaus verfügt die ZAL GmbH über eigene Forschungsaktivitäten, um ausgewählte Potenzialtechnologien voranzutreiben. In dieses Projekt wurden zum Beispiel Technologien zur passiven Elektronik-Flüssigkühlung eingebracht. Die ZAL GmbH hat im Jahr 2020 einen Umsatz von 14,7 Mio. € erlöst und ist eng vernetzt mit mehreren großen europäischen Luftfahrtunternehmen.

Die **AED Engineering GmbH** mit Sitz in München wurde 2006 gegründet und beschäftigt derzeit rund 60 Mitarbeiter. Kerngeschäft von AED sind einerseits die interne Forschung und Entwicklung im Bereich drahtlose und drahtgebundene Kommunikationsnetzwerke sowie Consulting und Entwicklungsprojekte für die Automobil- und Luftfahrtbranche. Es wurden bereits erfolgreich mehrere Forschungsprojekte in den Bereichen drahtgebundene und drahtlose Kommunikation, Hardware- und Firmware-Entwicklung sowie PHY- und MAC-Entwicklung im Rahmen von öffentlich geförderten Projekten durchgeführt. AED ist mit Kunden und Partnern wie BMW, Audi oder Airbus sowohl in der Automobil- als auch der Luftfahrtindustrie eng vernetzt und daher bestens vertraut mit den Technologien, Anforderungen, Normen und Trends in diesen Branchen. Insofern bietet AED Engineering sehr gute Voraussetzungen auch dieses Vorhaben zusammen mit den Partnern erfolgreich umzusetzen.

Die **Solectrix GmbH** ist seit über 10 Jahren Entwicklungsdienstleister für Embedded Systeme. Mit über 90 Mitarbeitern entwickelt Solectrix Kundengeräte und Applikationen von der Elektronik bis zur Anwendungssoftware. Die Kunden kommen unter anderem aus den Bereichen Imaging, Medizintechnik, Automotive und Industrial. Solectrix bietet sowohl Komplettentwicklungen an als auch einzelne Entwicklungsschritte je nach Kundenwunsch und Anforderung. Gemeinsam mit der Solectrix Systems bieten wir Produktion und Produktionsbetreuung im Kleinseriensegment bis 1.000 Stück an. Die Solectrix arbeitet mit einem Team aus hoch qualifizierten Entwicklungsingenieuren sowie Technikern und Produktionsmitarbeitern. Dadurch ist Solectrix in der Lage, vor allem Kunden mit komplexen Entwicklungsaufträgen aus dem High-End-Bereich anzusprechen und zufrieden zu stellen. Eine flache Hierarchiestruktur im Unternehmen ermöglicht kurze Kommunikationswege und schnelle Reaktionszeiten auf Kundenwünsche. Die Verbindung aus Entwicklungsdienstleistung und anschließender Produktionsüberführung und Produktion sichert dem Kunden einen gleichbleibend hohen Standard auch in der Produktionsphase sowie einen Zeit- und Kostenvorteil.

Die Forschungsschwerpunkte des Instituts für Luftfahrtsysteme (ILS) der **Universität Stuttgart** sind die Verbesserungen von Methoden und Werkzeugen zur Auslegung, Implementierung, Integration, Test und Dokumentation von sicherheitskritischen digitalen Luftfahrtsystemen. Seit der Gründung wurden wesentliche Beiträge zur automatischen Instanziierung von Fly-by-wire-Systemen in Kleinflugzeugen, bis hin zur Flugerprobung, geleistet (Flexible Avionik Plattform) sowie die Entwicklung von Drive-by-Wire-Demonstratoren. Weitere Schwerpunkte sind selbst-konfigurierende Avionik Systeme (Adaptive Avionik Plattform) und die modell-basierte Auslegung von Avionik Plattformen. Weiter hat das ILS eine eigene Avionik Hardware nebst Betriebssystem entwickelt, welche in zahlreichen Forschungsprojekten zum Einsatz kommen. Zurzeit sind 25 wissenschaftliche Mitarbeiter und Doktoranden beschäftigt. Hinzu kommen fünf Festangestellte. In der Ausbildung vermittelt das ILS an Luft- und Raumfahrtingenieure die Grundlagen der Informatik und Embedded-Systeme, sowie Methoden der Sicherheitsanalyse, Redundanz, Meta-Modellierung und Modelltransformation. Am ILS existieren spezialisierte Avionik Labore mit elektrischen und physikalischen Testkomponenten sowie Flugsimulatoren und Systemfunktionssimulationen.

Die Arbeitsgruppe IT-Sicherheit und Sicherheitsmanagement am Fachbereich für Informatik der **Universität Hamburg** (geleitet von Prof. Dr. Mathias Fischer) arbeitet auf der Schnittstelle zwischen Netzen und klassischer IT-Sicherheit. Forschungsthemen sind resiliente, verteilte

Systeme, der Schutz kritischer Infrastrukturen, Angriffserkennung und -behandlung in großen Netzinfrastrukturen und fortschrittliche Malware wie beispielsweise Botnetze. Aktuell in der Gruppe bearbeitete Forschungsprojekte befassen sich mit dem Aufbau sicherer Sensornetze (LFF-Projekt SANE) sowie mit dem Schutz zukünftiger industrieller Produktionsinfrastrukturen (BMW-Projekt IMMUNE).

Die **Airbus Operations GmbH** ist ein weltweit führender Flugzeughersteller mit der modernsten und effizientesten Familie von Verkehrsflugzeugen. Die Produktpalette umfasst vier Flugzeug-Familien vom Single-Aisle-Flugzeug mit 100 Sitzen bis hin zum größten Verkehrsflugzeug aller Zeiten - der A380 mit zwei Passagierdecks. Neue Lösungen für zukünftige Avionik-Architekturen werden insbesondere für die zukünftigen Flugzeugprogramme (CityAirbus) sowie die Verbesserung bestehender Produkte (z.B.: A320 Neo plus) dringend benötigt.

Tyco Electronics Raychem GmbH, ein Unternehmen der TE Connectivity Ltd., ist ein weltweit führendes Technologieunternehmen mit einem Umsatz von 12 Milliarden US-Dollar. Unser Engagement für Innovation ermöglicht Fortschritte in verschiedenen Bereichen: Transport, industrielle Anwendungen, Medizintechnologie, Energietechnik, Datenkommunikation und für das Zuhause. TE bietet eine unübertroffene Bandbreite an Verbindungs- und Sensorlösungen, die sich unter anspruchsvollen Bedingungen bewährt haben. So helfen wir, unsere Welt zuverlässiger, sauberer, intelligenter und vernetzter zu machen. Mit 75.000 Mitarbeitern, darunter 7.000 Entwicklungsingenieuren, arbeiten wir mit Kunden aus fast 150 Ländern in allen führenden Industriebranchen zusammen. Unsere Überzeugung ist auch unser Motto: EVERY CONNECTION COUNTS.

3 Voraussetzungen und Ausgangssituation

Die heutige Bordelektronik mobiler Anwendungen wird in Teilen von verschiedenen Zulieferern aufgebaut und besteht somit häufig aus abgeschotteten Systemen, die unterschiedliche Kommunikationskanäle nutzen. Diese Kommunikationskanäle sind häufig nicht miteinander kompatibel und genügen den zukünftigen Anforderungen an Übertragungskapazitäten, Echtzeitfähigkeit und Interoperabilität nicht. Diese Struktur bremst Neuentwicklungen, denn Informationsflüsse zwischen den Hardwarekomponenten lassen sich nur schwer oder gar nicht realisieren und verhindern so eine einheitliche Softwareentwicklung. Um Technologien zu untersuchen, die in der digitalen Kabine Anwendungsfelder finden können, helfen Integrationstests in Laboren, wie sie z.B. an der ZAL GmbH angewendet werden.

Die Arbeitsgruppe *Data & Power Networks* der ZAL GmbH bringt langjährige vertiefte Kenntnisse aus dem Fachgebiet der Verteilten Systeme und Systemintegration von Sensorik ein. Des Weiteren bietet das Technologielabor der ZAL GmbH die Möglichkeit, einen State-of-the-Art Demonstrator aufzubauen, in welchem unterschiedliche Technologien vereint interagieren. Darüber hinaus bedient die ZAL GmbH forschungsfeldübergreifende Kompetenzen, da z.B. weitere Arbeitsgruppen am ZAL, wie z.B. *Advanced Materials*, die In-house Produktion von 3-D gedruckten Gehäusen für die entwickelten DELIA-Komponenten ermöglicht.

4 Planung und Ablauf des Vorhabens

In diesem Kapitel ist der detaillierte Arbeitsplan des Vorhabens der ZAL GmbH beschrieben. Das Kapitel gliedert sich in den Projektstrukturplan, die Beschreibung der ZAL-Arbeitspakete und Meilensteine. Wie in den technologischen Zielen definiert und in Einklang mit der Verwertungsstrategie, fokussiert sich die ZAL GmbH auf das übergreifende Systemengineering und die Konfiguration, sowie den Aufbau eines entsprechenden DELIA Demonstrators.

4.1 Projektstrukturplan

Die ZAL GmbH leistet einen Beitrag zu den Arbeitspaketen 1.1 Funktionale Anforderungen, 1.3 Systemarchitektur, 2.1 Plattform-Hardware & Firmware, 3.1 Systemintegration und 3.2. Aufbau eines Verbunddemonstrators. Darüber hinaus fungiert die ZAL GmbH als Verbundführer und übernimmt entsprechend das Arbeitspaket Verbundprojektmanagement (AP0). Die Nummerierung der Arbeitspakete wurde aus dem Verbundvorhaben übernommen. Innerhalb der Arbeitspakete wurden von den beteiligten Kooperationspartnern einzelne Unterarbeitspakete im maximalen Zeitumfang von 6 Personalmonaten gebildet. Für die ZAL GmbH entstehen dadurch die folgenden Unterarbeitspakete:

- 0.ZAL.1 Verbundprojektmanagement
- 0.ZAL.2 Öffentlichkeitsarbeit
- 1.1.ZAL.1 Funktionale Anforderungen
- 1.3.ZAL.1 Auslegung der Systemarchitektur
- 1.3.ZAL.2 Modellbeschreibung der Systemarchitektur
- 2.1.ZAL.1 Kühlkonzept
- 2.1.ZAL.2 Design Kühllösung
- 2.1.ZAL.3 Realisierung Kühllösung
- 2.1.ZAL.4 Auswahl von Sensoren und Aktoren
- 2.1.ZAL.5 Anbindung von Sensoren und Aktoren
- 2.1.ZAL.6 Datenweitergabe über das Netzwerk implementieren
- 2.1.ZAL.7 Implementieren erster Use-Cases
- 3.1.ZAL.1 Konzeptentwicklung Verbunddemonstrator
- 3.2.ZAL.1 Aufbau Verbunddemonstrator
- 3.2.ZAL.2 Test Verbunddemonstrator.

In den einzelnen Arbeitspaketen der ZAL GmbH wurden die Arbeiten des projektspezifischen Projektmanagements für das Einzelvorhaben der ZAL GmbH integriert. Sie umfassen die detaillierte Steuerung der Arbeitspakete bezüglich der fach-, termin-, kosten- und förderrichtliniengerechten Durchführung. Die Zeitaufwände als Verbundführer für alle Projekttreffen wurden dem AP 0.ZAL.1 zugeschlagen.

4.2 Arbeitspakete und Meilensteine

Das Verbundprojektmanagement in Arbeitspaket 0.ZAL.1 ist eine begleitende Aktivität, welche sich über die gesamte Projektlaufzeit erstreckt.

Im Berichtszeitraum wurden monatlich Videokonferenzen auf Verbundebene durchgeführt. Im Berichtszeitraum fanden zusätzliche fachliche Videokonferenzen und acht Projekttreffen statt.

Arbeitspaket 0.ZAL.1 beinhaltet die Öffentlichkeitsarbeit für das Verbundprojekt. In diesem Arbeitspaket hat die ZAL GmbH im Berichtszeitraum die zu Projektstart erstellte Internetseite¹ kontinuierlich gepflegt. Des Weiteren hat die ZAL GmbH auf dem Deutschen Luft- und Raumfahrtkongress 2020 das Verbundprojekt DELIA durch ein Poster und eine entsprechende Posterpräsentation dargestellt. News Artikel auf der zal.aero Homepage der ZAL GmbH lieferten außerdem weitere Informationen zu dem Projektfortschritt in DELIA.

Meilenstein M1

Vorangehend zu dem Meilenstein M1 wurde in den Arbeitspaketen die Grundlage für die Vorentwicklung der Systemeinkomponenten aufgestellt. Im Rahmen der Festlegung der

¹ <https://delia-project.com/>

funktionalen Systemanforderungen wurde ein beispielhafter Nutzungsablauf der DELIA Plattform und ihrer Shareholder beschrieben. Im weiteren Verlauf wurden dann für jeden Shareholder User Stories abgeleitet und dokumentiert, bevor sie unterschiedlichen Anwendungsgebieten (Flugtaxi, Auto, Kabine, Industrie 4.0) zugeordnet wurden.

Kategorie	Industrie 4.0	Als...	möchte ich...	datenintensiv	sicherheitskritisch (High Data)	zeitkritisch	personellenrelevant	C	I	A	Priorität	Kategorie
X	1	1	Passagier	Videos aus dem Internet streamen	x		x				x	Infotainment
X	1	8	Passagier	während des Transports Videokonferenzen halten (~6 MBit/s)		x	x				x	
X			Passagier	Informationen erhalten							x	
	2		Passagier	den Fahrzeugstatus abfragen		x					x	Infotainment
	1	1	Passagier	Informationen zum Akkustand/Restreichweite des Fahrzeugs erhalten		x			x	x	x	
	2		Passagier	Informationen zu Fehlern/Alarmer des Fahrzeugs erhalten		x					x	
X			Passagier	Sicherheit während des Transports							x	Safety / Secur
X	1		Passagier	frühzeitige Fehlererkennung		x			x		x	
X	2		Passagier	auf Fehler im System hingewiesen werden, die mein Eingreifen erfordern							x	
X	4		Passagier	Notfallmanagement		x	x			x	x	Safety / Secur
X			Passagier	Ressourcenmanagement um kritische Systemkomponenten laufen zu lassen		x			x	x	x	
X	3	1	Passagier	Redundanz im System		x				x	x	
X	X	2	Passagier	TSN mit dynamischer Konfiguration (IEEE 802.1Qca & IEEE 802.1CB IEEE 802.1Qcc)		x	x				x	Anpassbarkeit
X	X		Integrator	das System anpassen					x		x	
X	X	1	Integrator	dem System Module hinzufügen		x					x	
X	X	2	Integrator	Module vom System entfernen		x					x	Anpassbarkeit
X	X	3	Integrator	auf dem System neue Funktionen implementieren		x					x	

Figure 1 Auszug der User Stories

Anschließend wurde eine Klassifizierung der User Stories, z.B. in sicherheitskritisch, datenintensiv oder zeitkritisch, vorgenommen, um übergreifende Anforderungen der User Stories herauszuarbeiten und den Projektpartnern als Vorlage bereitgestellt. Die dazu individuellen Beiträge der Projektpartnern wurden dann von der ZAL GmbH zusammengeführt, hinsichtlich ihrer Priorität eingestuft und fortlaufend aktualisiert, um den Bearbeitungsstand im Projekt zu reflektieren

Meilenstein M2

Mit Meilenstein M1 als Grundlage, konnte in Arbeitspaket 1.3.ZAL.1 die Architektur des Gesamtsystems ausgelegt werden. Ziel hierbei war es, auf Basis der hervorgehenden Use Cases eine Schnittstellenspezifikation für das Gesamtsystem zu definieren und eine Topologie für das aufzubauende Netzwerk zu entwickeln. Figure 2 zeigt die entwickelte Systemarchitektur. Hierbei sind fünf DELIA-Module abgebildet, welche Netzwerk- und Rechenfunktionalitäten auf sich vereinen. Diese bilden eine Ringtopologie, um den Ausfall einer Netzwerkverbindung kompensieren zu können.

Da die Konzeptentwicklung der Projektpartner ergeben hat, dass die DELIA-Module nur eine begrenzte Anzahl an Anschlüssen für Sensoren und Aktoren haben, ergab sich die Notwendigkeit zusätzlich sogenannte Endpunkte in der Systemarchitektur zu berücksichtigen. Durch diese werden Sensoren und Aktoren an das Netzwerk angebunden. Endpunkte sind dabei deutlich weniger leistungsfähig, stromsparender und kleiner als die DELIA-Module selbst.

An die Module sind jeweils mehrere Endpunkte angeschlossen, welche die für die Use-Cases nötige Peripherie anbinden. Diese Endpunkte sollen dabei in der Passenger Service Unit (PSU) und im Sitz platziert sein. Hiermit wird sowohl der Einsatz im Flugtaxi als auch in der Flugzeugkabine abgebildet. Die Endpunkte an einem DELIA-Modul unterscheiden sich und repräsentieren so zum Beispiel die Anbindung von Sensoren und Aktoren im Bereich der Bordküche eines Flugzeuges. In einem Flugtaxi wäre vermutlich der Aufbau für vier Sitze identisch, zusätzliche Endpunkte könnten zum Beispiel im Bereich der Bordelektronik verbaut werden, um dort Überwachungs- und Steuerungsfunktionen zu realisieren.

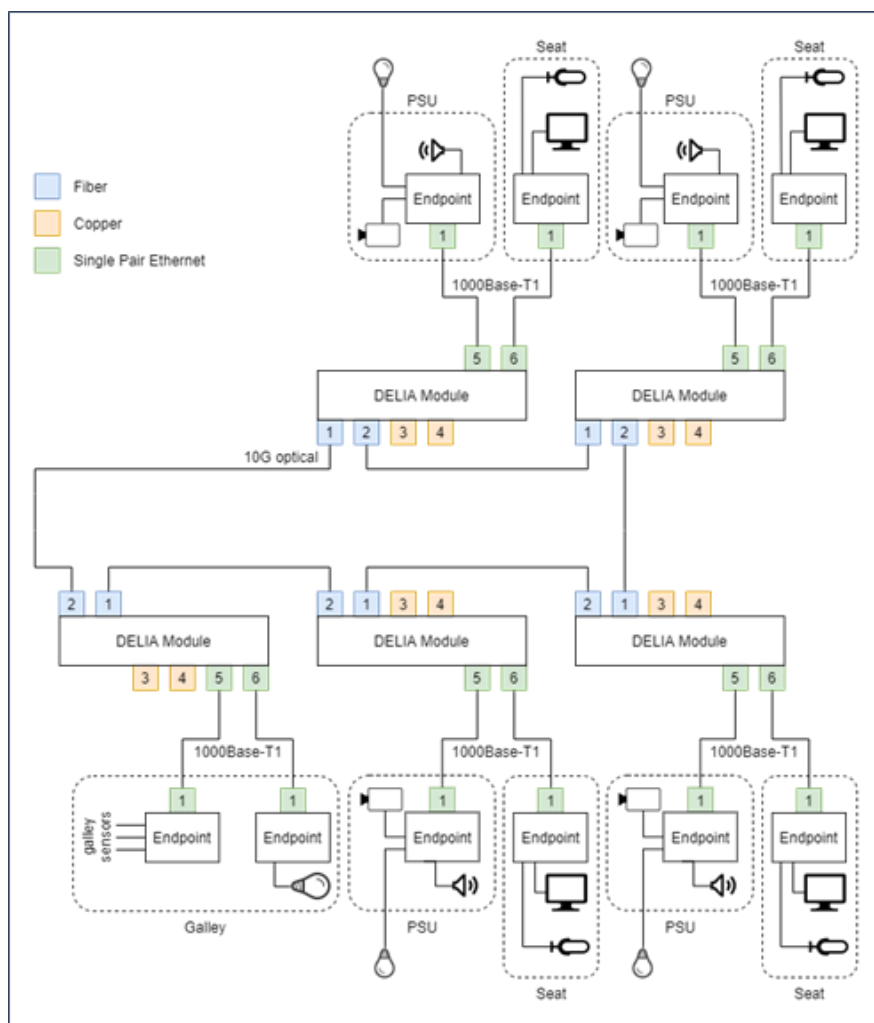


Figure 2: Systemarchitektur Konzept

Auf den DELIA-Modulen selbst wurde eine Virtualisierungslösung umgesetzt, welche es ermöglicht, mehrere Funktionalitäten parallel auszuführen. Dies erfüllt eines der Projektziele: den gemeinsamen Betrieb von sicherheitskritischen und nicht sicherheitskritischen Funktionalitäten auf einem Modul, ohne dass eine gegenseitige Beeinflussung gegeben ist.

Aufbauend auf dieser Virtualisierung ist der Einsatz von Containern für nicht sicherheitskritische Funktionalitäten angedacht, um Integratoren die Entwicklung und Orchestrierung neuer, zusätzlicher Funktionalitäten zu erleichtern. Anschließend dazu wurde eine Systemarchitektur als MBSE-Modell abgebildet, um hiermit Einzelkomponenten, Schnittstellen und Kommunikationsströme zu modellieren.

Arbeitspaket 2.1.ZAL.1 umfasste die Entwicklung neuer Kühlkonzepte für die in DELIA angestrebte Dezentralisierung der Einzelkomponenten. Ziel war es hierbei, unter Berücksichtigung der Einbauorte in der mobilen Plattform, der thermalen Bedingungen und der Use-Cases eine Kühlung für die DELIA-Module zu entwerfen (siehe Kühlungskonzept Kapitel 6).

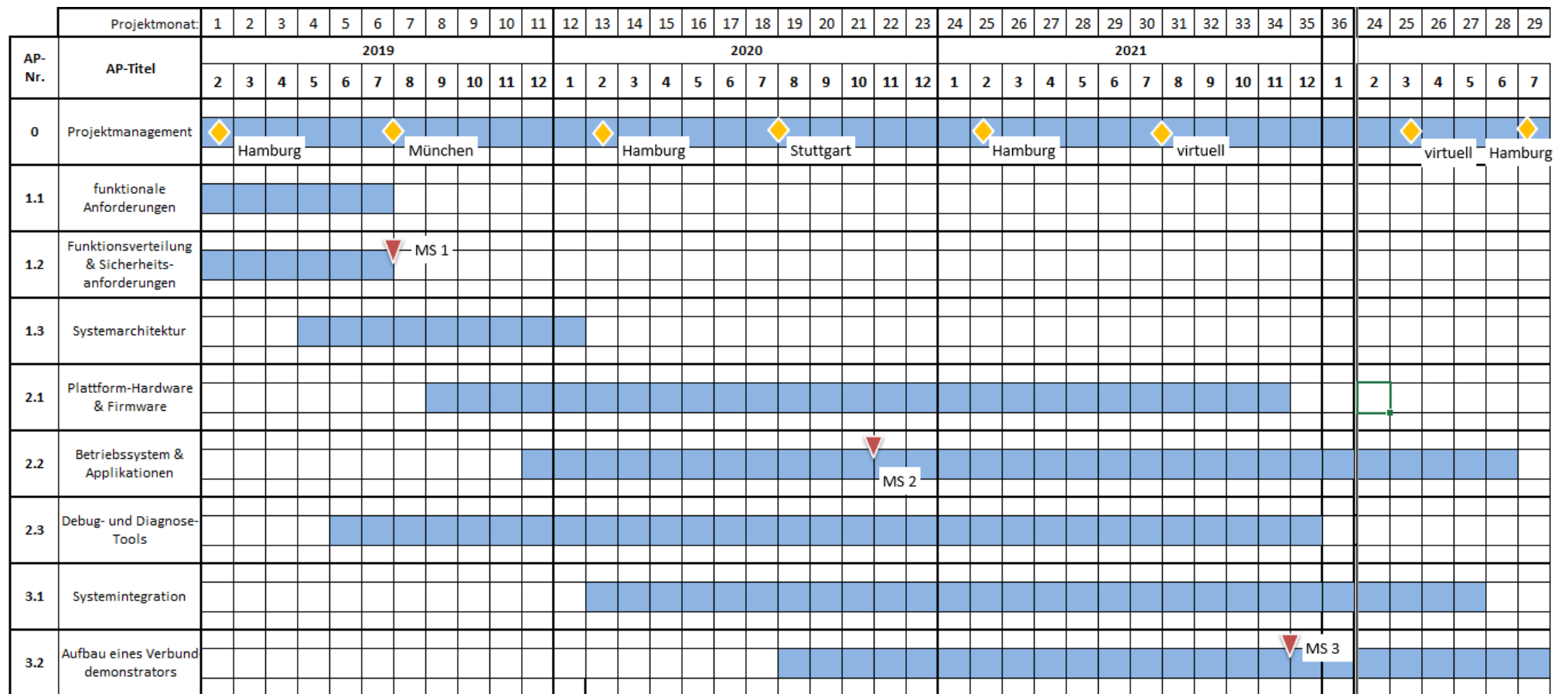
Abschließend zum Meilenstein M2 folgte darauf die Auswahl von Sensoren und Aktoren, sowie deren Anbindung und Implementierung über das Netzwerk (siehe ZAL Endpoint Kapitel 7).

Meilenstein M3

Zum Meilenstein M3 erfolgte die Anbindung von Sensoren und Aktoren an das Netzwerk, das Evaluieren möglicher Kommunikationsprotokolle, wie z.B. OPC UA PubSub oder Zenoh², und das Implementieren erster Use-Cases. Hierbei erstellte die ZAL GmbH federführend ein Konzept für den Demonstrator, als Basis für die nachfolgenden Arbeitspakete 3.2.ZAL.1 und 3.2.ZAL.2. (siehe DELIA Demonstrator Kapitel 10).

In Zusammenarbeit mit den Projektpartnern wurde hierbei erfolgreich eine Softwareplattform entwickelt, bei welcher die ZAL GmbH die Aufgabe übernommen hat, den Hypervisor XEN als Virtualisierungslösung auf den DELIA-Modulen umzusetzen. Dafür wurde ein Skript der Projektpartner Solectrix zum Erstellen eines SD-Karten-Images angepasst, um XEN beim Booten einzubinden. Hiermit hatten die Partner die Möglichkeit, ein angepasstes und einheitliches Softwaresystem für die DELIA-Module zu generieren und aktuell zu halten.

²<https://zenoh.io/>



5 Stand der Wissenschaft und Technik

Für autonomes Fahren werden immer mehr Kameras, Sensoren und Steuergeräte eingesetzt, die eine große Datenmenge erzeugen. Zusammen mit dem immer aufwendiger werdenden Fahrgast-Infotainment mit WiFi-Access-Points im Fahrzeug oder zukünftiger 5G-Konnektivität ist in den nächsten Jahren mit einer notwendigen Gesamtbandbreite im Fahrzeug jenseits von 10Gbit/s zu rechnen. Da Standard-Ethernet u.a. aufgrund des Kabelgewichts und des Abstrahlverhaltens ungeeignet für die Verwendung in Fahrzeugen ist und der 2016 finalisierte Automotive-Ethernet-Standard 1000BASE-T1 (IEEE 802.3bp) zu wenig Datenrate bietet, wird in der IEEE 802.3ch „Multi-Gig Automotive Ethernet PHY Task Force“ derzeit an einem Standard für 2,5Gbit/s, 5Gbit/s und 10Gbit/s Zwei-Draht-Ethernet gearbeitet. Der fehlende Determinismus wird zukünftig durch die Standardisierung der Time-Sensitive Networking Task Group innerhalb der IEEE 802.1 erreicht werden. Auch in der Luftfahrtindustrie führen immer bessere Navigationssysteme, Flugführungssysteme, Health & Usage Monitoring Systeme, Inflight Entertainment Systeme, Connectivity und vor allem auch die zunehmende Digitalisierung aller Prozesse an Bord zu steigenden Anforderungen an die Avionik- und Kabinennetzwerke. Urban Air Mobility verwischt die Grenzen zwischen Fahr- und Flugzeug und erfordert eine einheitliche, dienstorientierte Architektur, die sich auch im laufenden Betrieb neu orchestrieren und konfigurieren lässt, um zwischen Flug- und Bodenmodus zu wechseln.

Die in DELIA erzielten Ergebnisse, um die technologischen Herausforderungen bei zeit- und sicherheitskritischen Anwendungen in zukünftigen Mobilitätskonzepten zu adressieren, setzen auf offene Industriestandards und einheitliche Softwareentwicklungen. Um alle notwendigen Funktionen, Sensoren und Aktoren ansprechen zu können, werden deshalb sowohl ein echtzeitfähiges Glasfasernetz und programmierbare Schaltungen (FPGAs) untersucht, um zu evaluieren, in welchem Anwendungsfall diese den zukünftigen Anforderungen in Mobilitätskonzepten beitragen können.

6 Kühlungskonzept

Früh im Verlaufe des DELIA Projektes, zum Arbeitspaket 2.1.ZAL.1, wurden neue Kühlkonzepte für die in DELIA angestrebte Dezentralisierung der Einzelkomponenten analysiert und konzeptioniert. Ziel war es hierbei, unter Berücksichtigung der Einbauorte in der mobilen Plattform, der thermalen Bedingungen und der Use-Cases eine Kühlung für die DELIA-Module zu entwerfen. Das Arbeitspaket wurde von September 2019 bis einschließlich Juni 2020 bearbeitet.

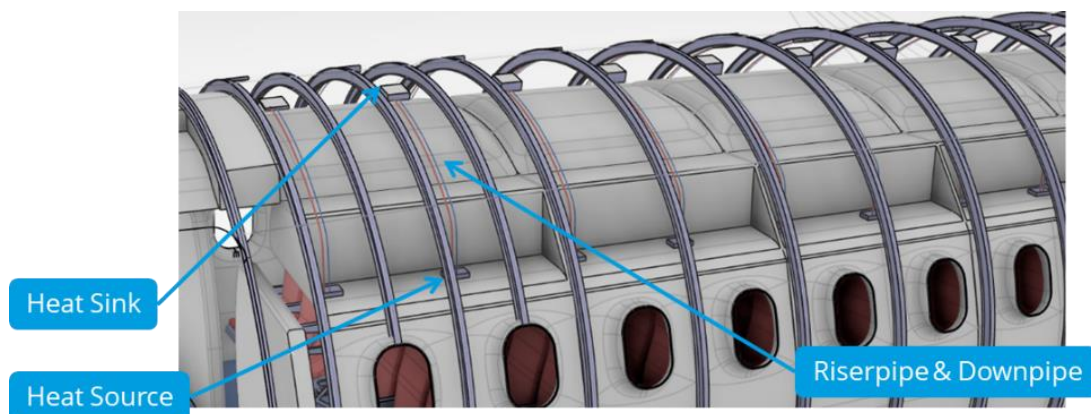


Figure 3: Aufbau einer möglichen Naturumlaufkühlung im Flugzeug

Insbesondere das Konzept einer Naturumlaufkühlung in der Flugzeugkabine wurde betrachtet. Figure 3 zeigt einen beispielhaften Aufbau im 3D-Modell. Dabei stellt die gezeigte Hitzequelle (Heat Source) ein sog. DELIA-Modul dar. Vom DELIA-Modul aus laufen Steigrohr und Fallrohr zu einer

Wärmesenke, die durch einen Kondensator z.B. an der Flugzeugaußenhaut gegeben sein kann. Am DELIA-Modul selbst kommt ein Verdampfer zum Einsatz.

In der konkreten Umsetzung der Naturumlaufkühlung wurden diverse Schwierigkeiten identifiziert, welche eine Umsetzung komplex und zum jetzigen Zeitpunkt unsicher erscheinen lassen. Hierzu gehört z.B. das Konzept der Gehäuse im MiniMRP Formfaktor auf Basis des Industriestandards ARINC 836A, welche die DELIA-Module beherbergen. Bei diesen Gehäusen wird ein Ansatz verfolgt, bei dem ein geschlossener Einschub in einen Einbaurahmen eingeschoben wird und sich dort durch eine Einführung selbsttätig mit den vorhandenen Anschlüssen verbindet. Der Kondensator des Naturumlaufkühlers müsste prinzipbedingt an dem Einbaurahmen angebracht sein, was demzufolge immer eine, wenn auch dünne, Luftschicht zwischen Kondensator und Einschub, in dem die Wärme abgegeben wird, zur Folge hätte. Da die Wärmeleitfähigkeit von Luft signifikant niedriger ist als dies z.B. für Aluminium der Fall ist, wäre die Leistungsfähigkeit des Systems hinsichtlich der abgeführten Wärme begrenzt. Der Einsatz von sog. Gap Füllern zum Herstellen einer Verbindung mit höherer Wärmeleitfähigkeit ist nicht praktikabel, da diese dem Konzept des einfachen Austausches der als Einschübe konzipierten DELIA-Module widersprechen würde. Zusätzlich erschwert der Aufbau des DELIA-Moduls aus Carrier, Processing Board und IO-Board die Wärmeabfuhr. Der Carrier liegt immer mittig und beherbergt auf gegenüberliegenden Seiten Processing Board und IO-Board. Hierdurch ergibt sich, dass die Wärme im genutzten Einschubgehäuse an gegenüberliegenden Seiten entsteht. Durch einen einzelnen Naturumlaufkühler lässt sich also nicht die komplette Wärme abführen, was der Stabilität des Prozesses aufgrund der geringen Wärmeleistung entgegenstehen kann.

Die in Arbeitspaket 2.1.ZAL.1 beschriebene Naturumlaufkühlung wurde in der Software Dymola Behavior Modeling auf Basis der Simulationssprache Modelica umgesetzt und für den konkreten Anwendungsfall ausgelegt. Figure 4 zeigt eine grafische Darstellung aus Dymola mit Verdampfer, Steigrohr, Kondensator und Fallrohr. In Arbeitspaket 2.1.ZAL.2 wurde das Kühlkonzept aus Arbeitspaket 2.1.ZAL.1 ausgelegt. Ziel war es, eine Kühllösung für die DELIA-Module numerisch zu validieren.

Es konnte gezeigt werden, dass ein stabiler Naturumlauf mit einer angenommenen Abwärme des DELIA-Moduls in Höhe von 30W möglich ist. Hierbei ergibt sich ein Massenstrom von 1 g/s und ein Systemdruck von ca. 1,19 bar. Die Simulationsergebnisse sind in **Fehler! Verweisquelle konnte nicht gefunden werden., Fehler! Verweisquelle konnte nicht gefunden werden. und Fehler! Verweisquelle konnte nicht gefunden werden.** dargestellt. Sichtbar ist jeweils ein numerischer Einschwingvorgang basierend auf den Erhaltungsgleichungen sowie der eingeschwungene Zustand des Systems. Durch die in **Fehler! Verweisquelle konnte nicht gefunden werden.** gezeigten Ergebnisse lässt sich der Systemdruck im eingeschwungenen Zustand ablesen, was die Basis für die Auslegung des Gesamtsystems bildet (z.B. Durchmesser des Fall- und Steigrohres).

Aufgrund der beschriebenen Schwierigkeiten und der vermutlich ausreichenden Luftkühlung der DELIA-Module besteht keine Verwertungsperspektive für eine Kühlung auf Basis einer Naturumlaufkühlung, zumindest für Module mit einer Wärmeleistung wie sie beim DELIA-Modul erwartet wird. Trotz dessen ist die Auslegung und Simulation eines solchen Systems hilfreich, um zukünftig die Eignung einer Naturumlaufkühlung für andere Systeme einfach bewerten zu können.

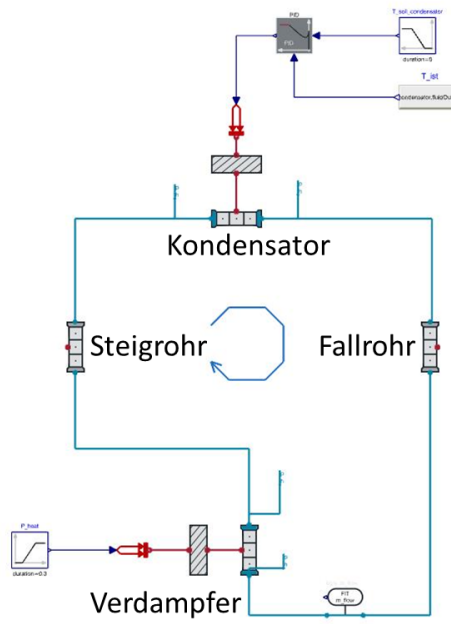


Figure 4: Umsetzung einer Naturumlaufkühlung in Dymola

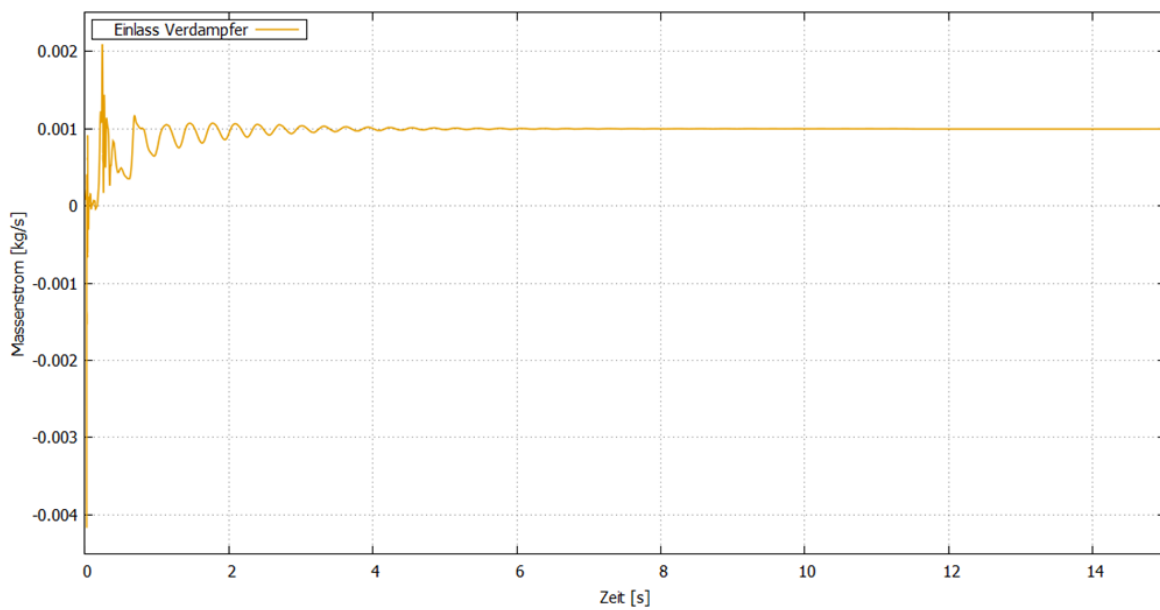


Figure 5: Massenstrom in der simulierten Naturumlaufkühlung

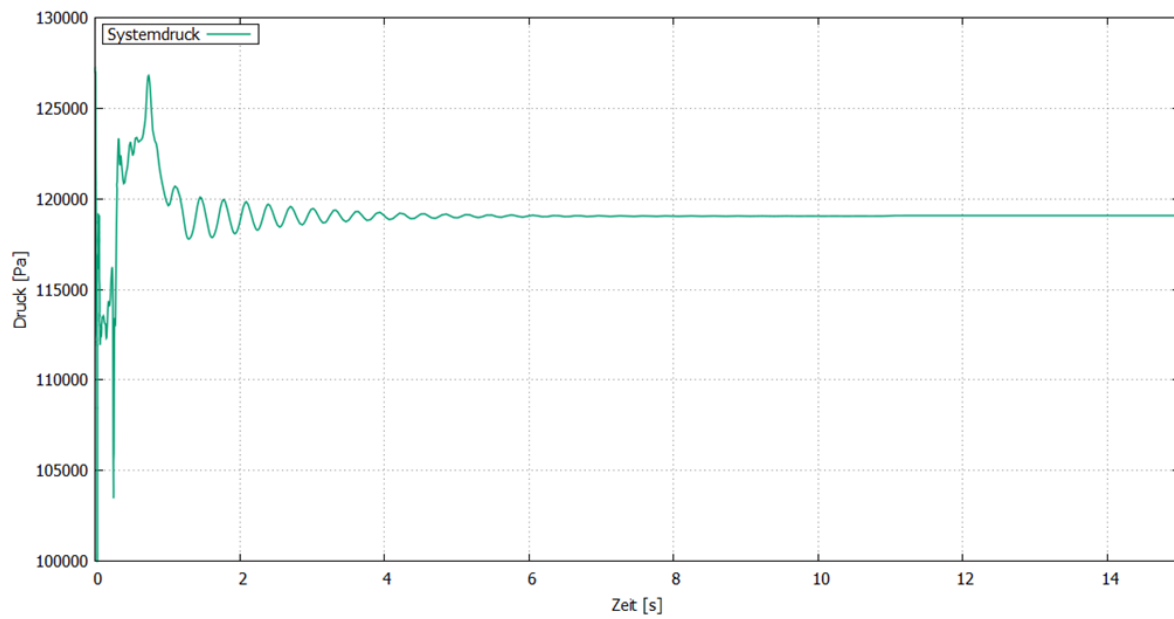


Figure 6: Druck in der simulierten Naturumlaufkühlung

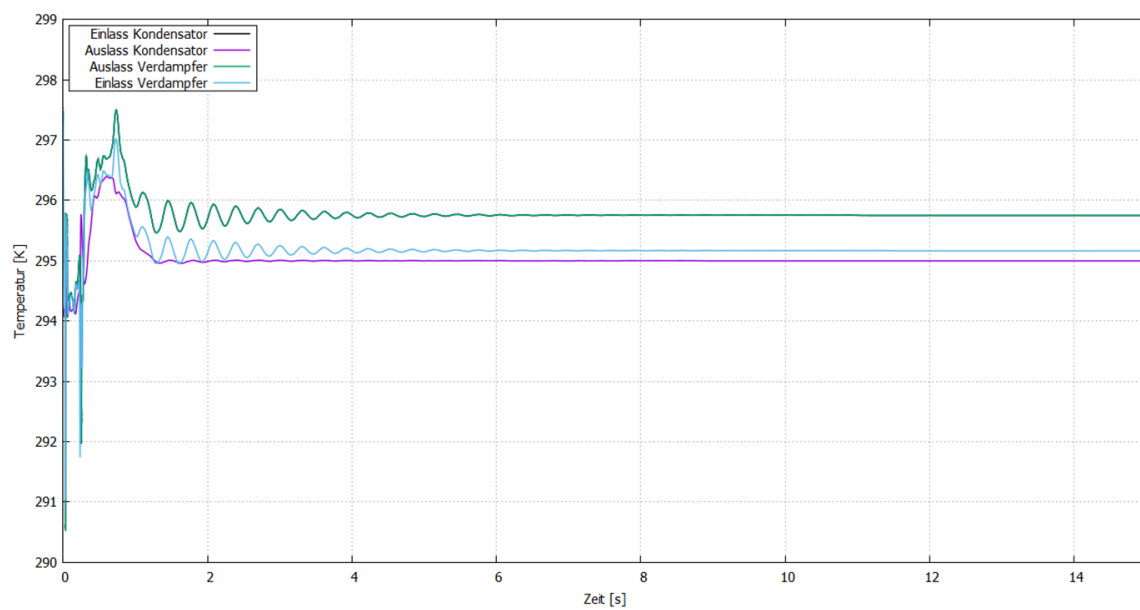


Figure 7: Temperatur in der simulierten Naturumlaufkühlung

7 ZAL Endpoint

In Arbeitspaket AP 2.1.ZAL.5 realisierte die ZAL GmbH eine Anbindung von Sensoren und Aktoren an das Kabinennetzwerk.

Das durch die Partner entwickelte DELIA-Modul sieht kaum Schnittstellen für Sensoren und Aktoren vor. Stattdessen ist die primäre externe Schnittstelle durch einen TSN-fähigen Netzwerkschalt gegeben, welcher sowohl 10 Gbit/s Anschlüsse als auch Single-Pair Ethernet Anschlüsse bis 1 Gbit/s bietet. Da der Einsatzzweck der DELIA-Module mit ihren Hochgeschwindigkeitsschnittstellen eher im Netzwerkbackbone zu sehen ist, wird eine Komponente notwendig, welche Sensoren und Aktoren an diesen Backbone anbinden kann.

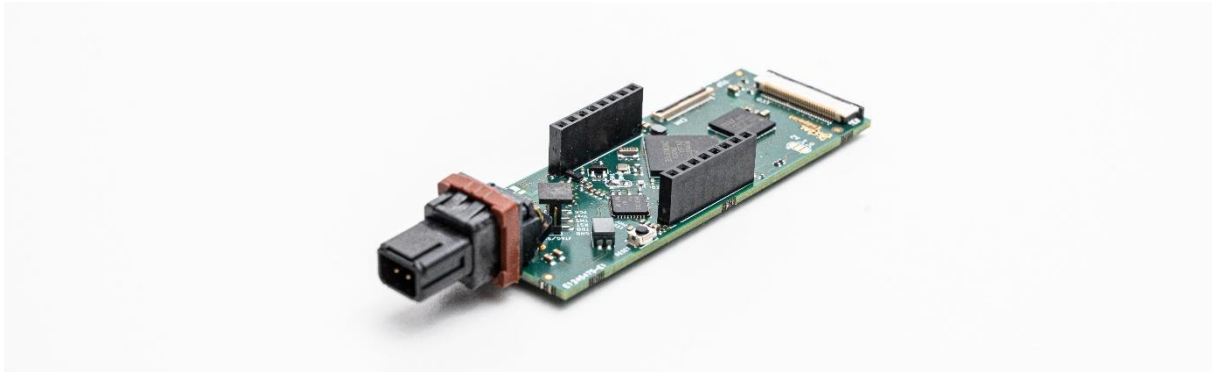


Figure 8: ZAL Endpoint

Aus diesem Grund hat sich die ZAL GmbH dazu entschlossen, den sog. ZAL Endpoint zu entwickeln (Figure 8: ZAL Endpoint). Der Endpoint ist eine auf einem Mikrocontroller basierende Elektronikkomponente, welche diverse Schnittstellen zur Anbindung von Sensoren und Aktoren bietet. Für die Versorgung kommt ein Stecker nach ARINC 854 zum Einsatz, der über je zwei Pins den Endpoint mit Energie und einer Datenverbindung über Single-Pair Ethernet versorgt. Weitere Schnittstellen sind z.B. I2C, I2S, SPI und ein Kamerainterface. Figure 9 zeigt die einzelnen Bauteile auf dem ZAL Endpoint.

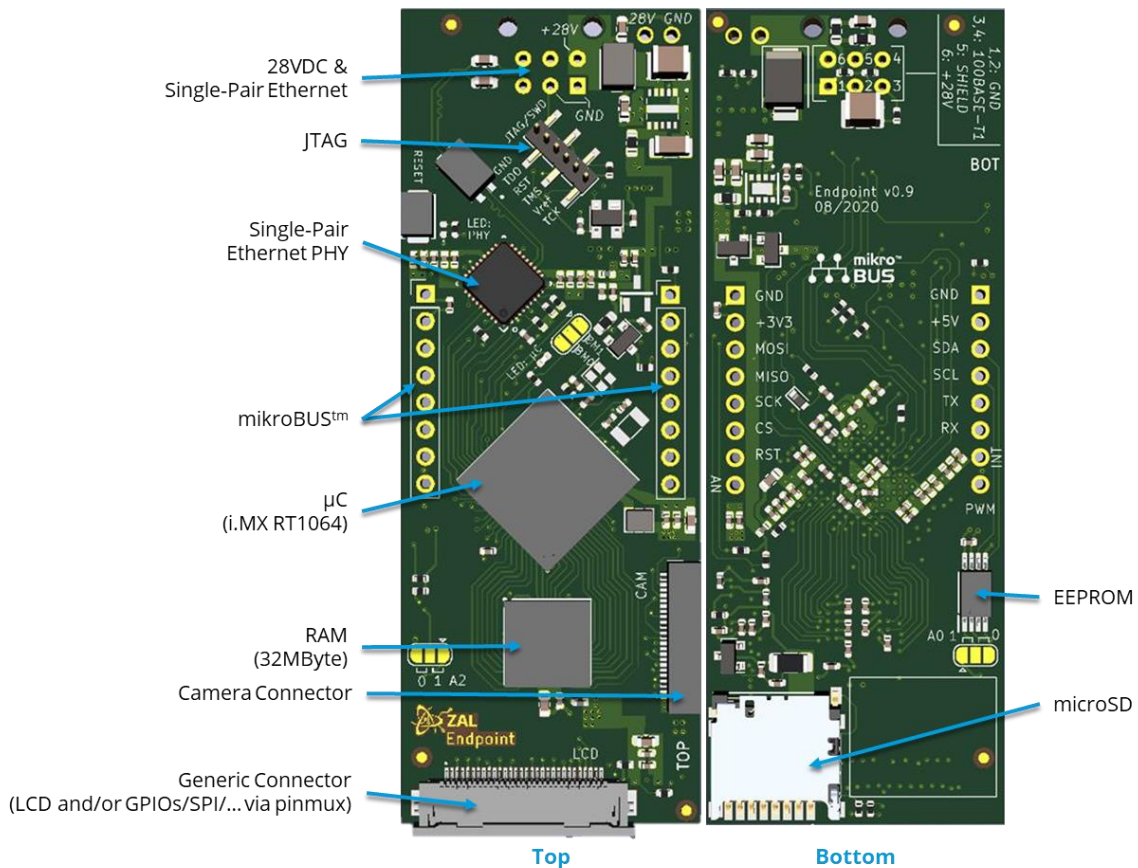


Figure 9: Bauteile des ZAL Endpoint

Als Betriebssystem kommt das Open-Source-Echtzeitbetriebssystem Zephyr OS zum Einsatz, welches entsprechend erweitert wurde, um Anwendungen für den ZAL Endpoint zu kompilieren.



Figure 10: ZAL Endpoint mit aufgesteckter Platine zur Anbindung von LED-Beleuchtung

Das Konzept des Endpoints sieht die Erweiterung für spezifische Anwendungsfälle über Aufsteckplatinen vor. So wurde im Berichtszeitraum zusätzlich eine Aufsteckplatine zur Anbindung von LED-Streifen zur Beleuchtung entwickelt (Figure 10). Diese beherbergt maßgeblich einen Spannungswandler mit ausreichend Leistung sowie Schnittstellen für weitere Sensorik und Überwachungsfunktionen. Über die Platine lassen sich handelsübliche LED-Streifen anbinden, bei denen die verbauten RGB-LEDs einzeln zur Darstellung von Lichteffekten angesteuert werden können. Figure 11 zeigt den Aufbau der Aufsteckplatine.

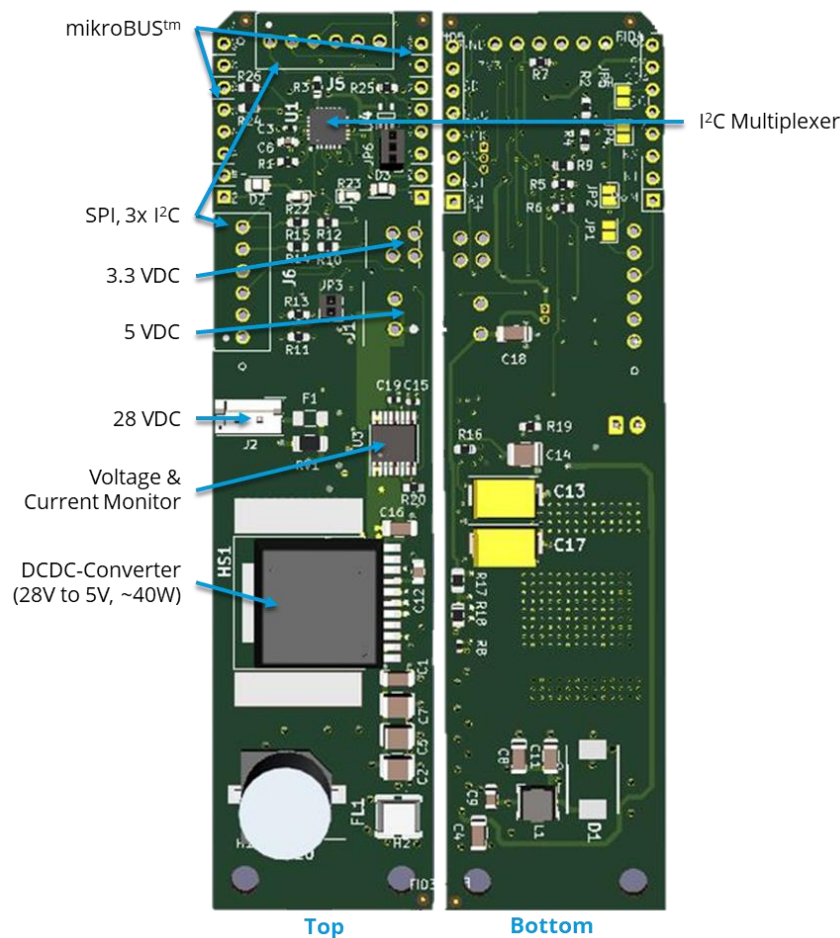


Figure 11: Bauteile der Aufsteckplatine für LED-Beleuchtung

8 Erweiterung der DELIA Softwareplattform

Ein essenzieller Bestandteil bei der Evaluierung der DELIA-Module ist die Erstellung einer einheitlichen, einfach zu benutzenden und funktionalen Softwareplattform. Hierfür hat die ZAL GmbH die Aufgabe übernommen, den Hypervisor XEN als Virtualisierungslösung auf den DELIA-Modulen umzusetzen. Auf Basis von XEN können auf den Modulen mehrere virtuelle Maschinen gestartet werden, welche entsprechend ihrer Anforderungen Ressourcen (z.B. Arbeitsspeicher, Anteil der CPU-Zeit) zugewiesen bekommen. Gleichzeitig kann sichergestellt werden, dass nicht-sicherheitskritische virtuelle Maschinen nicht in der Lage sind, sicherheitskritischen virtuellen Maschinen die zugewiesenen Ressourcen zu verwehren (z.B. in einer Überlastsituation).

Mit den Projektpartnern wurde sich auf das Betriebssystem Alpine Linux als Basis für die DELIA-Module geeinigt. Alpine Linux ist eine Distribution, welche sich insbesondere durch ihren geringen Speicherbedarf auszeichnet. Auch sind verfügbare Softwarepakete für diese Distribution zumeist von höherer Aktualität, als dass dies bei einigen anderen Distributionen (z.B. Debian) der Fall ist.

Die DELIA-Module nutzen von Haus aus „U-Boot“ als Bootloader, welcher speziell für eingebettete Systeme gedacht ist. Die ZAL GmbH hat ein entsprechendes Skript erstellt, welches durch U-Boot beim Booten des Systems interpretiert wird und sowohl XEN als auch den Linux-Kernel der ersten, privilegierten virtuellen Maschine (dom0) in den Arbeitsspeicher lädt. Zusätzlich wird der Device-Tree für das System geladen und um Informationen zum Booten von XEN und der dom0 ergänzt. Anschließend wird XEN gestartet. Des Weiteren wurde Alpine Linux um die für XEN relevanten Dienste und Einstellungen ergänzt.

Die ZAL GmbH hat ein Skript erstellt, welches ein Alpine Linux Dateisystem für das DELIA-Modul auf Basis des aktuellen Versionsstandes von Alpine Linux erstellen kann und dabei die projektspezifischen Anpassungen automatisiert vornimmt. Außerdem wurde ein Skript des Projektpartners Solecatrix zum Erstellen eines SD-Karten-Images angepasst, um das erstellte Skript für U-Boot sowie XEN einzubinden. Durch diese Skripte haben auch die Projektpartner die Möglichkeit, ein angepasstes und einheitliches Softwaresystem für die DELIA-Module zu generieren und aktuell zu halten.

9 Evaluierung von Kommunikationsprotokollen

In Arbeitspaket 2.1.ZAL.6 wurde die Datenweitergabe umgesetzt, sodass Sensoren und Aktoren über das Netzwerk ausgelesen bzw. angesprochen werden können. Hierbei hat die ZAL GmbH drei Übertragungsprotokolle evaluiert: MQTT, OPC UA PubSub und Zenoh.

MQTT

Das MQTT-Protokoll hat sich zunehmend zu einer starken Alternative zu Client-Server-Architekturen etabliert und findet besonders im Bereich des IoT Einsatz. Hier können IoT-Devices mit einer Backend-Plattform verbunden werden und Nachrichten miteinander austauschen. Das Protokoll folgt dabei den Regeln einer Publish/Subscribe-Kommunikation, bei welcher zwei verschiedene Teilnehmer, ein Broker und mehrere Clients, das Netzwerk spannen. Die Clients identifizieren sich durch eine Client-ID und veröffentlichen (publishen) ihre Nachrichten auf sogenannten „Topics“. Die Schreibsyntax der Topics ähnelt stark der Ordner-Syntax eines Filesystems, z.B. „cabin/side/led“. Jede MQTT-Nachricht enthält außerdem eine Payload und ist einem Topic zugeordnet. Eine Nachricht kann nur von den Clients empfangen werden, die das entsprechende Topic abonnieren (subscribe). Der Broker übernimmt hierbei die Nachrichtenverwaltung und leitet alle Nachrichten mit dem Topic an die Subscriber weiter.

Das MQTT-Protokoll sorgt demnach für drei Effekte:

- Räumliche Entkopplung: Die Subscriber und Publisher haben keinen direkten Kontakt zueinander (via Broker)
- Zeitliche Entkopplung: Die Subscriber und Publisher dürfen zeitlich versetzt laufen
- Synchronisationsentkopplung: Die Nachrichtenübermittlung ist nicht blockierend (asynchrones Messaging)

Vorteile von MQTT Pub/Sub sind seine gute Skalierbarkeit, im Vergleich zu klassischen Client-Server-Architekturen und Priorisierbarkeit, da Nachrichten außerdem mit „Quality of Service“ (QoS) Werten versehen werden können, wodurch sich Übertragungsgarantien konfigurieren lassen. Der gezielte Datenverkehr im schlanken und simplen Aufbau machen das MQTT-Protokoll demnach interessant für den Einsatz im DELIA-Aufbau.

OPC UA PubSub

OPC UA ist ein Protokoll, das aus dem Industrieumfeld, z.B. der Steuerung von Maschinen, bekannt ist. Mit der Erweiterung OPC UA PubSub wurde 2018 ein Protokoll veröffentlicht, welches wie MQTT einem Publish/Subscribe-Schema folgt. Hierbei liegt der Vorteil gegenüber MQTT aber insbesondere in dem Verzicht auf einen zentralen Broker.

Im Verlauf der Bearbeitung des Arbeitspaketes wurde ein OPC UA-Datenmodell für die Steuerung einer LED-basierten Kabinenbeleuchtung entwickelt. Des Weiteren wurde auf Basis von Open-Source-Software eine OPC UA PubSub-Implementierung auf dem ZAL Endpoint sowie auf dem DELIA-Modul umgesetzt und erprobt. Für beide Gerätetypen kamen dabei unterschiedliche Bibliotheken zum Einsatz, sodass auch die Interoperabilität verschiedener OPC UA PubSub-Umsetzungen überprüft werden musste. Zumindest in einem vereinfachten Modus ohne

Verschlüsselung konnte das Zusammenspiel der unterschiedlichen Umsetzungen bereits gezeigt werden.

Nach der Evaluierung von MQTT und OPC UA PubSub wurde zusätzlich nach Alternativen gesucht. Der Grund hierfür war die teils noch nicht vollständige Umsetzung von OPC UA PubSub in Open-Source Softwarebibliotheken, das Fehlen von Dokumentation zu diesen und die nötige steile Lernkurve zum Einsatz dieser. Eine vielversprechende Alternative schien das Protokoll zenoh² zu sein.

Zenoh

Ebenso wie OPC UA PubSub kommt dieses Protokoll ohne zentrale Instanz (wie z.B. einem Broker) aus. Verschiedene Geräteklassen nehmen dabei Rücksicht auf die Einschränkungen (z.B. hinsichtlich Ressourcen) und Anforderungen bestimmter Kommunikationsteilnehmer. Sog. Peers kommunizieren in einer vollvermaschten Architektur (jeder mit jedem) und sog. Clients kommunizieren nur jeweils mit einem Zenoh-Router, welcher wiederum die Verbindung zu anderen Routern bzw. lokalen Zusammenschlüssen von Peers herstellt. Ähnlich wie MQTT, beruht Zenoh auf einer Abstraktion mittels Key/Value, wobei Keys z.B. folgende Syntax haben können „/demo/example/test“. Die Values können hierbei z.B. StringUTF8, Integer oder JSON als Encoding haben.

Die ZAL GmbH hat Zenoh evaluiert, indem es Tests auf Basis der C-API durchgeführt hat. Dazu wurde das Open-Source-Projekt aus dem offiziellen Git-Repository ausgecheckt, gebaut und anschließend als statische Bibliothek in ein Test-Projekt integriert. Dieses diente dazu, die Zenoh-Testprogramme in einen Docker-Container zu hüllen und mittels des ZAL Docker-Repositorys verfügbar zu machen. Hierbei wurde sowohl die Zenoh-Bibliothek als auch der Docker-Container auf ARM64 kompiliert und kann mittels Qemu-Container auch auf einer AMD64-Architektur gestartet werden.

Im ersten Schritt wurde Zenoh auf einem ARM64-basierten Multi-Core-System mit einem Publisher und einem Subscriber und anschließend mit zwei Publishern und zwei Subscribern getestet. Die Tests zeigten, dass sich die Komponenten wie erwartet verhielten und nach dem Starten der Container zeitnah ein Datenaustausch zwischen den Einheiten stattfand. Wie erwünscht, führte das Abschalten einer Einheit zur Unterbrechung und das erneute Hochfahren wieder zur Aufnahme des Datenaustausches.

Im zweiten Schritt wurde Zenoh im LAN getestet, wobei zusätzlich zum ARM64 Multi-Core-Rechner noch ein X86-basierter Server verwendet wurde. Hier wurden die gleichen Testszenarien wie oben genannt durchgeführt. Es wurde ein zeitnaher Datenaustausch zwischen den Einheiten nach dem Starten der Container beobachtet. Die Einheiten benötigten keine zusätzlichen Konfigurationen, wie gegenseitiges Bekanntmachen der IP-Adressen, da sie sich auch im LAN finden konnten.

Die Evaluation von Zenoh hat ergeben, dass Messdaten und Steuerbefehle gut über das Netzwerk ausgetauscht werden können. Jedoch sind für das Übertragen von Streams, wie z.B. dem Audio-Stream einer Durchsage, andere Technologien bzw. Protokolle besser geeignet.

² <https://zenoh.io/>

10 Aufbau des DELIA Demonstrators

Hauptaufgabe des Arbeitspakets 3.1.ZAL und 3.2.ZAL war die Systemintegration der DELIA-Komponenten und der Aufbau eines Demonstrators. Nach Erhalt der DELIA Carrier- und Processing-Boards des Projektpartners Solectrix, konnte die ZAL GmbH erste Systemtests als Tischaufbau vornehmen und mehrere Skripte zum Kompilieren von Images erstellen. In Zusammenarbeit mit der Universität Stuttgart konnte das DELIA-Betriebssystem und dessen Konfiguration durch Patches und verschiedene Versionen des Kernels verbessert werden. Dies wurde dann durch den Erhalt der IO-Boards vom Projektpartner AED im Laufe des Projektes ergänzt.

10.1 Erster Tischaufbau der DELIA-Komponenten

Mittels eines Tischaufbaus wurde zunächst ein Netzwerk zwischen einem ZAL-Server, zwei DELIA-Modulen und einem Laptop mithilfe von Ethernet und Glasfaser aufgebaut und Latenzen gemessen (siehe Figure 12: Erste Glasfaser Tests an den DELIA-Modulen).

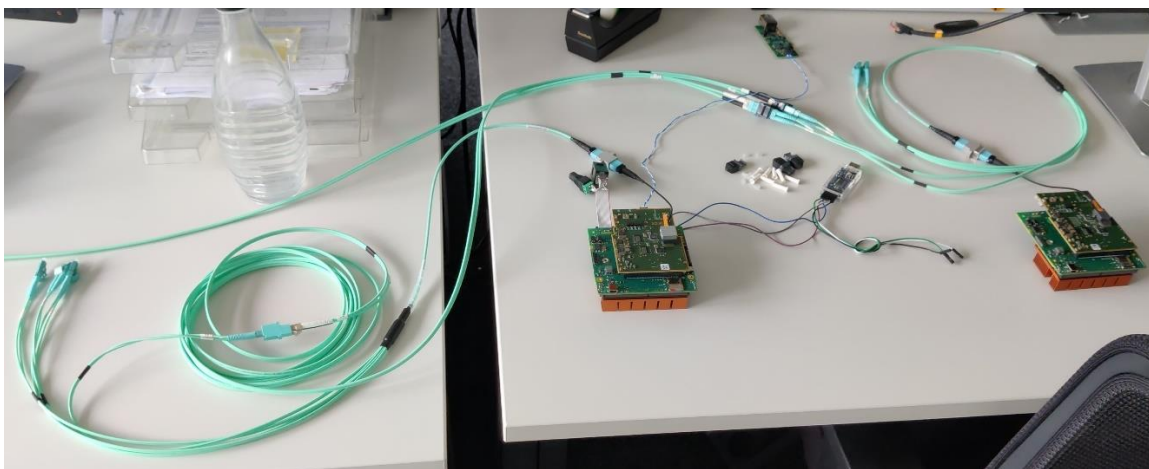


Figure 12: Erste Glasfaser Tests an den DELIA-Modulen

Bei dem Aufbau in Figure 12 wird der Basisswitch getestet: Hierbei sendet ein Server Daten über das ZAL-Labornetz per Ethernet an einen EVB-Converter, welcher diese zu Single-Pair-Ethernet wandelt. Anschließend erreichen die über Single-Pair-Ethernet gesendeten Pakete dann das DELIA-Modul und werden über das 10Gbit Glasfaserinterface weitergeleitet. Über den Samtec PCIe FireFly Glasfaseradapter werden die Daten zunächst an ein Fan-Out und anschließend an das zweite DELIA-Modul geschickt. Per Ethernet wird das zweite DELIA-Modul verbunden, um die Datenpakete entgegenzunehmen. Die Bandbreite wurde mit dem Open-Source-Netzwerk-Performance-Tool *Iperf* gemessen, bei welchem UDP und TCP Datenstreams zwischen einer Client- und Server-Seite ausgetauscht werden können. Durchsatz und erfolgreich übertragene Daten können hiermit in einem Bericht mit Zeitstempel festgehalten werden. Die Messungen mit *Iperf* zeigten, dass von den möglichen 10 Gbit/s bei 10 TCP Streams 3Gbit/s erreicht werden konnte. Die Abweichung zwischen der möglichen und realen Übertragungsrate lässt sich zum einen darauf zurückführen, dass das verwendete Netz über mehrere Etagen vom Kabinendemonstrator bis zum Tischaufbau gespannt wurde. Weiterer Faktor könnten die werksinternen Restriktionen der Fibre-Channel-PCI-Netzwerkkarten des verwendeten ZAL-Servers sein.

Auf Basis dieses ersten Testaufbaus, konnte zum einen besser geplant werden, welche fehlende Hardware noch für den Aufbau im DELIA-Demonstrator zu beschaffen ist. Zum anderen verdeutlichte sich hierbei auch, dass ohne einen funktionierenden CPU-Treiber im IO-Board, welcher vom Projektpartner AED erstellt werden sollte, keine Ringtopologie und auch keine andere Topologie umsetzbar ist. Ein CPU-Treiber ist ein fundamentaler Bestandteil jedes Systems, und

bildet die Brücke zwischen dem Prozessor und den auf dem Modul verbauten Schnittstellen. Da dieser nicht rechtzeitig von AED zur Verfügung gestellt werden konnte, hat die ZAL GmbH in Rücksprache mit den Projektpartnern den DELIA-Demonstrator-Aufbau so gestaltet, dass der geplante Use Case auch bei Nichtverwendung der IO-Boards und der DELIA-Module umgesetzt werden konnte.

Die monatlichen DELIA-Meetings mit den Projektpartnern ermöglichten es, dass zum letzten Viertel des Projektes, gemeinsam ein möglicher Use-Case diskutiert und festgelegt wurde den die ZAL GmbH dann im TechLab aufgebaut hat. Offene Diskussionspunkte konnten durch separate Meetings mit den jeweiligen Projektpartnern gelöst und den restlichen Parteien kommuniziert werden. Die ZAL GmbH hat hierbei federführend eine DELIA-Demonstrator-Architektur aufgestellt, welche das Lichtszenario mit erhöhtem Traffic in der Kabine und TSN-Fähigkeiten umfasst.

10.2 Konzeptionierung und Auswahl von Use-Cases

Nachdem in Arbeitspaket 2.1.ZAL.7 zunächst mögliche Use-Cases diskutiert und dokumentiert wurden, erfolgte nun die Auswahl, die Implementierung und das Testen eines passenden Anwendungsfalls für die DELIA-Ergebnisse. Da der ZAL Endpoint als Universal-Plattform im Rahmen dieses Forschungsprojektes entstanden ist, und mit seiner Architektur als nützliche Komponente in der Kabine überzeugt, war bereits früh klar, dass dieser Bestandteil des Demonstratoraufbaus sein würde.

Bei der Auswahl eines Use-Cases galten folgende Auswahlkriterien:

- Der Use-Case sollte einen alltäglichen Betriebszustand in der Kabine abbilden
- Die involvierten Komponenten sollten erhöhtem Netzwerkverkehr ausgesetzt sein
- Der Use-Case sollte auch den Ausfall einer Komponente umfassen (Emergency)

Die Wahl des Demonstrator-Use-Cases fiel aus diesem Grund auf ein Kabinen-Licht-Szenario. Ausgangszustand des Use-Cases ist zunächst, dass nachdem die Software auf allen Komponenten gebootet hat, ein Lauflicht in der Kabine im ZAL GmbH TechLab sichtbar ist. Hoher Netzwerkverkehr ist hierbei erfüllt, da der Befehl für das Ausführen des Lauflichtes nicht einmalig erfolgt (z.B. run chaser light), sondern sich auf Netzwerkebene auf viele einzelne Befehle herunterbricht (LED1 aus, LED2 an, LED3 aus, etc.). Somit wird das Netz mit vielen Datenpaketen geflutet und künstlich ausgelastet. Die TSN-Fähigkeiten und somit auch die verbundene Pfadredundanz beim Ausfall einer Komponente soll dadurch simuliert werden, dass eines der DELIA-Komponenten plötzlich vom Netzwerk getrennt wird. Dies deckt den Ausfall einer Systemkomponente im Falle eines Stromausfalles in der Kabine, den Ausfall aufgrund eines Angriffs auf das Netzwerk (von innerhalb und außerhalb der Kabine) und den Ausfall aufgrund von Überlastung einer Komponente durch den zu hohen Netzwerkverkehr ab. Das gewünschte Verhalten hierbei soll sein, dass obwohl ein Teilnehmer des Netzwerkes ausfällt, das Lauflicht von den verbliebenen Modulen weiterhin aufrechterhalten werden kann und keine Störungen (Sprünge, Einfrieren oder Ausbleiben der LED-Darstellung) aufweist.

10.3 Entwicklung des DELIA Demonstrators

Um den Anwendungsfall eines Licht-Szenarios in der Kabine mit den oben beschriebenen Anforderungen umsetzen zu können, galt es nun die nötigen Systembestandteile für den DELIA-Demonstrator zu identifizieren. Die ZAL GmbH hat dabei die folgenden Teilnehmer des DELIA-Demonstrator Netzwerkes identifiziert:

1. Raspberry Pi 4: Von diesem werden die Befehle zur Lichtsteuerung auf den LED-Streifen in der Kabine gesendet.

2. Vier Microchip-Switches: Die beschriebenen TSN-Standards lassen sich in diesen Switchen durch Konfigurationsdateien umsetzen und auf das DELIA-Netzwerk im Demonstrator anwenden
3. Vier LED-Streifen: Das Lauflicht (und auch andere Farbszenarios) werden durch LED-Streifen, die in der Kabine im ZAL TechLab links und rechts über den Fenstern unter dem Hatrack verbaut werden, angezeigt. Es handelt sich hierbei um RGB-LEDs mit 60 LEDs/Streifen.
4. Vier ZAL Endpoints: Diese übernehmen die Aufgabe der Ansteuerung der LED Streifen
5. Vier EVB-LAN8770M_MC Converter: Diese wandeln „normales“ 100BASE-TX Ethernet, welches von den TSN-Switchen ein- und abgeht, in 100BASE-T1 Single-Pair-Ethernet (SPE), das die Endpoints entgegennehmen.



Figure 13: Farbverlauf im finalen Demonstrator Aufbau

Die Systemarchitektur in der umgesetzten Ringtopologie wird in Abbildung Figure 14: DELIA Systemarchitektur verdeutlicht.

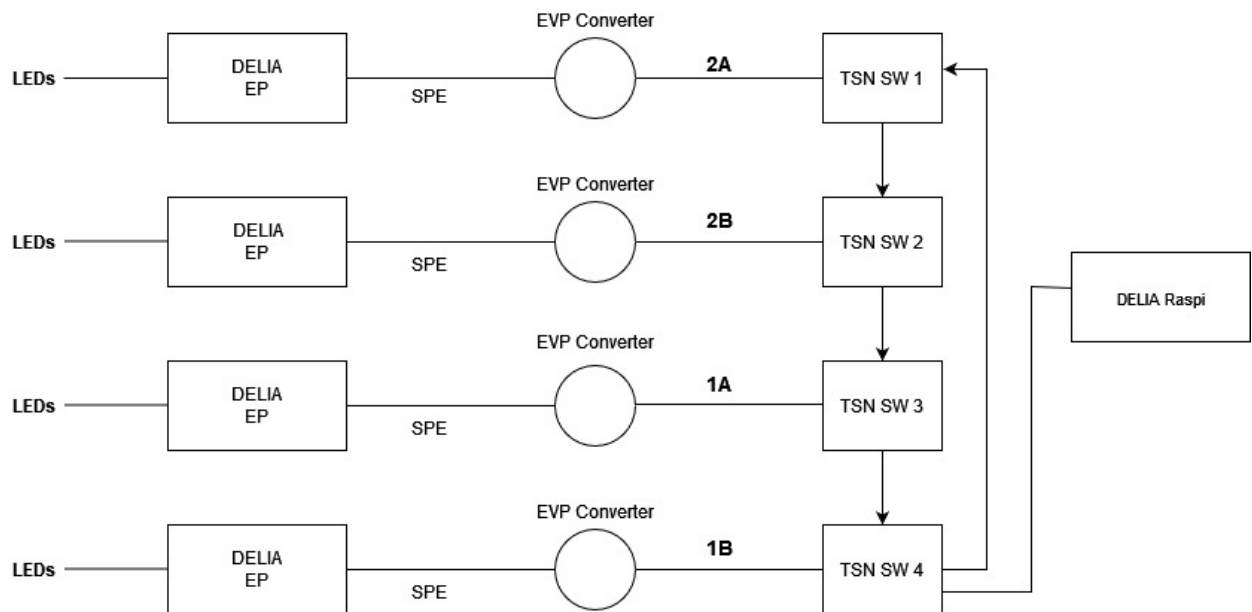


Figure 14: DELIA Systemarchitektur

10.4 Softwarearchitektur

Die Befehle zur Steuerung der verbauten LED-Streifen in der Kabine im Demonstrator werden vom Raspberry Pi 4 erstellt. Um dies zu erreichen, bilden die folgenden drei Bauteile das Herzstück der Software im Demonstrator:

- Controller mit MQTT upstream
- Translator mit MQTT downstream
- Fileserver (HTTP Webserver Kommunikation)

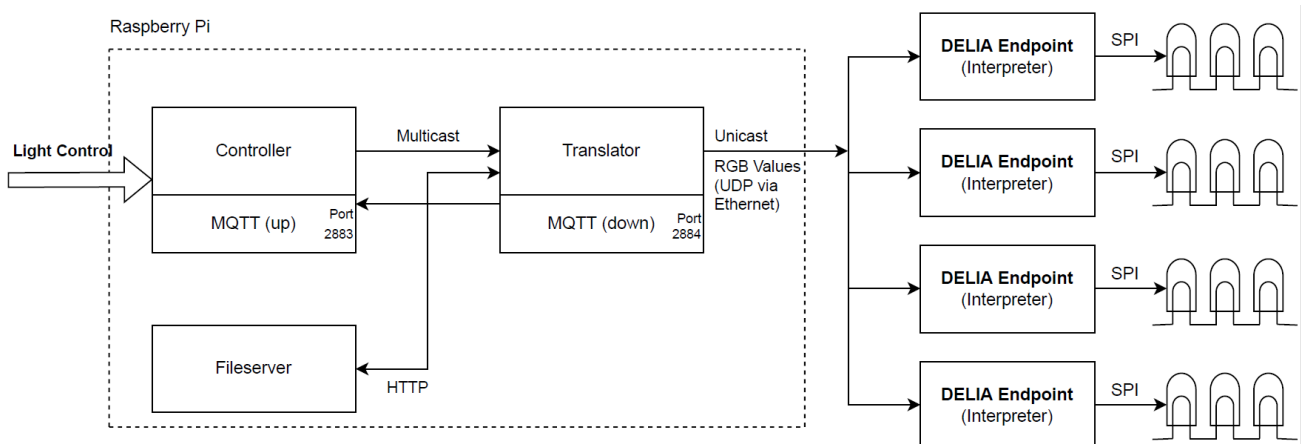


Figure 15: DELIA Demonstrator Software Architektur

Ein User kann die Befehle zum Wechsel der Lichteffekte der LED-Streifen über ein MQTT-Interface (z.B. MQTT Explorer) an den Controller senden. Dafür muss sich das MQTT-Interface im gleichen Netz wie der Raspberry Pi befinden. Weiterhin wird die IP-Adresse des Raspberry Pi und der MQTT-Port benötigt, um auf dem entsprechenden topic den Befehl für die Lichteffekte zu publishen. Hierfür gibt es festgelegte Befehle, wie z.B. „chaser light“ oder einen RGB-Farbwert für die LEDs.

Der Controller übergibt die Parameter anschließend zur Ausführung an den Translator über Multicast. Der Translator fragt per HTTP-Request einen Fileserver an, welche Befehle zu welchen Daten und Effekten gehören, um diese dann an die DELIA-Endpoints per UDP weitergeben zu können. Alle drei Instanzen laufen als einzelne Docker Container auf dem Raspberry Pi, um Daten- und Rechteverwaltung sicherzustellen (siehe Figure 16: Translator Docker Container Ausgabe).

```

nglighttranslator1 [2022-08-31 06:15:33.541] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:33.541] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... topic: "cabin/devices/illumination/132:12/heartbeat"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... payload: "71"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:33.599] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... topic: "cabin/devices/illumination/132:13/heartbeat"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... payload: "71"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:33.599] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... topic: "cabin/devices/illumination/132:11/heartbeat"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... payload: "71"
nglighttranslator1 [2022-08-31 06:15:33.599] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:33.599] [info] Scenario Control from 10.10.254.250:39002, Step 63043 ...
nglighttranslator1 [2022-08-31 06:15:34.533] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... topic: "cabin/devices/illumination/132:10/heartbeat"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... payload: "72"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:34.607] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... topic: "cabin/devices/illumination/132:12/heartbeat"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... payload: "72"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:34.607] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... topic: "cabin/devices/illumination/132:13/heartbeat"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... payload: "72"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:34.607] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... topic: "cabin/devices/illumination/132:11/heartbeat"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... payload: "72"
nglighttranslator1 [2022-08-31 06:15:34.607] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:34.607] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:35.004] [info] ... topic: "cabin/devices/illumination/132:10/heartbeat"
nglighttranslator1 [2022-08-31 06:15:35.004] [info] ... payload: "73"
nglighttranslator1 [2022-08-31 06:15:35.004] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:35.004] [info] Scenario Control from 10.10.254.250:39002, Step 63044 ...
nglighttranslator1 [2022-08-31 06:15:35.533] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... topic: "cabin/devices/illumination/132:10/heartbeat"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... payload: "73"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:35.676] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... topic: "cabin/devices/illumination/132:12/heartbeat"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... payload: "73"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:35.676] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... topic: "cabin/devices/illumination/132:13/heartbeat"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... payload: "73"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:35.676] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... topic: "cabin/devices/illumination/132:11/heartbeat"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... payload: "73"
nglighttranslator1 [2022-08-31 06:15:35.676] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:35.676] [info] Scenario Control from 10.10.254.250:39002, Step 63045 ...
nglighttranslator1 [2022-08-31 06:15:36.080] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... topic: "cabin/devices/illumination/132:10/heartbeat"
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... payload: "74"
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:36.080] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... topic: "cabin/devices/illumination/132:12/heartbeat"
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... payload: "74"
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... forwarding to upstream
nglighttranslator1 [2022-08-31 06:15:36.080] [info] Downstream Message arrived ...
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... topic: "cabin/devices/illumination/132:13/heartbeat"
nglighttranslator1 [2022-08-31 06:15:36.080] [info] ... payload: "74"

```

Figure 16: Translator Docker Container Ausgabe

Die Endpoints senden außerdem periodisch einen heartbeat an den MQTT-Broker, womit detektiert werden kann, ob und welche Teile des Netzwerks offline sind.

10.5 Datenweitergabe über das Netzwerk

Die Datenübertragung ist im DELIA-Demonstrator über mehrere Protokolle sichergestellt. Der User interagiert mit dem System über die Eingabe von MQTT-Befehlen beim Raspberry Pi (oder über eine SSH-Verbindung zum Pi). Dieser kommuniziert dann ggf. mit dem Fileserver und leitet die Befehle per Ethernet-Pakete an die TSN-Switche weiter. An diesen werden die TSN-Standards angewendet und die Pakete dann an die EVB-Konverter und ZAL Endpoints per SPE weitergeleitet.

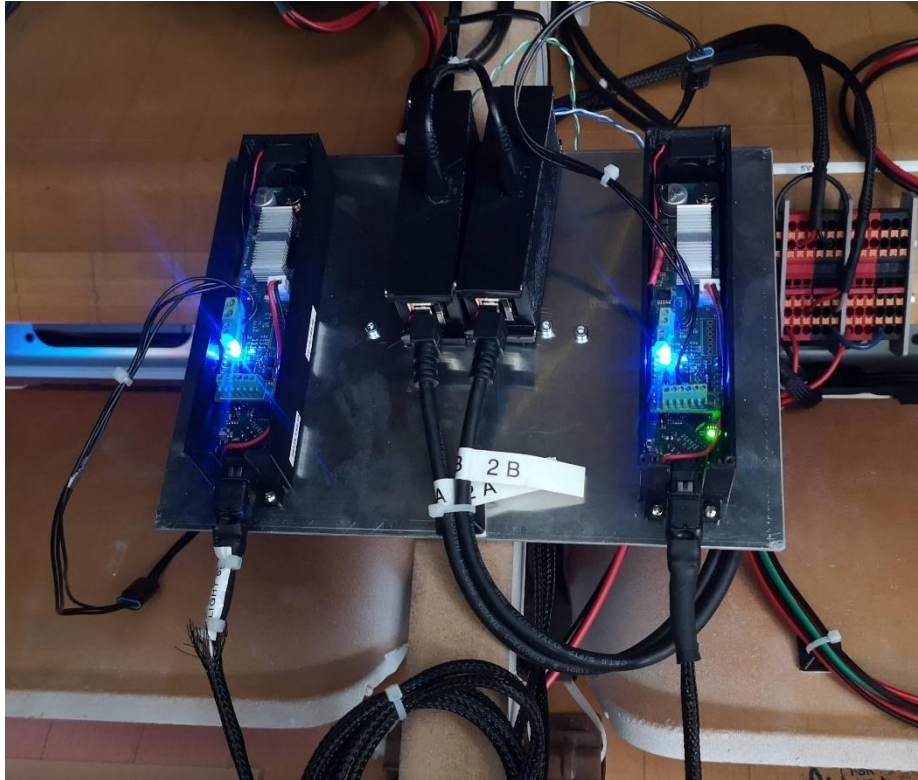


Figure 17: Endpoints mit Light add-on und EVB Converter an der Außenseite der Kabine

Im Folgenden wird genauer auf die implementierten TSN-Konfigurationen in den TSN-Switches eingegangen.

10.6 Time-Sensitive Networking

Für die Kommunikation zwischen den einzelnen Modulen und Endpunkten ist Time-Sensitive Networking (TSN) ein integraler Bestandteil der Architektur des DELIA-Demonstratoraufbaus der ZAL GmbH. TSN umfasst eine Reihe von Standards für die Übertragung von Daten über Ethernet-Netzwerke. Hierbei waren folgende Anforderungen die Grundlage der Evaluation im Rahmen von DELIA:

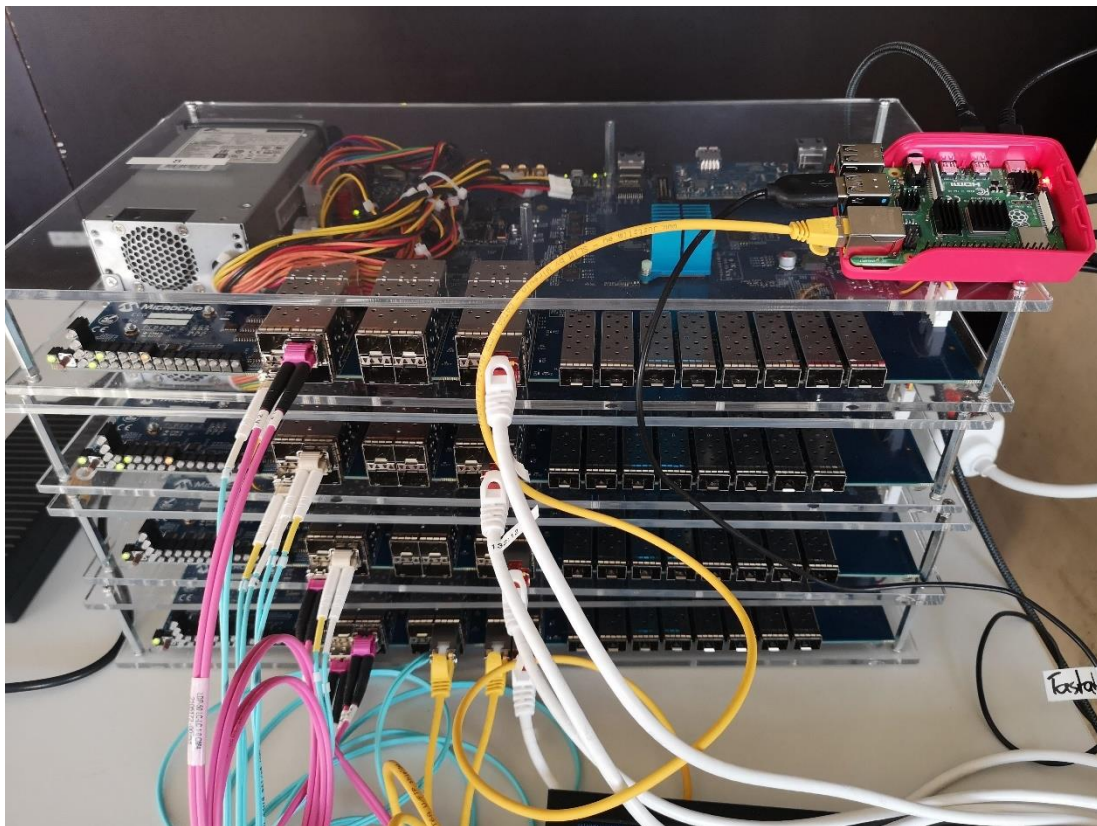
- Latenzgarantien für bestimmte Funktionalitäten
- Ankunftsgarantien für bestimmte Funktionalitäten
- Zeitsynchronisation im einstelligen Mikrosekunden-Bereich für bestimmte Funktionalitäten

Zur Einhaltung dieser Anforderungen war ursprünglich der Switch angedacht, welcher in den DELIA-Modulen vom Projektpartner AED integriert werden sollte. Aus unterschiedlichen Gründen konnte AED allerdings bis zum Projektende keine ausgereifte Version eines TSN-fähigen Switches auf dem IO-Board zur Verfügung stellen, welche den Anforderungen des Demonstrators genügte. Des Weiteren konnte der fehlende CPU-Treiber für das Betreiben der DELIA-Module nur knapp vor Projektende von AED geliefert werden, was der Grund dafür war, dass die ZAL GmbH sich dazu entschlossen hat, stattdessen den DELIA-Demonstrator mit kommerziellen TSN-Switches von Microchip aufzubauen (siehe Switch-Aufbau in Figure 18: TSN Switches mit Ethernet und Glasfaser Anschlüssen – Rechts oben der Raspberry Pi

Folgende TSN-Substandards wurden von der ZAL GmbH als relevant für die Netzwerkarchitektur in DELIA identifiziert: IEEE 802.1Qbv, IEEE 802.1AS, IEEE802.1CB und IEEE 802.1Qci. Diese werden im Folgenden kurz diskutiert:

IEEE 802.1Qbv ermöglicht es, im Zusammenspiel mit anderen Techniken wie Cut-Through-Switching, ein Datenpaket in der für das Netzwerk minimalst möglichen Zeit zu übertragen, was für Echtzeitkommunikation erforderlich ist. Hierfür wird allerdings eine Zeitsynchronisation der Switches benötigt, was einen weiteren TSN-Standard, IEEE 802.1AS, notwendig macht. Auch Endgeräte können hierdurch synchronisiert werden, was die vorangehend genannte Anforderung erfüllt.

Darüber hinaus ist es für Ankunftsgarantien erforderlich, dass es redundante Pfade im Netzwerk gibt, falls bestimmte Teile ausfallen sollten. Diese Pfade zu nutzen, erfordert neue Netzwerktechniken. In der Vergangenheit gab es hierzu bereits Protokolle wie PRP oder HSR, die allerdings vergleichsweise starr sind und wenig Flexibilität im Netzwerkdesign zulassen. TSN bietet



daher den IEEE 802.1CB Standard, der individuell erstellbare Pfade und eine selektive Redundanz nur für die Funktionalitäten ermöglicht, die sie benötigen. Aus diesem Grund wird dieser Standard ebenfalls als relevant für DELIA angesehen.

Figure 18: TSN Switch mit Ethernet und Glasfaser Anschlüssen – Rechts oben der Raspberry Pi

Zuletzt gibt es noch einen weiteren Standard, der sich nicht direkt in einer der drei Anforderungen widerspiegelt, aber ohne den keine davon garantiert werden könnte: IEEE 802.1Qci. Dieser Standard setzt Verkehrsregeln im Netzwerk durch, sodass jede Funktionalität nur die Daten verschickt, die vorgesehen sind. Andernfalls können falsch konfigurierte Geräte oder sogar Angreifer das Netzwerk lahmlegen, indem sie große Mengen an Datenverkehr erzeugen und damit anderen Datenverkehr behindern.

Angewandt auf den DELIA-Demonstratoraufbau ist jedoch hauptsächlich der IEEE 802.1CB mit FRER relevant. Die ZAL GmbH hat für die Integration der TSN-Switche in den Demonstrator, vier Konfigurationsdateien erstellt (siehe Figure 19: Ausschnitt TSN-Switch-Konfiguration).

```

11 mac address-table static AA:F0:55:86:D3:E3 vlan 52 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
12 mac address-table static AA:F0:23:EC:68:89 vlan 53 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
13 mac address-table static AA:F0:44:B2:F9:9D vlan 54 interface 10GigabitEthernet 1/12
14 mac address-table static ff:ff:ff:ff:ff:ff vlan 40 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/12
15 mac address-table static e4:5f:01:75:24:f1 vlan 41 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
16 mac address-table static ff:ff:ff:ff:ff:ff vlan 41 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
17 mac address-table static e4:5f:01:75:24:f1 vlan 42 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
18 mac address-table static ff:ff:ff:ff:ff:ff vlan 42 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
19 mac address-table static e4:5f:01:75:24:f1 vlan 43 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
20 mac address-table static ff:ff:ff:ff:ff:ff vlan 43 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
21 mac address-table static e4:5f:01:75:24:f1 vlan 44 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
22 mac address-table static ff:ff:ff:ff:ff:ff vlan 44 interface 10GigabitEthernet 1/1 10GigabitEthernet 1/4
23 spanning-tree mst name 02-00-cl-95-21-98 revision 0
24 green-ethernet fan temp-on 40 temp-max 60
25 !
26 stream 54
27   outer-tag vid 54 / 4095
28 stream 44
29
30 !
31 !
32 voice vlan oui 00-01-E3 description Siemens AG phones
33 voice vlan oui 00-03-6B description Cisco phones
34 voice vlan oui 00-0F-E2 description H3C phones
35 voice vlan oui 00-60-B9 description Philips and NEC AG phones
36 voice vlan oui 00-D0-1E description Pingtel phones
37 voice vlan oui 00-E0-75 description Polycom phones
38 voice vlan oui 00-E0-BB description 3Com phones
39 !
40 interface GigabitEthernet 1/1
41   switchport access vlan 4000
42 !
43 interface 10GigabitEthernet 1/1
44   switchport hybrid allowed vlan 40-54
45   switchport hybrid ingress-filtering
46   switchport mode hybrid
47   qos trust tag
48   no spanning-tree
49   stream-id 54
50 !
51 interface 10GigabitEthernet 1/2
52   no spanning-tree
53 !
54 interface 10GigabitEthernet 1/3

```

Figure 19: Ausschnitt TSN-Switch-Konfiguration

Hierbei wird bei Paketen vom Raspberry Pi oder Endpoint, die einen Port erreichen (ingress), zunächst eine Stream-Identifikation und anschließend eine FRER-Generation-Instanz gebildet. So können Daten über ein eigenes VLAN auf zwei Wegen durchs Netzwerk geschickt werden, bis sie dann am Zielswitch aus beiden Richtungen wieder abgefangen werden. Von hier aus wird eines der Pakete ans Endgerät weitergeleitet und das Duplikat verworfen. Dieses Verhalten sorgt demnach für Daten-Redundanz bei gleichzeitiger Begrenzung der Bandbreite. Dafür wurden in dem Raspberry Pi ARP-Einträge angelegt, sodass diese die Ziel-MAC-Adressen der Endpoints bereits kannte, und dieser somit keine ARP-Requests (Broadcasts) mehr verschicken muss, sondern die Endpoints direkt über Unicast erreichen kann. Bei den Unicast-Paketen kann darauf wieder kleinteilig FRER-Redundanz angewandt werden, mit Ausnahme der Endpoints, da diese direkt am Switch mit dem Raspberry Pi hängen.

Da TSN einen wichtigen Teil von DELIA und den Demonstratoraufbau umfasst, wurde im Laufe der Projektdauer viel Rücksprache mit dem eigentlich angedachten TSN-Switch-Lieferanten AED gehalten. Weitere Gespräche zu TSN mit der Universität Hamburg führten dazu, dass sich die ZAL GmbH dazu entschloss ein Paper zu TSN in Kooperation mit der Universität Hamburg im Jahr 2021 zu veröffentlichen. Hierbei entstand das Paper *On the Security of IEEE 802.1 Time-Sensitive Networking* [1], welches im 2021 IEEE International Conference on Communication Workshops (ICC Workshops) veröffentlicht wurde.

11 Die wichtigsten Positionen des zahlenmäßigen Nachweises

11.1 Material

Das Materialbudget wurde für die folgenden übergeordneten Anschaffungen verwendet:

- Elektronik-Boards (Carrier Board, Processing Board, IO-Board) für den Aufbau des Demonstrators
- Herstellkosten des ZAL Endpoints (Leiterplatten-Fertigung, Bestückung)
- Elektronische und mechanische Kleinteile

Hauptsächlich stand immer die Demonstration im Kabinendemonstrator im Fokus.

11.2 Personalkosten

Im Vergleich zum beantragten Personalbudget waren die tatsächlichen Personalkosten um ca. 4,3 % höher. Dies liegt daran, dass das Konzept für die Abschlussdemonstration geändert werden musste, da die DELIA-Plattform leider nicht die gewünschten Funktionalitäten aufwies. Dadurch entstand ein Mehraufwand.

11.3 Reisekosten

Aufgrund der Corona-Pandemie wurde das Reisebudget nicht ausgeschöpft. Die meisten Projektmeetings wurden als Webkonferenz abgehalten.

11.4 Abschreibungen auf vorhabenspezifische Anlagen

Abweichend von der ursprünglichen Planung waren keine Investitionen notwendig, um das Projekt umzusetzen. Ein Teil der beantragten Mittel wurde zwischenzeitig in Materialkosten umgewidmet.

11.5 Sonstige unmittelbare Vorhabenkosten

Unter den sonstigen unmittelbaren Vorhabenkosten liefen hauptsächlich die Kosten für das Betreiben der Projekt-Webseite (www.delia-project.com).

12 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die ZAL GmbH ist ein 2009 gegründetes kleines Unternehmen in Hamburg. Es soll als Schnittstelle zwischen OEM, Zulieferer und Forschungseinrichtung den weltweit drittgrößten Standort der zivilen Luftfahrt sichern und kontinuierlich ausbauen.

Derzeit befindet sich das Unternehmen jedoch noch in der wirtschaftlichen Aufbauphase. Dies betrifft insbesondere den Aufbau der technologischen Kompetenzen, die zur Wahrnehmung der Schnittstellen- und Ankerfunktion strategisch notwendig sind.

Ohne eine Zuwendung wäre der Aufbau spezialisierter technologischer Leistungen in der Luftfahrt nicht möglich, da die ZAL GmbH allein nicht das erhöhte wirtschaftliche und wissenschaftlich-technische Risiko tragen könnte.

Durch die Förderung konnte die ZAL GmbH sich an diesem komplexen Projekt beteiligen und so ihr Kompetenzwissen aufbauen und vertiefen. Die Untersuchung neuer Kommunikationsprotokolle und die Integration dieser in ein Gesamtsystem barg wirtschaftliche und wissenschaftlich-technische Risiken. Ohne Förderung hätte daher dieses Projekt nicht bzw. nur in einem kleineren Maße umgesetzt werden können. Das Ergebnis dieses Projektes umfasst jedoch nicht nur den Aufbau von technischer Expertise, sondern es entstand außerdem der ZAL

Endpoint, welcher für zukünftige Industrie- und Forschungsprojekte weiterverwendet werden kann.

13 Verwertbarkeit und voraussichtlicher Nutzen

Die ZAL GmbH ist sowohl Dienstleister als auch Inhaber eigener Kompetenzen in speziellen Themengebieten. Im Bereich der digitalen Kabine und speziell der Integration von neuen Sensoren mit kundenspezifischen Anforderungen konnte die ZAL GmbH durch die Förderung Ihre Kernkompetenzen weiter ausbauen und vertiefen.

Zusätzlich konnte die ZAL GmbH wertvolles Verständnis und Know-how im Bereich der Konfiguration von eingebetteten Systemen und Sensorik sammeln, wodurch sich potentiell vielseitige, sowie auf Kundenanforderungen anpassbare und kostengünstige Lösungen im Bereich der Hard- und Software ergeben haben. Die Zusammenarbeit mit den Projektpartnern von DELIA war stets interessant und fruchtbringend, sodass sich auch Folgeprojekte entwickelt haben, bei welchen die ZAL GmbH z.B. mit Solectrix und der Universität Stuttgart zusammenarbeitet.

Zu den Projektergebnissen gehört der im DELIA-Projekt entwickelte Demonstrator des *ZAL Endpoint*, eine mikrocontrollerbasierte Recheneinheit, die zahlreiche intelligente Aufgaben in der Flugzeugkabine übernehmen kann.

Der *ZAL Endpoint* und seine Software wurden nach Projektende in einem industriellen Forschungsprojekt weiterentwickelt, sodass der Endpoint dort zum Einsatz kommen kann (z. B. zur Ansteuerung von Lautsprechern und Sensorik).

Die Erkenntnisse aus dem DELIA Projekt werden zu weiteren Entwicklungen führen; so wird z. B. bis voraussichtlich Mitte 2023 ein weiterer Endpoint entwickelt, der eine noch höhere Leistungsfähigkeit aufweist und für rechenintensive Aufgaben gedacht ist (z. B. KI-basierte Bildauswertung).

Im LuFo-Projekt DaKliF (Start: Juli 2023) ist die Entwicklung eines weiteren Endpoint-Demonstrators geplant, der besonders energieeffizient und ressourcenschonend ist.

Zu den Projektergebnissen gehört zudem die Erweiterung des Kabinenteststands der ZAL GmbH, sodass dort weitere Testmöglichkeiten entstanden sind; dies ermöglicht die weitere Vergrößerung des Angebotsspektrums der ZAL GmbH für weitere Forschungstätigkeiten, erhöht ihre Wettbewerbsfähigkeit und sichert Arbeitsplätze am Standort Hamburg.

Auch das Know-How bezüglich Netzwerktechnologien wie *Time Sensitive Networking*, das im DELIA-Projekt gewonnen werden konnte, konnte bereits in einem industriellen Forschungsprojekt erfolgreich eingebracht und dort weiter ausgebaut werden.

Der wissenschaftliche Erkenntnisgewinn und die technischen Innovationen des Projekts liegen insbesondere in den Bereichen Hardware- und Softwareentwicklung von *Eingebetteten Systemen*, *Netzwerkkommunikation* und *Netzwerktechnologien* (insb. *Time Sensitive Networking*).

Die Entwicklungen und die gewonnene Expertise werden in zukünftigen industriellen und geförderten Forschungsprojekten eingesetzt (siehe auch 3b, 3d). Zu den aktuellen und potenziellen Partnern für Folgeprojekte gehören insbesondere Luftfahrtunternehmen wie Airbus, Diehl Aviation, Lufthansa Technik und Safran.

Ein Teil der Projektergebnisse werden bereits im laufenden LuFo VI-1-Projekt „VERDIKA“, in welchem Verfahren zum automatisierten Testen von Kabinennetzwerken erforscht werden, verwertet und dort weiterentwickelt.

Im beantragten LuFo VI-3-Projekt „DaKliF“ werden für den im DELIA-Projekt entwickelten *ZAL Endpoint* Softwarealgorithmen erforscht werden, mit denen sich die Energieeffizienz in der

Flugzeugkabine steigern lässt (Projektstart 07/2023). Durch weitere Entwicklungen wird auf Basis der Erkenntnisse von DELIA eine Endpoint-Familie angestrebt, mit dem Ziel, die unterschiedlichsten Anforderungen an effiziente Flugzeugkabinen abdecken zu können.

Diese Forschungstätigkeiten erweitern die Expertise der ZAL GmbH weiter, erhöhen die Wahrscheinlichkeit von industriellen Forschungsaufträgen und sichern ihre Wettbewerbsfähigkeit.

14 Fortschritt auf dem Gebiet des Vorhabens bei anderen Stellen

Nach Kenntnisstand des Zuwendungsempfängers sind während der Durchführung des Vorhabens keine Arbeiten von Dritten entstanden, die der geplanten Verwertung der Projektergebnisse entgegenstehen.

Während der Projektlaufzeit sind im Consumer-Bereich weitere Versionen kommerzieller Mikrocomputer auf den Markt gebracht worden, z. B. der „Raspberry Pi 4“. Im Gegensatz zu dem im DELIA-Projekt entwickelten *ZAL Endpoint* werden solche Recheneinheiten nicht den Anforderungen der Flugzeugkabine gerecht. Der *ZAL Endpoint* hingegen lässt sich mit der in Flugzeugkabinen üblichen Versorgungsspannung von 28V betreiben, ist auf eine Netzwerkeinbindung mittels *Single Pair Ethernet* ausgelegt und verfügt über ein breites Schnittstellenspektrum zur Anbindung unterschiedlichster Sensoren und Aktoren.

Im Bereich der Netzwerktechnologien und der Netzwerkkommunikation, die ebenfalls im DELIA-Projekt erforscht wurden, sind ebenfalls keine Fortschritte während der Projektlaufzeit bekannt geworden, die Einfluss auf den Nutzen der Projektergebnisse haben.

15 Erfolgte Veröffentlichungen des Ergebnisses

Ein Teilergebnis des Vorhabens wurde unter dem Titel „*On the Security of IEEE 802.1 Time-Sensitive Networking*“ [1] im 2021 IEEE International Conference on Communication Workshops (ICC Workshops) veröffentlicht.

Literaturverzeichnis

- [1] D. Ergenç, C. Brühlhart, J. Neumann, L. Krüger and M. Fischer, On the Security of IEEE 802.1 Time-Sensitive Networking, Germany: IEEE, 2021.
- [2] IEEE, "802.1CB-2017 - IEEE Standard for Local and metropolitan area networks--Frame Replication and Elimination for Reliability," 2017.
- [3] B. Fang, Q. Li, Z. Gong and H. Xiong, "Simulative Assessments of Credit-Based Shaping and Asynchronous Traffic Shaping in Time-Sensitive Networking," in *ICAIT*, China, 2020.
- [4] D. E. M. F. Nurefsan Sertbas Bülbül, "Towards SDN-based Dynamic Path Reconfiguration for Time Sensitive Networking," in *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, 2022.
- [5] G. Heiser, "The role of virtualization in embedded systems," in *Association for Computing Machinery*, New York, NY, United States, 2008.
- [6] J. Q. D. C. Alexandre Santos, "Realizing Zenoh with programmable dataplanes," in *ANCS '21: Proceedings of the Symposium on Architectures for Networking and Communications Systems*, 2022.
- [7] S. S. & W. F. Fabien Geyer, "Performance Evaluation of an Ethernet-Based Cabin Network Architecture Supporting a Low-Latency Service," in *Lecture Notes in Computer Science* .
- [8] J. Z. J. S. & L. Y. Haigang Yang, "Review of advanced FPGA architectures and technologies," in *Journal of Electronics*, China, 2014.
- [9] L. L. M. D. G. P. S. S. M. Mohammad Ashjaei, "Time-Sensitive Networking in automotive embedded systems: State of the art and research opportunities," in *Journal of Systems Architecture*, 2021.
- [10] E. D. & F. Mathias, "On the Reliability of IEEE 802.1CB FRER," in *IEEE Conference on Computer Communications*, 2021.