

Schlussbericht

Verbundvorhaben

FLYSMART – FBW23

Titel des Teilvorhabens: **FBW-Plattformtechnologie**

Zuwendungsempfänger: **Universität Stuttgart,
Institut für Luftfahrtsysteme (ILS)
Institut für Flugmechanik und
Flugregelung (iFR)**

Förderkennzeichen: **20V1108N**

Autoren: **Prof. Dr.-Ing Reinhard Reichel
Prof. Dr.-Ing. Walter Fichter**

Laufzeit des Vorhabens: **01.01.2012 – 30.06.2015**

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

Stuttgart, Oktober 2015

Prof. Dr.-Ing. Reinhard Reichel
(Projektleiter)



Inhaltsverzeichnis

Glossar.....	3
1. Kurzdarstellung des Projekts.....	5
1.1 Aufgabenstellung.....	5
1.2 Voraussetzungen.....	6
1.3 Planung und Ablauf des Projekts.....	7
1.4 Stand der Wissenschaft und Technik.....	8
1.5 Zusammenarbeit mit anderen Stellen.....	11
2. Darstellung der Projektergebnisse.....	12
2.1 Zuwendungsverwendung und Ergebnisse.....	12
2.1.1 Vorgegebene Zielsetzung.....	12
2.1.2 Ergebnisse der Arbeitspakete.....	13
2.2 Zusammenfassung des zahlenmäßigen Nachweises.....	44
2.3 Notwendigkeit und Angemessenheit der Arbeit.....	45
2.4 Nutzen und Verwertbarkeit.....	45
2.4.1 Wirtschaftlicher Nutzen und Verwertbarkeit.....	45
2.4.2 Wissenschaftlich / technischer Nutzen und Verwertbarkeit.....	45
2.4.3 Wissenschaftliche und wirtschaftliche Anschlussfähigkeit.....	46
2.5 Fortschritt anderer Stellen.....	47
2.6 Veröffentlichungen.....	47
Anlagen.....	48
Referenzen.....	48



Glossar

AAA-Prozess

Entspricht einem Prozess wie folgt:

- Eine neue FCP23-Instanz wird von System-Ingenieuren auf hoher Systemebene definiert. Die Definition wird mit Hilfe einer GUI festgelegt.
- Der AAA-Prozess, umgesetzt mit Hilfe einer Tool-Suite, erzeugt daraus automatisch
 - das Plattform-Management (PLAMA) sowie die Konfigurationsdaten für Software-Driver und spezielle Konfigurationsdaten für das Betriebssystem (für alle Module der FCP23-Instanz),
 - die dazugehörigen Requirements der Dokumente YRD, SRD und SDD,
 - die zur Verifikation dieser Requirements erforderlichen Test-Prozeduren.

Der heutige System- und Software-Entwicklungsaufwand reduziert sich somit weitgehend auf die Ebene der Systemkonzeption.

ATOL

Automatischer Take-Off und Landung.

Basic-Service (BS)

Entspricht einer einfachen Software-Einheit, deren Funktion von außen durch Konfigurationsdaten angepasst werden kann. Diese Software-Einheit ist quasi als ein einfaches Software-Produkt zu verstehen, fertig qualifiziert gegenüber einer Usage Domain. Basic-Services können fast beliebig miteinander kombiniert werden, um so daraus eine komplexe PLAMA-Funktion zu bilden.

FCLs/Funktionen

Die Flight Control Laws umfassen alle Regelgesetze, die zur Lenkung und Regelung eines Flugzeugs benötigt werden.

In diesem Projekt sind dies: Planer, Missionsmanagement, Path-Tracker, Guidance und Innere Kreise.

FCP23

Die Flight Control Plattform für CS23-Fly-by-Wire-Systeme ist – stark vereinfacht – eine Bibliothek von Hardware-Modulen, Basic-Services (Software), OS und Driver. Daraus lässt sich in variabler Weise ein Fly-by-Wire-System (ohne Laws) erstellen.

FCP23-Instanz

Dies ist ein mit Hilfe der FCP23 erstelltes Fly-by-Wire-System. Dieses umfasst alle Avionik-Module, Aktuatoren und Sensoren sowie das Fehler- und Redundanz-Management mit BITE für eine gesamte FCP23-Instanz inkl. Redundanz-Management für Sensoren und Aktuatoren – also ein ohne Laws komplett funktionsfähiges Fly-by-Wire-System.



Law/Systemfunktion

Ein Law entspricht i.d.R. einem Regel- oder Steuergesetz, beispielsweise einem Flugregelgesetz oder einer Autopilotenfunktion. Ein Law ist „simplex minded“ entwickelt, umfasst also keinerlei Plattform-Management-Funktionen wie Redundanz-Management u.ä.

Mode (Betriebs-Mode)

Der Betrieb eines Fly-by-Wire-Systems muss den gesamten Life-Cycle abdecken. Dies geschieht mit Betriebs-Modes. Sie umfassen jegliche Aktivität, sowohl während als auch außerhalb einer Mission. Typische „Missions“-Modes sind „Power-On/Down“, „Preflight/Aftflight BIT“ (mit seinen hierarchischen Unterteilungen in „Modul-BIT“, „FCP-BIT“, „System-BIT“), „Dispatchability Ermittlung“ etc. Typische „Nicht-Missions“-Modes sind „Maintenance“ mit seinen Sub-Modes wie „Interaktive-Mode“, „Debug-Mode“, „Selectiv-BIT-Mode“, „Installation-Mode“ u.ä.

Mode-Management

Ist ein Teil von PLAMA(S) und steuert die Betriebs-Modes.

Modul (Avionik-Modul)

Entspricht einer Simplex- oder Duplex-Avionikeinheit. Die FCP23 umfasst CPM (Duplex-Core-Processing Module), IOM (Simplex oder Duplex IO-Module), ACM (Duplex-Actuator Control Module) und PSM (Power-Supply-Module).

OS (Operating System)

Ist ein Betriebssystem.

Peripherie-Einheit

Entspricht typischerweise einem Sensor oder Aktuator.

Plattform-Management (PLAMA, PLAMA(C), PLAMA(S))

Umfasst alle Managementaufgaben einer FCP23-Instanz inkl. ihrer Peripherie-Einheiten. Dazu unterteilt sich das Plattform-Management in PLAMA(C) (*Consolidation*) und PLAMA(S) (*Supervisor*). **PLAMA(C)** umfasst die gesamte Kommunikation, Fehlererkennung, Datenkonsolidierung, das Framework für Laws u.ä. **PLAMA(S)** setzt auf PLAMA(C) auf und trifft Entscheidungen. Diese Entscheidungen betreffen das Replica-Management und das Mode-Management.

Plattform

Entspricht grob formuliert einer Bibliothek von Hardware- und Software-Einheiten, aus denen durch Kombination etwas Ganzes, also eine Plattform-Instanz gebildet werden kann.

Replica-Management

Ist ein Teil von PLAMA(S) und entscheidet über replizierte Ressourcen (Module, Peripherie-Einheiten) bzgl. on/off, Master/Slave/Shadow, active/standby, fehlerbedingte Reallozierung von Laws etc. Das Replica-Management stellt hier also einen speziellen Bereich des Redundanz-Managements dar.

Tool-Suite

Ist die tool-basierte Umsetzung des AAA-Prozesses. Nach Eingabe der Systemvorgaben führt die Tool-Suite den AAA-Prozess weitgehend automatisch aus.



1. Kurzdarstellung des Projekts

1.1 Aufgabenstellung

Die Entwicklung und Zertifizierung von Fly-by-Wire-Systemen für automatisierte Arbeitsflugzeuge und intelligente Transportflugzeuge der Klassen EASA CS23/Class..III ist eine Herausforderung und nur über grundsätzlich neue Techniken möglich. Die AAA-Plattformtechnologie kann als ein solcher Ansatz betrachtet werden. Diese Technologie zeichnet sich durch einen streng layerhaften Aufbau der Gesamt-Software mit dem Plattform-Management als zentralem Layer aus. Damit ist für Laws die Systemkomplexität mit Redundanz und Fehlertoleranz nicht mehr sichtbar. Sie können „simplex minded“ entwickelt werden. Der AAA-Prozess ermöglicht es, für ein neu zu entwickelndes Fly-by-Wire-System das Plattform-Management weitgehend automatisch zu erzeugen, zu dokumentieren und zu verifizieren. Damit lassen sich derartige Systeme in Zukunft äußerst effizient entwickeln und qualifizieren. Zentrale Aufgabenbereiche dieses Teilvorhabens waren die Weiterentwicklung und Validierung des Axx-Prozesses zur weitgehend automatischen Erzeugung des Plattformmanagements im Hinblick auf einen Einsatz im industriellen CS23-Umfeld sowie die Entwicklung und Validierung der Funktionen (Laws) zur Erweiterung der automatisierten Missionsfähigkeit (inkl. ATOL) für zukünftige Arbeits- und Transportflugzeuge EASA CS23.

Die Umsetzbarkeit des Axx-Prozesses ist durch zahlreiche Forschungsprojekte erbracht. Seine Flexibilität ist durch Demonstratoren im CS23-, Hubschrauber- und Automotivbereich grundsätzlich verifiziert. Dennoch war noch ein deutlicher Forschungsschritt erforderlich, um diese Technologie in effizienter Weise im industriellen CS23-Umfeld zum Einsatz zu bringen.

Auch auf funktioneller Seite mussten grundsätzliche Lücken geschlossen werden. Dies betraf die Automatisierung möglichst vieler Flugzeugsysteme (Systemtransparenz für den Piloten) sowie die grundsätzlich vorhandene ATOL-Fähigkeit zur Gewährleistung zentraler zukünftiger Missionsanforderungen.

Die Aufgaben im Detail waren:

1. Weiterentwicklung der Plattform-Middleware für CS23 Anwendung. Den unteren Layern der Middleware ist eine Software-Struktur aufzuprägen, die zu einem vereinfachten Meta-Modell der Middleware und somit zu einer Vereinfachung der Spezialisierungs-Tool-Suite führt. Kernherausforderung ist hierbei, die enorm hohe Heterogenität externer Signale, Daten, Busse und Geräte durch wenige generische Funktionselemente im Signal-Layer der Middleware abzudecken.
2. Weiterentwicklung der Spezialisierungs-Tool-Suite vor dem Hintergrund der Anwendung durch Entwickler des industriellen CS23 Umfeldes. Das bedeutet:
 - Entwicklung und Umsetzung eines Konzeptes zur deutlichen Erhöhung der Datentransparenz für Anwender im CS23 Umfeld.
 - Entwicklung eines auf CS23 zugeschnittenen Meta-Modells der Middleware.
 - Entwicklung und Umsetzung eines neuen Ansatzes zur Abbildung der Auto-Instanzierungs-Regeln (Synthesis Rules).



3. Automatisierte Dokumenten- und Testcase-Erzeugung.

Grundlage der automatisierten Instanziierung der Middleware ist ihr absolut generischer Aufbau. Dieser generische Aufbau ist auch die Grundlage zur Automatisierung von Dokumentation und Verifikation (automatisierte Testcase Erzeugung). Dazu gilt es, ein grundlegendes Konzept zu ermitteln. Die Instanziierung der Middleware aus generischen Funktionseinheiten des Plattform-Managements lässt sich weiter nutzen zum Aufbau eines teilweise generisch basierten Sicherheitsnachweises. Beispielsweise basiert das fehler-tolerierende Verhalten des Kerns der Fly-by-Wire Plattforminstanz auf einer generischen Protokollschicht. Diese muss zwar auf jede Anwendung hin neu spezialisiert werden, basiert jedoch selbst ausschließlich auf generischen Mechanismen. Folglich ist auch dieser Bereich des Sicherheitsnachweises generischer Natur. In diesem Vorhaben galt es, ein Konzept eines generischen Sicherheitsnachweises für die CS23 relevante Technologie „Flexible Plattform“ zu erarbeiten.

4. Automatisierung weiterer Systeme.

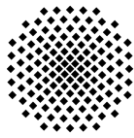
Das hoch automatisierte Arbeitsflugzeug sowie das intelligente Transportflugzeug (Small Sized A/C) setzt ein weitgehend automatisiertes Verhalten der Flugzeugsysteme voraus (Systemtransparenz selbst im Fehlerfall). Hierzu ist ein Konzept zur weitgehenden Automatisierung der absolut sicherheitskritischen elektrischen Energieversorgung (in CS23 Flugzeugen nicht gegeben), Klappensteuerung (Flaps, Spoiler), Fahrwerkssteuerung, Bugradlenkung, Bremsen, Antriebseinheiten (Engine) zu erarbeiten. Da die Fly-by-Wire Plattform Zugriff auf alle notwendigen Komponenten dieser Systeme hat, lassen sich die Automatisierungsfunktionen einfach als Systemfunktionen entwickeln und als Applikationen implementieren.

5. Erweiterung der Funktionalität.

Notwendige Voraussetzung für das weitgehend automatisierte Arbeitsflugzeug und das intelligente zukünftige Transportflugzeug ist die Erweiterung der automatisierten Missionsfähigkeit auf ATOL. Automatischer Start und automatische Landung mit Ausrollen und Abbremsen sollen dabei ohne externe Landesysteme auskommen.

1.2 Voraussetzungen

AAA-Plattformtechnologie für Avioniksysteme ist Kernforschungsgebiet und somit Kernkompetenz des ILS der Universität Stuttgart. Diese Technologie kommt zurzeit einer Reihe deutscher Unternehmen der Luftfahrt und des Automotivbereichs zugute (Stemme, Rheinmetall Defence Electronics, Airbus Helicopters Deutschland, Liebherr-Aerospace Lindenberg, Diehl Aerospace, Daimler). Die Technologie wurde seit der Gründung des ILS im Jahre 2003 konsequent verfolgt und in den letzten fünf Jahren schrittweise in folgenden Forschungsprojekten weiterentwickelt bzw. zur Anwendung gebracht: PEIT, SPARC, HaveIT, SAFAR (alle EU) sowie LAPAZ, HC-FBX, SAFER FLIGHT (LUFO) (Projekte der letzten 5 Jahre). Ein Avionik- oder FBW-System lässt sich aus der Plattformtechnologie im Wesentlichen dadurch generieren, indem man die Middleware mit Hilfe einer Tool-Suite spezialisiert.



Die Entwicklung von Funktionen zur Automatisierung von Systemen der Luft- und Raumfahrt ist eines der Kerngebiete des iFR der Universität Stuttgart. Diese Aktivitäten finden häufig im Rahmen von Kooperationen mit entsprechenden Firmen und anderen Forschungsorganisationen statt. Funktionen für den autonomen/automatischen Flug von kleinen unbemannten Flächenflugzeugen (<10kg), wie onboard Bahnplanung mit Hindernisvermeidung, nichtlineare Filterung oder kamerabasierte Landungen, wurden im Rahmen früherer Projekte am iFR entwickelt. Einige der am iFR entwickelten Autopilotenfunktionen finden im Bereich der Meteorologie und der Photogrammetrie Anwendung. Neben diesen Vorarbeiten stand seitens Diamond Aircraft ein Simulationsmodell zur Verfügung, das um ein Modell des Bodeneffekts und des Fahrwerks ergänzt werden musste.

1.3 Planung und Ablauf des Projekts

Der Ablauf des Projekts erfolgte auf Grundlage der im Antrag skizzierten Planung in drei Hauptarbeitspaketen (1100, 1200, 1300). In den Arbeitspaketen AP11xx wurden die wesentlichen Anforderungen an das zu realisierende FBW-System sowie ein Konzept zum automatisierten Sicherheitsnachweis erarbeitet. Die Erforschung und Entwicklung der Technologie gemäß der oben beschriebenen Aufgabenstellung erfolgte in den Arbeitspaketen AP12xx. Auf Basis der Ergebnisse aus AP12xx wurde das in AP11xx spezifizierte System realisiert und anhand eines „Iron Bird“ verifiziert sowie anschließend in den Erprobungsträger Diamond DA42 integriert und im Flugversuch evaluiert (AP13xx).

ID	Arbeitspaket	2012				2013				2014				2015	
		Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2
AP1100	Grundlegende Req. zur FBW-Plattform														
AP1110	System Requirements Capture / Synthesis	■													
AP1120	FBW-Plattforminstanz	■													
AP1130	Safety-Prozess	■	■	■	■										
AP1200	Grundlagen FBW-Plattformtechnologie														
AP1210	Generische Plattform-Middleware		■	■	■	■	■	■	■	■	■	■	■	■	
AP1220	Verfahren z. Spezialisierung der gen. Plattform		■	■	■	■	■	■	■	■	■	■	■	■	
AP1240	Systemfunktionen		■	■	■	■	■	■	■	■	■	■	■	■	■
AP1300	FBW-Validierungsplattform														
AP1320	Instanziierung der Plattform-Middleware			■	■	■	■	■	■	■	■	■	■	■	
AP1330	Plattform/System Integration & Verifikation			■	■	■	■	■	■	■	■	■	■	■	
AP1340	Flugzeugint. & PtF			■	■	■	■	■	■	■	■	■	■	■	
AP1350	Flugtests & Auswertung			■	■	■	■	■	■	■	■	■	■	■	■

Tabelle 1: Balkenplan zur zeitlichen Durchführung des Projekts in Arbeitspaketen



Zur Evaluierung im Flugversuch waren umfangreiche Testflüge mit dem Erprobungsträger erforderlich. Aufgrund außerordentlicher Arbeiten innerhalb eines Zertifizierungsprojektes des assoziierten Partners Diamond Aircraft Industries verzögerte sich die Flugzeugintegration, Flugfreigabe (PtF) sowie die Flugerprobung. Mit der genehmigten kostenneutralen Verlängerung um sechs Monate konnte FLYSMART erfolgreich abgeschlossen werden. Die angepasste zeitliche Planung ist mittels des Balkenplans (Tabelle 1) dargestellt.

1.4 Stand der Wissenschaft und Technik

CS23-Flugzeuge werden heute nicht mit Fly-by-Wire-Steuerungen ausgerüstet. Ein zentraler Grund hierfür liegt bei den bisher extrem hohen Entwicklungskosten (inkl. Qualifikation), aber auch bei den hohen Produktionskosten eines solchen Fly-by-Wire-Systems. Daher ist die Entwicklung von FBW-Systemen weitgehend auf die Flugzeugkategorie „Air Transport“ bzw. auf das obere „Regional“ Segment beschränkt. Erst ein neuer Ansatz, der die Entwicklungs- und Produktionskosten deutlich reduziert, wird zukünftig Fly-by-Wire-Steuerungen für CS23-Flugzeuge ermöglichen. Die AAA-Plattformtechnologie ist ein solch neuer Ansatz. Ein ähnlicher Ansatz ist bislang nicht bekannt.

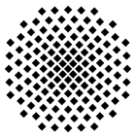
Aspekt: Avionik-Architektur

Abbildung 1 zeigt die Evolution der Avionik für komplexe Systeme nach [1], [2], [3], [4], [5], [6] und [7]. Technischer Standard für sicherheitskritische Systeme ist in weiten Bereichen noch immer die „Federated Architecture“. Anders sieht es im Bereich weniger sicherheitskritischer Systeme aus, bei denen heute der Integrierte Architekturansatz als Standard betrachtet werden darf.

Die Flexible Plattform¹ ([8], [9], [10], [11], [12] und [13]) ist als eine „Advanced Integrated Avionics Architecture“ einzustufen. Sie stellt gegenüber IMA und IMA2G einen wesentlichen technologischen Fortschritt dar:

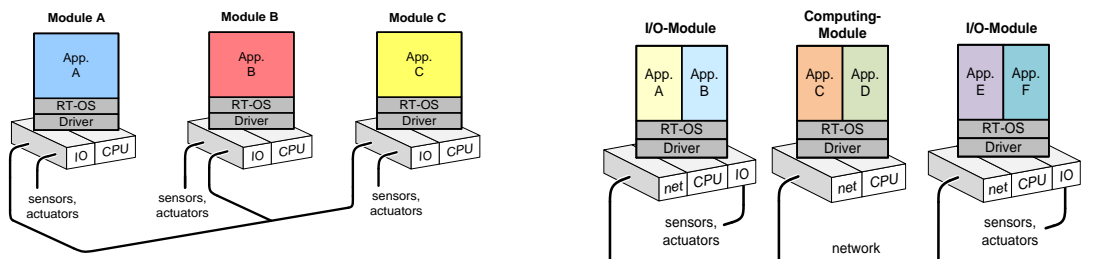
- Der Abstraktionsgrad ist durch die Einführung von PLAMA als Middleware deutlich höher als jener bei IMA oder IMA2G ([14], [15]). Dort umfasst der Abstraktionsgrad im Wesentlichen nur die reine Hardware bzw. bei IMA2G voraussichtlich auch die Kommunikation zwischen Modulen. Das eigentliche System-Management ist in beiden Fällen Teil der Applikation bzw. des Laws.
- Das System-Management der Flexiblen Plattform ist bereits heute für absolut sicherheitskritische Anwendungen wie Fly-by-Wire ausgelegt und umfasst alle Funktionen, einschließlich des Redundanz-Managements innerhalb in sich redundanter Module, zueinander redundanter Module und (redundanter) Peripherie-Einheiten. Dies geht weit über IMA und IMA2G hinaus.
- Grundsätzlich neu ist auch, absolut ausfallkritische Systeme wie Fly-by-Wire-Systeme überhaupt mit einer „Integrated Avionics Architecture“ bzw. sogar einer „Advanced Integrated Avionics Architecture“ umzusetzen. Airbus-Fly-by-Wire-Systeme basieren bis einschließlich dem Airbus A350 auf einer „Federated Avionics Architecture“. Das Boeing B787 Fly-by-Wire-System weist zwar in Teilen eine „Integrated Avionics Architecture“ auf, das Basissystem ist dafür umso konservativer ausgelegt und basiert auf einer *prozessorlosen*

¹ Eine verallgemeinerte Form der FCP23



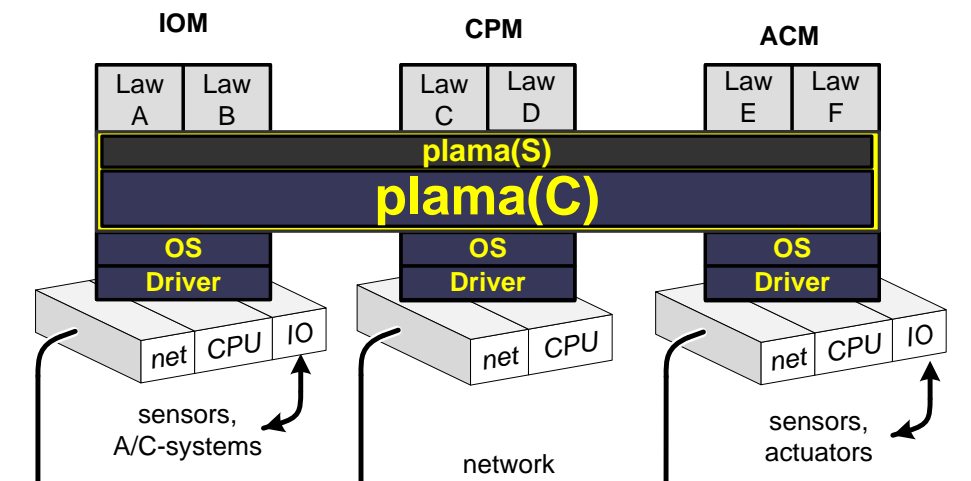
„Federated Avionics Architecture“. Erst das Airbus Patent [15] bezieht sich auf eine reine IMA basierte Fly-by-Wire-Systemrealisierung.

Das System-Management der Flexiblen Plattform umfasst Funktionen zur fehlerbedingten Reallozierung von Laws/Applikationen im Flug mit dem Ziel einer „maximalen/maximal sicheren“ Performance des Gesamtsystems für jeden fehlerbedingten Systemzustand. Dies geht ebenfalls weit über IMA und IMA2G hinaus.



a) Federated Avionics Architecture

b) Integrated Avionics Architecture (IMA)



c) Advanced Integrated Avionics Architecture (Flexible Platform / FCP23)

Abbildung 1: Evolution der Avionik Architektur

Aspekt: Plattform

Der Plattformgedanke ist generell nicht neu ([16], [17], [18] und [19]). Er findet hauptsächlich für Enterprise-Systeme oder Systeme der Prozessautomatisierung Anwendung. Plattformansätze für verteilte Instanzen basieren klassischerweise auf einer Middleware.



Die wesentlichen Eigenschaften solcher Plattformen sind typischerweise:

- Kein Echtzeitbezug (bei Enterprise-Systemen).
- Keine sicherheitskritische Anwendung, kein Redundanz-Management. Wenn Replika überhaupt notwendig sind, dann lediglich als „Shadow-Server“ zur Datensicherung im Falle eines System-Crashes bei Enterprise-Systemen.
- Homogenität bzgl. der verwendeten Hardware-Modul- und Peripherie-Einheiten-Typen. Inhomogene Peripherie-Einheiten werden i.d.R. nicht berücksichtigt.
- Kein komplexes, zentral gesteuertes Mode-Management.

Aufgrund dieser im Vergleich zur Flexiblen Plattform sehr eingeschränkten Anforderungen reduziert sich die Middleware dieser Plattformen klassischerweise auf eine reine Kommunikationsschicht, welche die Systemverteilung abstrahiert.

Eine im Automobilbereich entwickelte Plattform ist AUTOSAR [20]. Diese Plattform ist im Ansatz durchaus vergleichbar mit IMA,

- umfasst aber keine Fehlertoleranz durch Hardware-Replika, also kein Redundanz-Management.
- Peripherie-Einheiten sind grundsätzlich nicht Teil des Plattform-Managements. Das Management der Peripherie-Einheiten ist Teil der Applikation bzw. des Laws.
- Komplexes Mode-Management ist nicht Teil des Plattform-Managements, sondern Teil der Applikation bzw. des Laws.

Für sicherheitskritische Anwendungen in der Avionik, mit dem hohen Anspruch der Flexiblen Plattform bzgl. Systemabstraktion, sind die bisher existierenden Plattformsätze nicht geeignet. Sie enthalten nicht die für die hohe Abstraktion erforderlichen Managementfunktionen.

Aspekt: Automatisierte Missionsfähigkeit (inkl. ATOL)

Automatische Landefunktionen (CAT IIIb) sind in CS25 Flugzeugen bereits seit vielen Jahren Standard. Sie sind auf am Boden installierten ILS-Systeme angewiesen, die präzise das automatische Einfliegen und Abfliegen eines Leitstrahls einschließlich Abfangen, Ausrichten, Aufsetzen, Ausrollen und Abbremsen ermöglichen. Im Juni 1989 wurde an der Universität Braunschweig (Prof. Schänzer) die weltweit erste automatische auf GPS basierende Landung demonstriert. Ebenfalls GPS basiert wurden Mitte der 90er Jahre am NASA Ames Research Center automatische CAT IIIb Landungen durchgeführt. Beide Vorhaben benötigten keinen vom Flughafen bereit gestellten Leitstrahl oder Gleitpfad. Die GPS / WAAS (SBAS) basierte automatische Landung für ein General Aviation Flugzeug (Cirrus SR-22) mit MTOW < 2t ist seit 2010 ein Forschungsgebiet am MIT International Center for Air Transportation in Zusammenarbeit mit der Firma Avidyne. Die Autoland Funktion führt das Flugzeug sowohl in lateraler als auch in vertikaler Richtung.

Dem Antragsteller ist momentan nur ein nationales Projekt bekannt, in dem eine SBAS basierte ATOL-Funktion realisiert werden soll. Im LUFO-Verbundvorhaben LAPAZ (Fa. Stemme (Verbundführer), Technische Universität Berlin, ILS (Universität Stuttgart)) wurde eine ATOL Funktion für einen Motorsegler auf einer Autopiloten-Plattforminstanz entwickelt und flugerprobt.



Im Bereich kleiner Flächenflugzeuge (<5 kg) sind zum Teil moderne und effiziente Verfahren bekannt, die die automatische Bahnplanung, Lenkung und Regelung auf der Basis weniger und ungenauer Sensorinformation ermöglichen (Ref. [21], [22], [23], [24], [25], [26]). Allerdings sind diese Anwendungen hinsichtlich Anforderungen an die Genauigkeit, Robustheit und Zuverlässigkeit nicht vergleichbar.

1.5 Zusammenarbeit mit anderen Stellen

Zusammenarbeit mit Projekt eSafe (Take OFF, FFG – Austria)

Um für FLYSMART die Ressourcen im Bereich der Flugerprobung bei Diamond Aircraft Industries AG zu sichern, wurde eine enge Zusammenarbeit mit dem Projekt eSafe – Emergency Safe Return (gefördert durch das österreichische Programm Take OFF, Projektträger FFG) etabliert. Ziel von eSafe ist die sichere Landung des Flugzeuges bei „Ausfall“ des Piloten. Hierzu wurde in eSafe ein Algorithmus zur automatischen Routenfindung und Routenvorgabe entwickelt und umgesetzt. Dieser Algorithmus ist nicht Teil des in FLYSMART zu realisierenden FBW-Systems, allerdings ergeben sich zwischen eSafe und FLYSMART folgende Synergieeffekte:

- Über ein Interface, welches in FLYSMART bereits für die 4D-Trajektorienfolge umgesetzt wurde, ließ sich die in eSafe realisierte Software grundsätzlich an das FBW-System anbinden und konnte somit nicht nur in einer simulierten Umgebung (bspw. Matlab/Simulink), sondern auch innerhalb eines grundsätzlich zertifizierbaren, fehlertoleranten FBW-Systems evaluiert werden.
- Durch enge Koordination der beiden Projekte standen FLYSMART gesicherte Ressourcen zur Durchführung der erforderlichen, intensiven Flugerprobung des FBW-Systems zur Verfügung. Zeitgleich beschränkt sich der zusätzliche Aufwand auf die Koordination der Flugversuche.

Zusammenarbeit mit Partnern des Projektes LAPAZ

Derzeit existiert eine Zusammenarbeit im Bereich der Flugregelung zwischen dem iFR und dem FMRA (Fachgebiet Flugmechanik, Flugregelung und Aeroelastizität, TU Berlin). Im Bereich des Aktuator-Servo-Loop wurden aktuelle Forschungsergebnisse aus dem Projekt FLYSMART für eine Anwendung im Erprobungsträger Stemme S15 des Projektes LAPAZ übertragen.



2. Darstellung der Projektergebnisse

2.1 Zuwendungsverwendung und Ergebnisse

2.1.1 Vorgegebene Zielsetzung

2.1.1.1 Gesamtzielsetzung

Gesamtziele dieses Teilvorhabens waren die Weiterentwicklung der AAA-Technologie, so dass der Axx-Prozess im industriellen CS23-Entwicklungsumfeld effizient einsetzbar ist und die Entwicklung aller Systemfunktionen für den hochgradig automatisierten Flug. Auf Basis der Ergebnisse wurde ein FBW-System für das Validierungsflugzeug Diamond DA42 realisiert. Die Evaluierung der Korrektheit der Ergebnisse erfolgte mittels Verifikation im Labor (anhand Iron Bird) als auch im Flugversuch durch Abfliegen der spezifizierten Referenzmission.

Wissenschaftliche Arbeitsziele dieses Vorhabens waren:

1. Erarbeitung der Grundlagen für einen teilweise generischen Ansatz zum Nachweis der Ausfallsicherheit von Systemfunktionen, welche auf FCP23-Instanzen integriert sind.
2. Weiterentwicklung PLAMA(C) mit Ziel der automatischen Generierung des Managements von Sensoren und deren Signalübertragung unter Berücksichtigung der Heterogenität der Peripherie im CS23-Anwendungsumfeld.
3. Weiterentwicklung der Spezialisierungs-Tool-Suite unter Berücksichtigung der Anwendbarkeit im CS23-Entwicklungsumfeld.
4. Konzeptionierung, Entwicklung und Umsetzung der Systemfunktionen zur Realisierung einer automatischen 4D-Trajektorienfolge für den automatischen Start (aus dem Stillstand), den automatischen Flug und die automatische Landung (inklusive Ausrollen und Bremsen).

2.1.1.2 Zielsetzung der Arbeitspakete

AP1110- System Requirements Capture / Synthesis

- Z11.1 – Zuarbeit bei der Erstellung grundlegender Anforderungen
- Z11.2 - Definition Sicherheits-, Zulassungs- und Qualifikations-Level
- Z11.3 – Ableiten Technologie-relevanter Anforderungen

AP1120- FBW-Plattforminstanz

- Z12.1 – Zuarbeit beim Ableiten grundlegender Anforderungen an das zu realisierende FBW-System

AP1130- Safety-Prozess

- Z13.1 – Grundlagen für einen teilweise generischen Ansatz zum Nachweis der Ausfallsicherheit

AP1210- Generische Plattform-Middleware

- Z21.1 – Entwicklung eines generischen, spezialisierbaren Ansatzes für Datenquellen mit Datenübertragung
- Z21.2 – Realisierung des generischen Sensor- und Kommunikationsmanagements als Teil von PLAMA(C)



- Z21.3 – Analyse zur Systematisierung von PLAMA(S)

AP1220- Verfahren zur Spezialisierung der generischen Middleware

- Z22.1 – CS23-relevante Weiterentwicklung der Spezialisierungs-Tool-Suite

AP1240- Systemfunktion

- Z24.1 – Konzept zur weitgehenden Systemautomatisierung
- Z24.2 – Erweiterung des nichtlinearen, flugmechanischen Modells und Ableitung linearer Entwurfsmodelle
- Z24.3 – Entwurf eines Missionsmanagements unter Berücksichtigung eines Piloteninterfaces
- Z24.4 – Entwicklung eines Planers für eine fliegbare ATOL-Referenzmission als Positions-Geschwindigkeitsverlauf (4D)
- Z24.5 – Entwicklung einer Guidancefunktion zur Einregelung der 4D-Trajektorie
- Z24.6 – Entwicklung Innerer Regelkreise zur einheitlichen Flugregelung in der gesamten Flugenveloppe und für alle Konfigurationen
- Z24.7 – Theoretische Stabilitätsbetrachtungen und Robustheitsanalyse der Regler
- Z24.8 – Aufbau eines Clusters für Monte-Carlo-Simulationen, Untersuchung der Robustheit unter dem Einfluss nichtlinearer Effekte
- Z24.9 – Implementierung aller Funktionen als effiziente und zuverlässige C-Software

AP1320- FCS-Plattform-Instanz (SW) / Generieren

- Z32.1 – Grobkonzeption FBW-System
- Z32.2 – Automatische SW-Generierung mittels Spezialisierungs-Tool-Suite
- Z32.3 – Manuelle Adaption des modellbasierten PLAMA(S)

AP1330- FCS Integration & Verifikation

- Z33.1 – Integration/Verifikation OS, Driver, PLAMA
- Z33.2 – Integration/Verifikation Laws

AP1340- FCS Flugzeugintegration & PtF

- Z34.1 – Zuarbeit Integration des FBW-Systems in das Validierungsflugzeug und bei der Durchführung der Integrations- und Bodentests

AP1350- Flugtests und Auswertung

- Z35.1 – Erprobung der generellen Funktionen im Reiseflug in großer Höhe
- Z35.2 – Erprobung der ATOL-Funktionen

Die Beschreibung der Ergebnisse im folgenden Kapitel 2.1.2 ist entsprechend der oben aufgeführten Ziele gegliedert.

2.1.2 Ergebnisse der Arbeitspakete

2.1.2.1 AP1110 – System Requirements Capture / Synthesis

Erzeugte Artefakte

- Dokument: YHD FLYSMART (siehe Fa. CAS)



Nicht erreichte Ziele

Alle Ziele erreicht.

Z11.1 – Zuarbeit bei der Erstellung grundlegender Anforderungen

Im Rahmen dieses Arbeitspaketes unterstützte USTUTT die Fa. CAS bei der Definition der grundlegenden Anforderungen an das Validierungsflugzeug aus Sicht der FCP23 Technologie und der Systemfunktion. Hierbei erarbeitete Ergebnisse sind:

- Definition des manuell bzw. automatisch pilotierten FBW-Fluges
- Festlegung der Referenzmissionen des Validierungsflugzeuges
- Interfaces Pilot-System, Flugzeug-System
 - Eingriff System-Flugzeug
 - Für das Validierungsflugzeug bleibt die bestehende mechanische Steuerung erhalten.
 - Das System greift in die mechanische Steuerung ein:
 - Mittels rotatorischer Aktuatoren (primäre Steuerung).
 - Mittels elektrischen Eingriffs in bereits elektrifizierte Steuer-/Stellorgane der mechanischen Steuerung (Flaps, Gear Retraction, Engines).
 - Festlegen von Momenten, Abtriebsgeschwindigkeiten bzw. Typ des elektrischen Interfaces.
 - Interaktion Pilot-System
 - Passivieren des Eingriffes des FBW-Systems in die mechanische Steuerung
 - Laden und Aktivieren von automatisch pilotierten Missionen
 - Überwachung des aktuellen Trimmzustandes
 - Überwachung des aktuellen Systemzustandes (im Sinne Fehlerzustand)

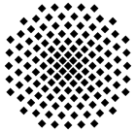
Details zu den Ergebnissen wurden im YHD (High Level Requirements Document) durch die Fa. CAS dokumentiert.

Z11.2 – Definition Sicherheits-, Zulassungs- und Qualifikations-Level

Die maximale Auftretenswahrscheinlichkeit eines Catastrophic Events für die hier betrachtete Flugzeugklassen (EASA CS23/CLASSI..III siehe [27]) darf $10E-6 \dots 10E-9$ pro Flugstunde nicht überschreiten. Für eine zu realisierende Systemfunktion $SFu(n)$ gilt somit für die Auftretenswahrscheinlichkeit der Zustände $zSFu(n)(t)$ der $SFu(n)$, die gemäß [28] als Catastrophic Event klassifiziert werden müssen:

$$P\{UzSFu(n)(1h) \rightarrow CAT\} \leq 10E-6 \dots 10E-9$$

Der für eine Musterzulassung erforderliche Development Assurance Level ist gemäß [28] für CLASSI&II: DAL C und für CLASSIII: DAL B. Informelle Gespräche mit Zulassungsbehörden (zwei Treffen mit Austro Control) indizieren, dass eine Anwendbarkeit der zitierten Norm im Fall von FBW-Systemen genauer geprüft werden muss und eine Anhebung des DAL im Falle CS23/ClassI&II auf Level B nicht ausgeschlossen werden kann. Hierzu sind weiterführende Gespräche mit der EASA innerhalb des Projektes CERTT-FBW23 für den 26.11.2015 in Köln geplant.



Ein wesentlicher Aspekt der Hardwarequalifikation für FBW-Systeme in CS23-Anwendungen sind die Festlegung der EMV- und Blitzschutzlevel. Diese werden insbesondere durch die moderne Leichtbauweise (GFK, CFK) bestimmt. Im Rahmen dieses Arbeitspaketes konnten anhand qualifizierter, sicherheitsrelevanter Cockpit Avionik in CS23-Flugzeugen ähnlicher Bauart die relevante „Waveform“ sowie die Level (Energieeinstrahlung) der notwendigen Blitz- und EMV-Tests auf Modul- und Systemebene festgelegt werden.

Alle Sicherheits-, Zulassungs- und Qualifikations-Level wurden durch die Fa. CAS im YHD dokumentiert.

Z11.3 – Ableiten Technologie-relevanter Anforderungen

Wesentliche Anforderungen an die Technologie zur Realisierung des spezifizierten FBW-Systems umfassen die Bereiche:

1. Ergebnis: Anforderungen getrieben durch Sensor- und Übertragungsheterogenität.
Heterogenität bezeichnet die Verschiedenartigkeit möglicher Sensoren, die mittels PLAMA(C) an eine FCP23-Instanz angebunden werden sollen. Sie bezieht sich nicht nur auf die Art und den Grad der Komplexität des Sensors und der Sensordaten, sondern auch auf den Übertragungsweg vom Sensor zur FCP23-Instanz.
2. Ergebnis: Anforderungen an die Systemfunktionen.
Die Anforderungen an die zu realisierenden Systemfunktionen wurden getrieben durch das Ziel einer hochgradigen Automatisierung der Missionsfähigkeit und die (daraus resultierende) notwendige Automatisierung weitere Systeme (gegenüber Basis SAFAR-EU-FP7).

Um in FLYSMART der Heterogenität der Sensorik (inkl. Übertragungsweg) Rechnung zu tragen, musste PLAMA(C) so überarbeitet werden, dass beliebige proprietäre Protokolle bei zeitgleich beliebiger Fragmentierung der Signale und Stati innerhalb empfangener Frames von der FCP23-Instanz eingelesen werden können.

Die in FLYSMART zu realisierenden Systemfunktionen mussten folgende Anforderungen erfüllen.

- Steuerung des Flugzeuges via 4D-Trajektorienfolge
- Centerline-Folge auf der Piste bei Start und Landung
- Flare zur Realisierung des Abfangbogens bei der Landung
- Decrab bei Landung mit Seitenwindkomponente (bis 5kn)
- Automatisches Bremsen
- Automatische Konfigurationsänderungen (Flaps, GearRetraction, etc.)

2.1.2.2 AP1120 – FBW-Plattforminstanz

Erzeugte Artefakte

- Dokument: Anforderungen an FBW-Plattforminstanz (siehe CAS)

Nicht erreichte Ziele

Alle Ziele erreicht.



Z12.1 – Zuarbeit beim Ableiten grundlegender Anforderungen an das zu realisierende FBW-System

Im Rahmen dieses Arbeitspaketes wurden die Anforderungen an die zu realisierende FCP23-Instanz definiert. Hierbei wurden hauptsächlich folgende Bereiche betrachtet.

1. Anforderungen an das Systemverhalten der FBW-Plattforminstanz.
Dies umfasst im Wesentlichen:
 - a. Definition der Topologie der Plattforminstanz
 - b. Definition der Megaapplikationen (MAPPs)
 - c. Anforderungen an Allokation der MAPPs / Reallokation der MAPPs im Fehlerfall
 - d. Anforderung an die Master / Slave Umschaltung
 - e. Definition der virtuellen Links (inkl. Rekonfiguration im Fehlerfall)
2. Anforderungen an die Integration des Systems in das Validierungsflugzeug.
3. Anforderungen an alle ergänzenden Aggregate.
Hier wurden die Anforderungen an das Verhalten der Aggregate in den zu betrachtenden Betriebsmodi (StandBy, StandBy-Init, Active, Passive) definiert. Darüber hinaus wurden Anforderungen zur Umsetzung des Emergency Disconnects – also einer sicheren Übergabe der Steuerautorität an den Piloten – definiert.

2.1.2.3 AP1130 – Safety-Process

Erzeugte Artefakte

- Konzept: Generischer Ansatz zum automatischen Sicherheitsnachweis
- Dokument: Sicherheitsanalyse des Emergency Disconnects eines vollautoritären Fly-by-Wire Systems für eine Diamond DA42

Nicht erreichte Ziele

Alle Ziele erreicht.

Z13.1 – Grundlagen für einen teilweise generischen Ansatz zum Nachweis der Ausfallsicherheit

Der im Rahmen dieses Arbeitspaketes entwickelte generische Ansatz ermöglicht einen automatischen Nachweis der Ausfallsicherheit von Systemfunktionen, die in einer FCP23-Instanz implementiert sind. Die Umsetzung in Form eines Tools erfolgte in einem anderen Projekt. Mittels des Tools definiert der Systemingenieur anhand eines Metamodells (via graph. Benutzeroberfläche) das zu analysierende System. Der entwickelte Sicherheitsanalyse-Algorithmus ermittelt anhand eines Failure-Propagation Modells die Ausfallsicherheit der Systemfunktion(en) unter Berücksichtigung von Topologie der Plattforminstanz, allokierte Voting/Monitoring Verfahren und Reallozierung der Megaapplikationen im Fehlerfall. Dabei basiert die prinzipielle Struktur des Failure-Propagation Modells auf Fenelon [29]. Abschließend werden automatisch die zugehörigen Minimalschnitte ermittelt. Zur Evaluierung des Algorithmus wurde die Ausfallsicherheit des FBW-Systems EU-FP7-SAFAR automatisch ermittelt und mit einem (früher erstellten) manuellen Sicherheitsnachweis verglichen. Beide Verfahren zeigten dabei eine sehr gute Übereinstimmung.

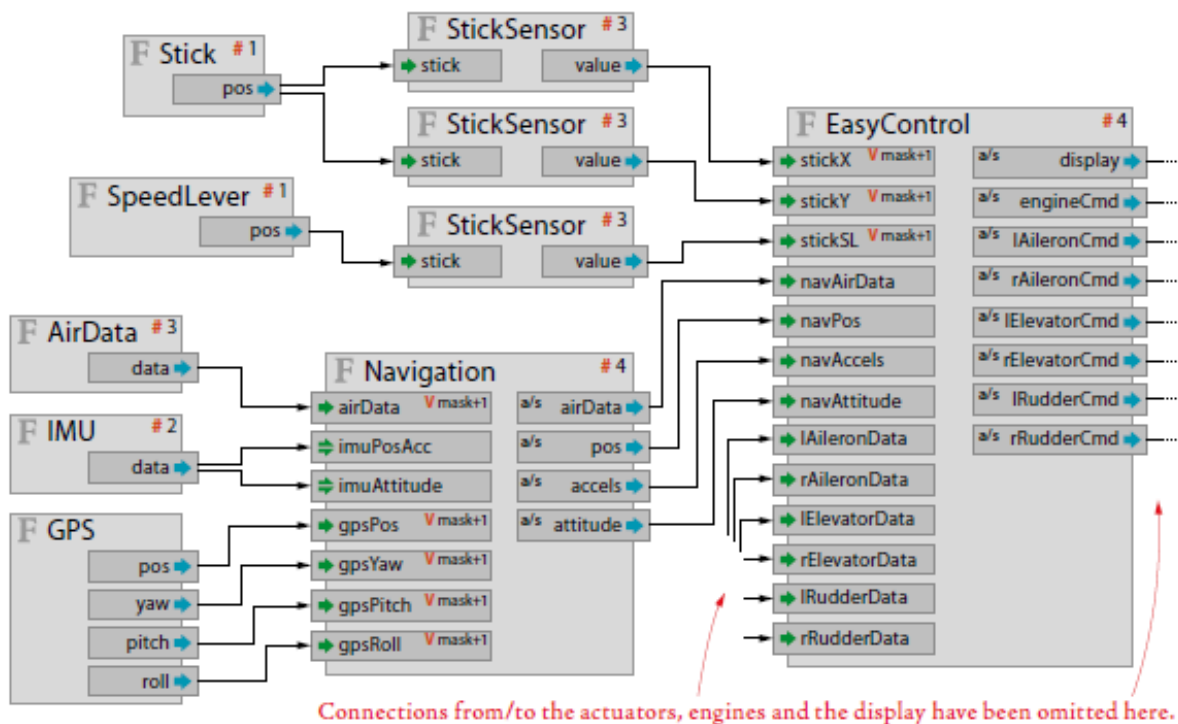
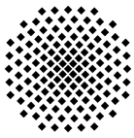


Abbildung 2: Auszug: Failure-Propagation Modell SAFAR-System

System Safety Assessment des Emergency Disconnects

Basis für die Realisierung in FLYSMART ist ein bereits integriertes Fly-by-Wire System aus dem Projekt EU-FP7-SAFAR. Grundlage dieses PtF sind allerdings Rutschkupplungen, die abtriebsseitig der Aktuatorik des Systems angebracht sind und die Steuerautorität des Systems maßgeblich und teilweise nicht deterministisch bzw. schwer reproduzierbar begrenzen. Um die in FLYSMART angestrebte Funktionalität umsetzen und im Flug erfolgreich evaluieren zu können, ist somit neben der Erweiterung des Systems (im Bereich Peripherie, Funktionen, etc.) auch die Entfernung der Rutschkupplungen erforderlich. Ein erneutes PtF soll dabei im Wesentlichen auf einem geeigneten Emergency Disconnect (EmD) basieren. Dieser ermöglicht dem Piloten jederzeit die Übernahme der vollen Steuerautorität mittels der bestehenden mechanischen Steuerung. In einem ersten Schritt gilt es nun zu untersuchen, ob das EmD des bestehenden Systems den spezifizierten Sicherheitsanforderungen in FLYSMART genügt. Hierzu wurde im Rahmen dieses Arbeitspaketes eine detaillierte Sicherheitsanalyse für das EmD des bestehenden Systems durchgeführt. Da das EmD ausschließlich mittels geeigneter Hardwarekomponenten (keine Software) realisiert ist, umfasst das hierzu erforderliche System Safety Assessment eine Failure Mode and Effect Analysis (FMEA) und eine anschließende Fault Tree Analysis. Das Ergebnis zeigte deutlich, dass im Bereich der primären Steuerflächen alle Anforderungen erfüllt wurden und im Bereich der Engines zusätzliche Modifikationen erforderlich waren.

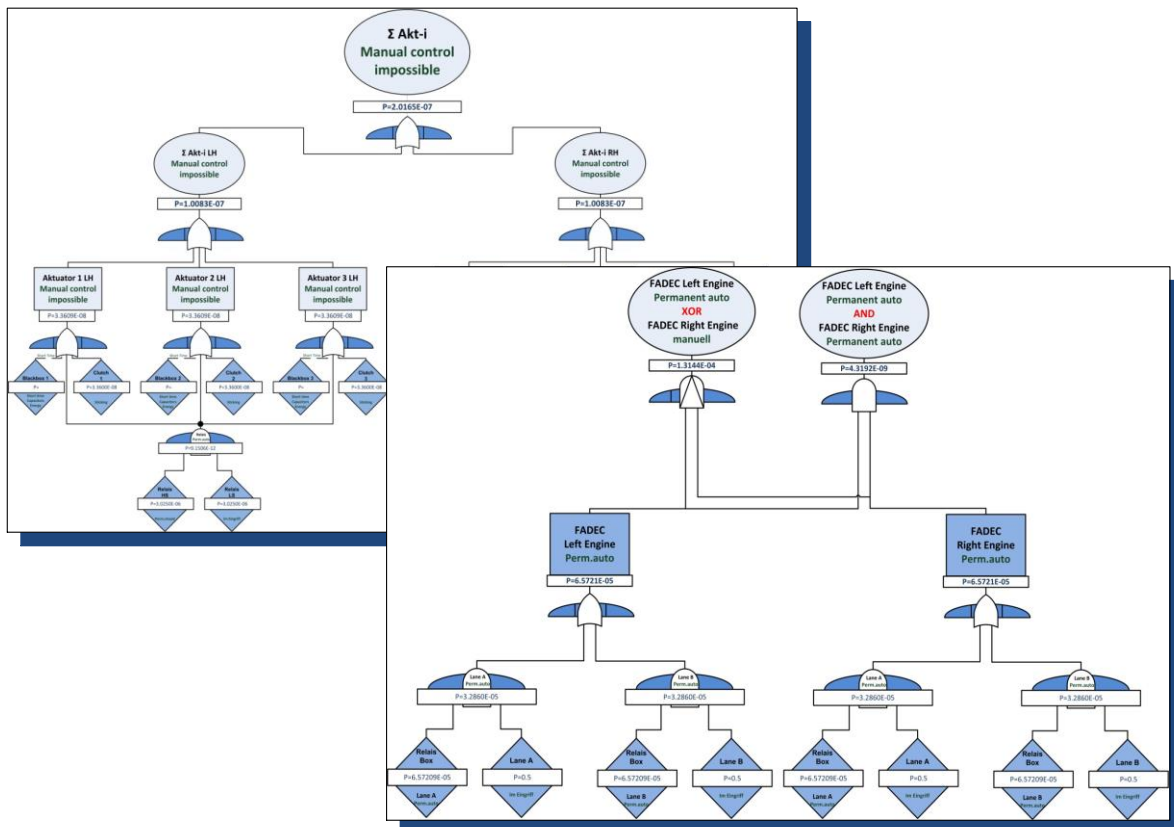


Abbildung 3: Auszug: Fault Tree Analysis des bestehenden Emergency Disconnects

2.1.2.4 AP1210 – Generische Plattform-Middleware

Erzeugte Artefakte

- Modell: Generisches Sensor- und Datentransfermodell
- Software: SRM als Teil des PLAMA(C)
- Konzept: Systematisierung des PLAMA(S) – Ausgangspunkt einer generischen Modellierung

Nicht erreichte Ziele

Alle Ziele erreicht.

Z21.1 – Entwicklung eines generischen, spezialisierbaren Ansatzes für Datenquellen mit Datenübertragung

Zielsetzung des Plattform-Managements allgemein ist es, die Systemfunktionen von allen Aufgaben zu entlasten, die nicht direkt mit ihrer eigentlichen Funktionalität (z. B. Flugregelung) zu tun haben. Speziell das Sensor-Redundanzmanagement (SRM, Teil des PLAMA(C)) leistet dazu einen signifikanten Beitrag, indem es eine Datenebene (API) zur Verfügung stellt, die im Wesentlichen Transparenz in der Art herstellt, dass eine Systemfunktion lediglich ein „virtuelles“, nicht-verteiltes, nicht-redundantes Simplexsystem mit einfachen Eingangssignalen u. QoS wahrnimmt. Geht man davon aus, dass beliebige Sensoren als Quelle für diese Eingangssignale dienen und betrachtet man weiterhin den Datenfluss zwischen Sensor und API der Systemfunktion, so muss Transparenz bzgl. einer Reihe von Einflüssen auf die Eingangssignale hergestellt werden. Dies beginnt mit der Transformation der Sensordaten (Messwerte und Meta-Daten) auf die Eingangssignale. Neben einer Transparenz gegenüber der sensorspezifischen Datendarstellung muss v. a. Transparenz gegenüber



Fehlern hergestellt werden (Fehlertoleranz). Dazu müssen Fehler grundlegend erkannt, zugeordnet und isoliert werden. Hierzu gehört auch, dass zur Fehlererkennung und -tolerierung benötigte Sensorredundanz in „virtuelle“ Simplexsignale aufgelöst wird. Ein weiterer wesentlicher Punkt ist die Transparenz gegenüber der Heterogenität möglicher Sensorik und ihrer Anwendung im System. Diese manifestiert sich zunächst in der Komplexität eines Sensors und der damit verbundenen Vielfalt der mitgesendeten bzw. bei der Übertragung generierten Meta-Daten (Betriebszustand, Fehlermeldung, Status, usw.). Die besondere Herausforderung besteht hier darin, die Menge unterschiedlicher Meta-Daten bzgl. Syntax und Semantik zu vereinheitlichen und in den QoS zu überführen.

Heterogenität bezeichnet die Verschiedenartigkeit möglicher Sensoren, die via (eines bestimmten Teils) des PLAMA(C) an eine FCP23-Instanz angebunden werden sollen. Sie bezieht sich nicht nur auf die Art und den Grad der Komplexität des Sensors und der Sensordaten, sondern auch auf den Übertragungsweg vom Sensor zur FCP23-Instanz. Die Heterogenität der Sensorik lässt sich unter folgenden Aspekten betrachten:

- Sensor Komplexität
- Übertragung zur Plattform-Instanz
- Sensordaten
- Verwendung der Sensordaten

Der erste Aspekt der Heterogenität bezieht sich auf die Komplexität möglicher Sensoren. Ein einfacher Sensor liefert den reinen Messwert ohne zusätzliche Information. Die komplexeste Form eines Sensors sei als Sensor Plattform bezeichnet. Hier sind mehrere komplexe Sensoren integriert. Durch Datenfusion ihrer Größen können neue „virtuelle“ komplexe Sensoren generiert werden. Es ist auch möglich, dass einzelne Messdaten ohne Vorverarbeitung direkt versendet werden. Neben den Stati zu Größen und komplexen Sensoren ist auch ein Status zur gesamten Sensor Plattform zu erwarten. Es ist ebenfalls möglich, dass die Daten der einzelnen komplexen Sensoren weitestgehend unabhängig und ggf. mit unterschiedlicher Aktualisierungsrate versendet werden. Beispielhaft sei hier eine IRU angeführt. Im Rahmen dieses Arbeitspaketes wurde ein generisches Sensor- und Datentransfermodell (umfasst einfachen Sensor bis hin zur Sensor Plattform) entwickelt, dies dient als Grundlage zur Realisierung des Sensorredundanzmanagements als Teil des PLAMA(C).

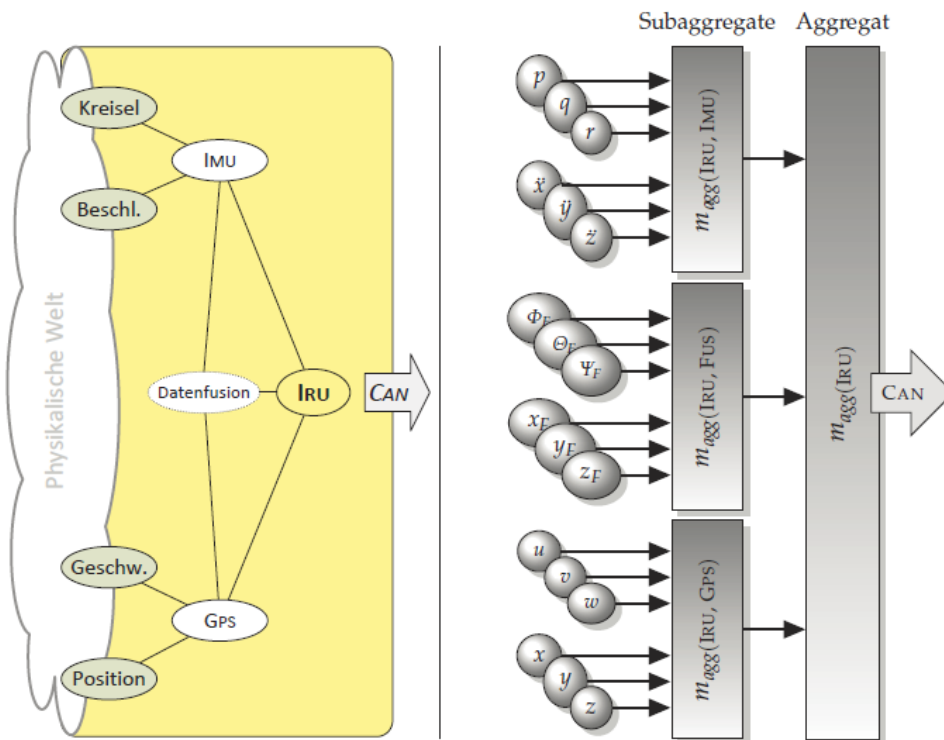
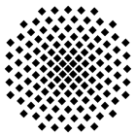


Abbildung 4: Sensor Plattform

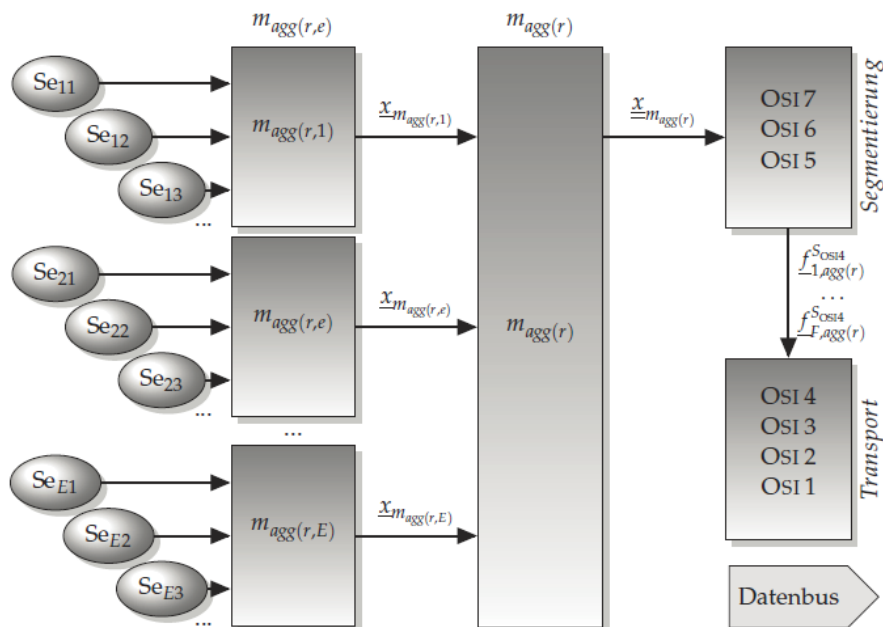
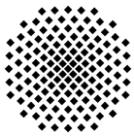


Abbildung 5: Modell Sensor Plattform

Z21.2 – Realisierung des generischen Sensor- und Kommunikationsmanagements als Teil von PLAMA(C)

Auf Basis des generischen Sensor- und Datentransfermodells wurde das SRM als Teil des PLAMA(C) realisiert. Das SRM sollte als schichtenbasierte Middleware ausgelegt werden und folgende Schichten enthalten:



- Schicht S1: Vereinheitlichung der Daten
- Schicht S2: Granularisierung der Daten
- Schicht S3: Replika Fusion

Die Schichten S1 u. S2 bilden dabei das SRM:IN, die Schicht S3 das SRM:X.

SRM: IN

Aufgabe dieses SRM-Teils ist die „Übersetzung“ der „Sprache“ eines beliebigen, sich auf dem Markt befindlichen Luftfahrt-Sensors in die vereinheitlichte „Sprache“ der Signale einer Systemfunktion. Für die „Übersetzung“ sind Kenntnisse notwendig, wo welche Information zu finden ist (Syntax) und wie diese zu interpretieren ist (Semantik). Die „Sprache“ des Sensors ist gegeben durch den Satz an Daten, der insgesamt an der Schnittstelle zur Plattform-Instanz ankommt. Sie ist meist definiert durch das Datenblatt bzw. das ICD der Schnittstelle und / oder des Sensors und definiert nicht nur den Aufbau des einzelnen Datenframes, sondern auch den Inhalt aller gesendeten Datenframes des Sensors. An dieser Schnittstelle finden sich demnach gebündelt alle Informationen über die verschiedenen Ebenen des (komplexen) Sensors hinweg; z. B. Messgrößen, Größen-Status, Gruppen-Status, Sensor-Status, Übertragungs-Status, etc. Die Herausforderung besteht darin, die sensorseitige Vielfalt dieser Informationen zu berücksichtigen und im SRM „übersetzen“ zu können.

Schicht S1 – Vereinheitlichung

Die Hauptaufgabe dieser SRM-Schicht ist zum einen die Rekonstruktion der Sensordaten aus den einzelnen empfangenen Datensegmenten.

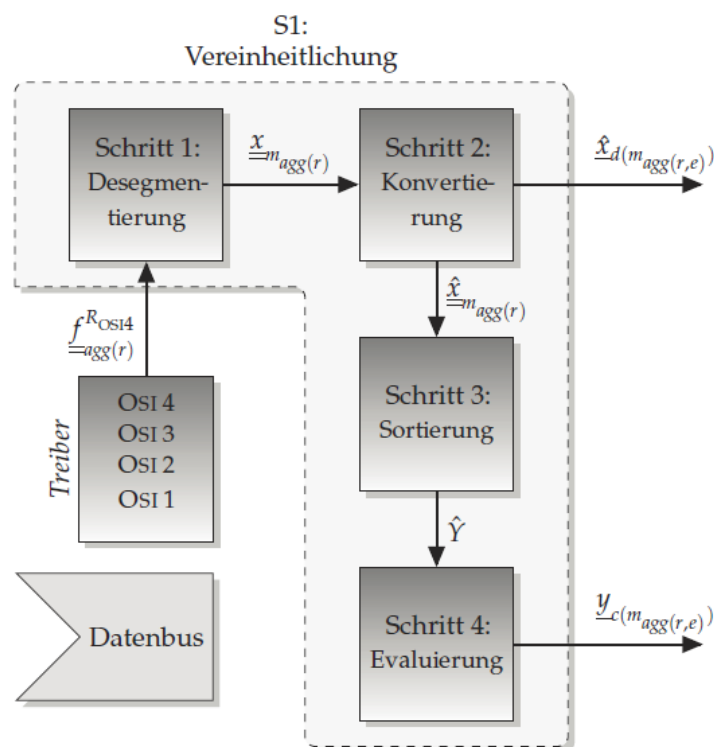
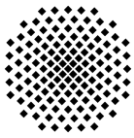


Abbildung 6: SRM:IN: Schicht 1 - Vereinheitlichung



Zum anderen soll bereits eine Auswertung bzw. eine Analyse der Daten durchgeführt und das Ergebnis pro Subaggregat in Form von Status-Indikationen zusammen mit den ggf. angepassten Nutzdaten ausgegeben werden. Schicht S1 umfasst die Schritte Desegmentierung, Konvertierung, Sortierung und Evaluierung.

- Zentrale Aufgabe der Desegmentierung ist das Entpacken aller, vom Treiber zur Verfügung gestellten, empfangen Datensegmente.
- Im Konvertierungsschritt werden die Nutzdaten transformiert. Diese Transformation wird a priori im Systemdesign spezifiziert und beschränkt sich auf eine Anpassung des Datenformats und/oder eine einfache Umrechnung.
- Im Sortierungsschritt werden alle Statusdaten mit gebundener Semantik von Aggregaten, Subaggregaten, Segmenten und Größen gesammelt, nach ihrer Semantik sortiert und direkt in Status-Indikationen umgewandelt.
- Bei der Evaluierung werden alle Indikationen aus dem Sortierungsschritt und dem Treiber zu einem Gesamt-Indikationsvektor pro Subaggregat kombiniert.

Schicht S2 – Granularisierung

Die Granularisierung erfüllt im Wesentlichen Anforderungen bzgl. Neugruppierung in Granulate und Erstellen eines standardisierten Status pro Granulat (QoS).

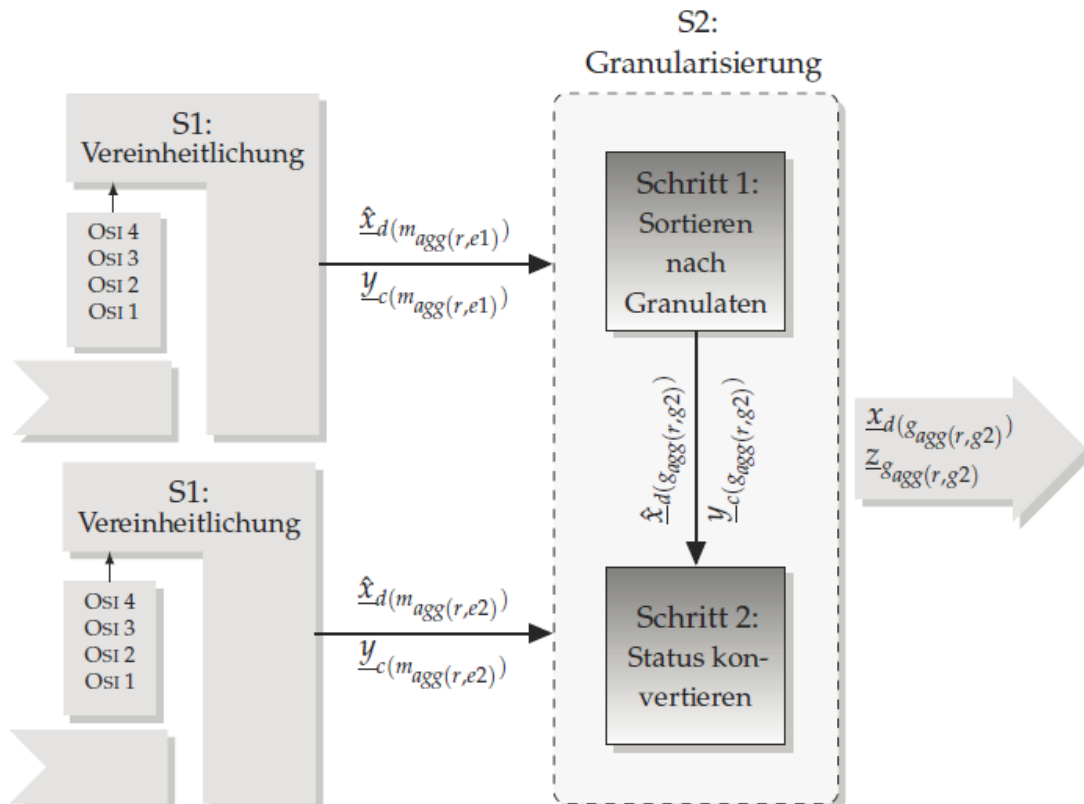
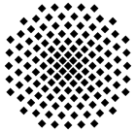


Abbildung 7: SRM:IN: Schicht 2 - Granularisierung



Die Granularisierung ist in zwei Schritte unterteilt:

- der Neuordnung der Daten gemäß der Granulatsgrenzen,
- der Überführung der Status-Indikationen in das Plattform-interne Format (QoS).

Im ersten Schritt (Sortierung) werden zunächst die Einzeldaten der Subaggregate entsprechend der a priori definierten Granulatsgrenzen neu sortiert. Im zweiten Schritt (Status-Konvertierung) wird lediglich der bereits gebildete Indikations-Vektor für das Granulat in einen Status QoS im vereinheitlichten, plattforminternen Format überführt.

SRM:X

Dieser Teil erzeugt im Wesentlichen Fehlertoleranz, indem Daten-Replika aller redundanten Sensor-Quellen zu einem „virtuellen“ Simplex-Sensor mit hoher Verfügbarkeit und Integrität fusioniert werden. Es stehen hier also die Aspekte Fehlertoleranz und Fehlererkennung auf Replika-Ebene im Vordergrund. Die eigentliche Ermittlung der Sensordaten wurde bereits innerhalb SRM:IN erledigt. Folglich kann man für diese SRM Schicht davon ausgehen, dass die Daten der Sensor-Replika schon in Form von Signalen und QoS, gruppiert in Granulaten, vorliegen.

Das SRM:X besteht aus den vier Schritten: Status vorbereiten, Voting, Monitoring und Status bilden.

- Im Schritt Status vorbereiten wird im Wesentlichen für jede Größe des Granulat-Replikas ein Indikationsstapel gebildet. Dieses gibt an, ob die Größe im Voting-Schritt bzw. im Monitoring-Schritt berücksichtigt werden soll. Hiermit wird im Wesentlichen eine notwendige Rekonfiguration des Voting-/Monitoring Mechanismus ermöglicht. Beispiel: Bei Ausfall eines Sensors bei triplex redundanten Sensoren, muss das Voting von Triplex- (für den ursprünglichen Triplex-Sensor) auf Duplex-Konfiguration (für den verbleibenden fehlerfreien Duplex-Sensor) umgeschaltet werden.
- Im Voting-Schritt wird aus den Einzelgrößen der Granulats-Replika ein gemeinsamer Wert ermittelt. Es werden nur Größen mit einer Voting-Indikation berücksichtigt.
- Im Monitoring Schritt werden alle replizierten Einzelgrößen auf Abweichung (Monitoring zum „gebotenen“ Wert des vorangegangenen Schrittes) hin untersucht. Liegt diese Abweichung über einer a priori festgesetzten Grenze, wird eine vorläufige Fehlerindikation für diese Größe ausgegeben. Bei dieser Überprüfung kann ggf. auch noch berücksichtigt werden, dass ein einzelnes Signal nur mit „verringertes Genauigkeit“ vorliegt und entsprechend „weicher“ zu monitoren wäre.
- Im letzten Schritt wird der Status gebildet. Dazu werden zunächst die Fehlerindikationen der Einzelgrößen zusammen mit der Fehlerindikation aus SRM:IN konfirmiert. Damit kann verhindert werden, dass transiente Fehler auf unteren Ebenen direkt zu (unnötigen) Systemreaktionen führen. Die genaue Festlegung der Konfirmierung erfolgt im Systemdesign. Die konfirmierte Fehlerindikation wird dann zusammen mit den bereits vorhandenen Einzelgranulatsdaten zu einem Granulat mit erhöhter Integrität verrechnet.

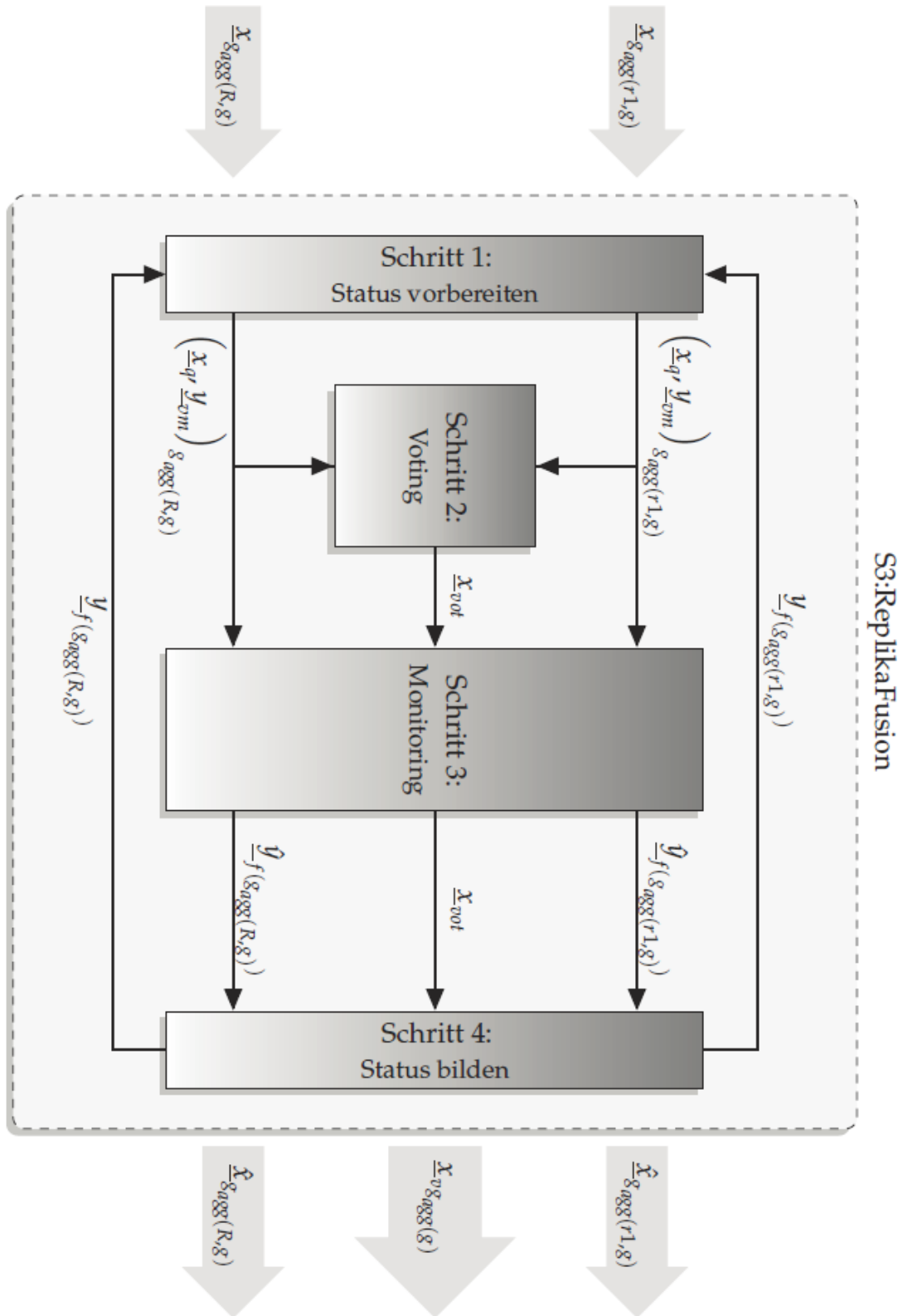
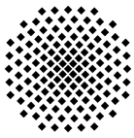


Abbildung 8: SRM:IN: Schicht 3 –Replika Fusion



Z21.3 – Analyse zur Systematisierung von PLAMA(S)

Wesentliches Merkmal des Anwendungsbereichs CS23 ist die Anforderung an eine kosteneffiziente Umsetzung von Systemen. Obwohl in der CS23 überwiegend geringere Anforderungen an die Verfügbarkeit und Integrität eines zu realisierenden Systems als im Bereich CS25 bestehen, kann eine kosteneffiziente Umsetzung von Systemen nur mittels einer möglichst flexiblen Nutzung der Ressourcen (Sensorik, Aktuatorik, Rechnermodule) gewährleistet werden. Im Bereich des PLAMA(S) für CS23 Anwendungen wird dieser Anforderung durch Umsetzung einer dynamischen Rekonfiguration von Mega-Applikationen (zur effizienten Nutzung verfügbarer fehlerfreier Rechnermodule) sowie einer Rekonfiguration der virtuellen Links im Flug bei Auftreten eines Fehlerfalls (zur effizienten Nutzung verfügbarer fehlerfreier Module, Sensorik und Aktuatorik) Rechnung getragen. Dies bedeutet aber, dass das Ressourcenmanagement für CS23-Anwendungen eine höhere Flexibilität als für CS25 Anwendungen aufweisen muss. Respektive ist das PLAMA(S) für CS23-Anwendungen komplexer als für CS25-Anwendungen. Innerhalb des momentan verfügbaren Axx-Prozesses wird das PLAMA(S) für jede neue FCP23-Instanz aus einem bestehenden SCADE-Modell durch manuelle Modellskalierung und manuelle Modelladaption gewonnen

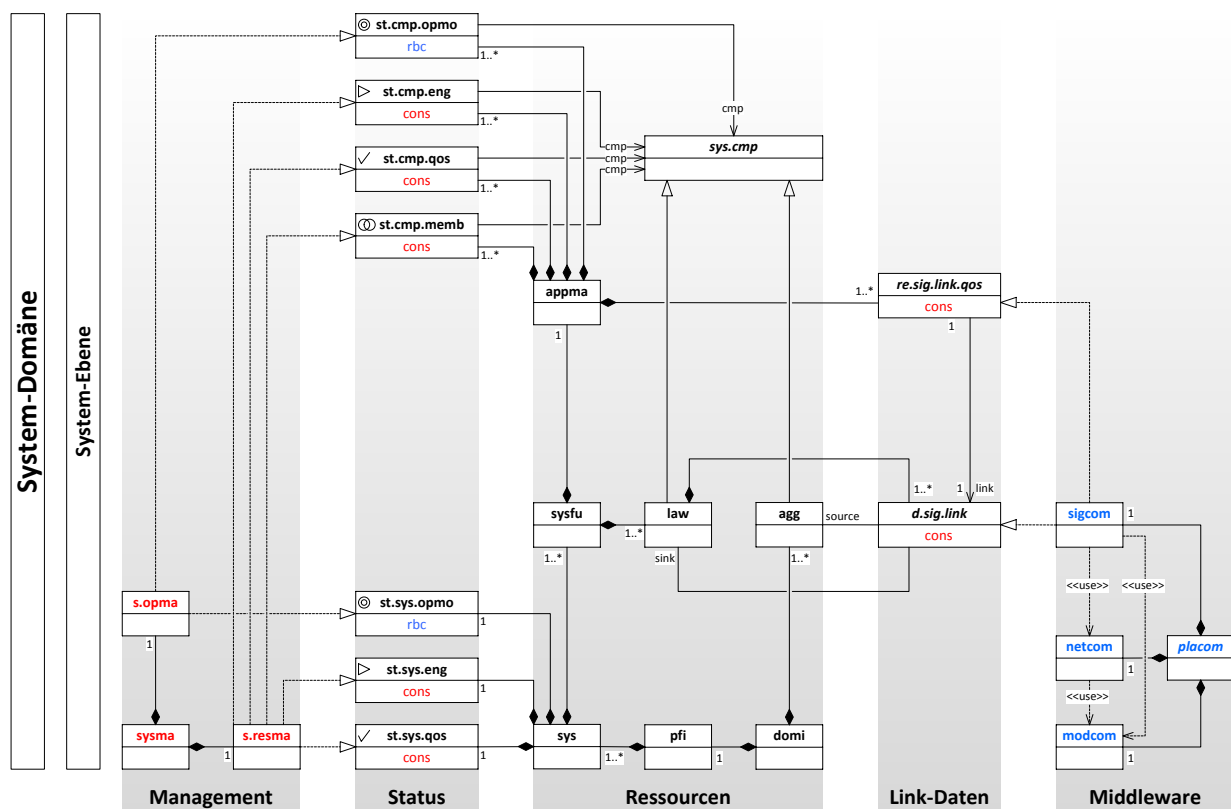


Abbildung 9: Systematisierung Ressourcenmanagement (System-Domäne)

Ergebnis dieses Arbeitspaketes ist die Systematisierung des PLAMA(S). Dies dient in einem beantragten Nachfolgeprojekt (AAA-FCP23) als Grundlage zur generischen Modellierung großer Teile des PLAMA(S). Ziel ist es, zukünftig auch das PLAMA(S) analog zu PLAMA(C) automatisch mittels der Spezialisierungs-Tool-Suite zu generieren.



Zentrale Aspekte zur Systematisierung des PLAMA(S) sind:

1. Domäne-spezifisch Unterteilung des PLAMA(S):
 - a. Ressourcen-Management-System-Domäne: Rekonfiguration der virtuellen Links bei Auftreten eines Fehlerfalls.
 - b. Ressourcen-Management-Plattform-Domäne: Dynamische Rekonfiguration von Mega-Applikationen.
2. Konzeption Schichtenmodell innerhalb der Domänen.
3. Konzeption der generischen (spezialisierbaren) Schritte innerhalb der einzelnen Schichten.

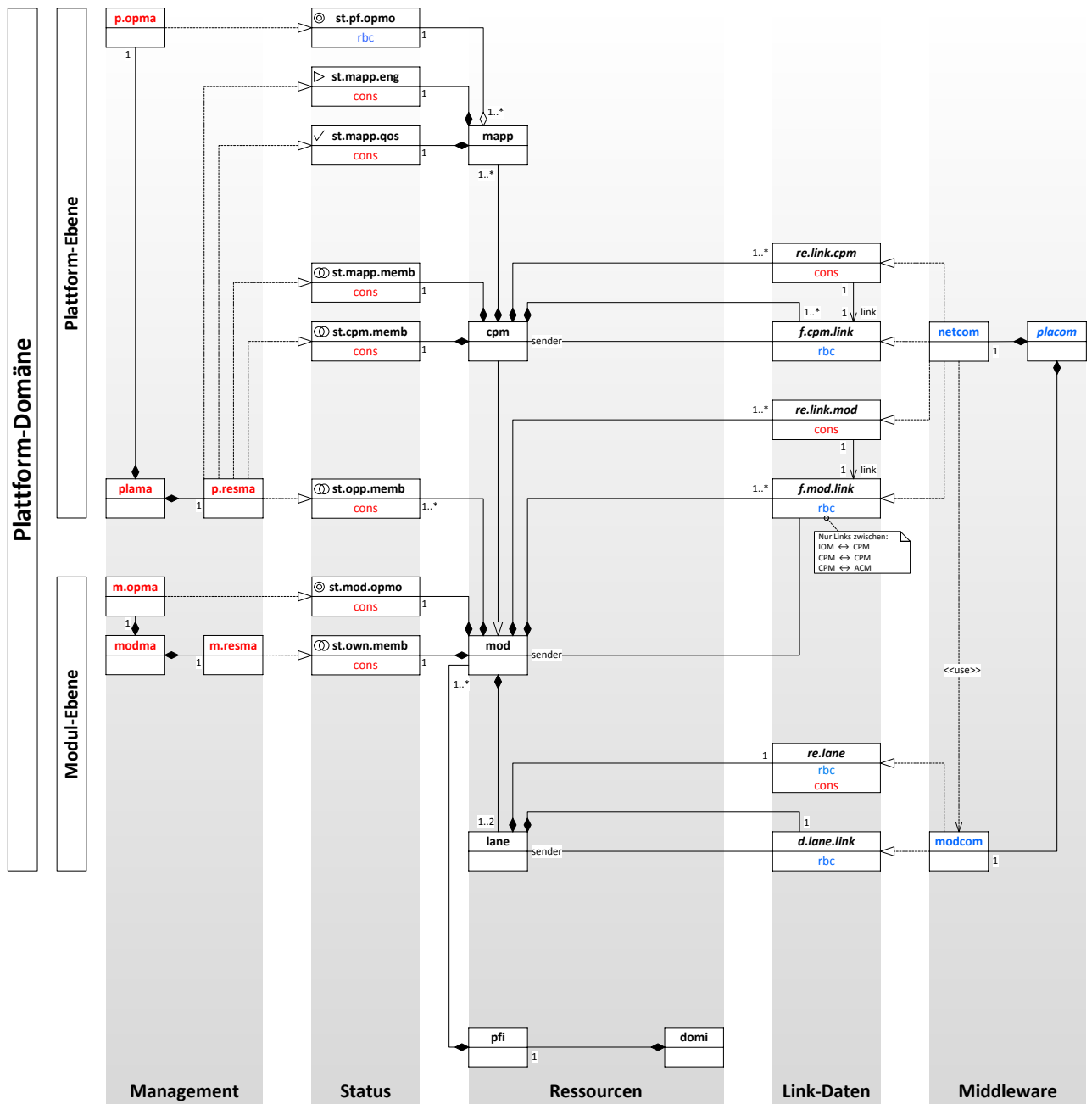


Abbildung 10: Systematisierung Ressourcenmanagement (Plattform-Domäne)



2.1.2.5 AP1220 – Verfahren zur Spezialisierung der generischen Middleware

Erzeugte Artefakte

- Software: GUI Spezialisierungs-Tool-Suite
- Software: Graphische Modellierung von Synthesis-Rules mittels GReAT

Nicht erreichte Ziele

Alle Ziele erreicht.

Z22.1 – CS23-relevante Weiterentwicklung der Spezialisierungs-Tool-Suite

Die Anwendbarkeit der Spezialisierungs-Tool-Suite im industriellen CS23-Entwicklungsumfeld ist maßgeblich durch zwei Aspekte geprägt:

- a) Die „Anwenderfreundlichkeit“ der Eingabeebene zur Festlegung einer neuen FCP23-Instanz.
- b) Die Handhabbarkeit bei der Eingabe neuer Synthesis-Rules.

Aspekt: Eingabeebene CS23-Anwender Spezialisierungs-Tool-Suite

Bei einem potentiellen Anwender aus dem CS23-Umfeld lassen sich nicht ohne weiteres vertiefte Kenntnisse im Bereich der sicherheitsrelevanten Avioniksysteme voraussetzen. Um dennoch eine Anwendbarkeit des Axx-Prozesses in diesem Bereich gewährleisten zu können, mussten folgende Grundsätze berücksichtigt werden.

- a) Die Eingabe (also Festlegung) einer neuen FCP23-Instanz muss auf hoher abstrakter Ebene erfolgen.
- b) Die notwendigen Eingaben sollen graphisch erfolgen.
- c) Die notwendigen Eingaben sollen in Teilbereiche (-probleme) strukturiert werden, deren Verkopplung untereinander erst am Schluss festgelegt werden muss.

Ergebnis der Arbeiten in FLYSMART ist eine graphische Eingabeoberfläche (graphische DSL), in der ein Anwender jede neue FCP23-Instanz in drei Schritten wie folgt festlegt.

- Eingabe der Systemstruktur: Graphische Eingabe (Drag & Drop aus einer Bibliothek) der Module und Peripherie (Sensorik, Aktuatorik) sowie Parametrieren ihrer Interfaces und Festlegen der Granulate der Peripherie.
- Eingabe der Datenflüsse: Graphische Eingabe der Datenflüsse als virtuelle Links, bspw. Datenfluss vom redundant erfassten physikalischen Wert über einen Voting-/Monitoring-Mechanismus zur Systemfunktion.
- Allokieren der Funktionen und Daten: Welcher (physikalische) Wert im Datenfluss kommt von welchem Sensor? Wo wird das Voting/Monitoring ausgeführt? Wo wird die Systemfunktion (Law) ausgeführt?

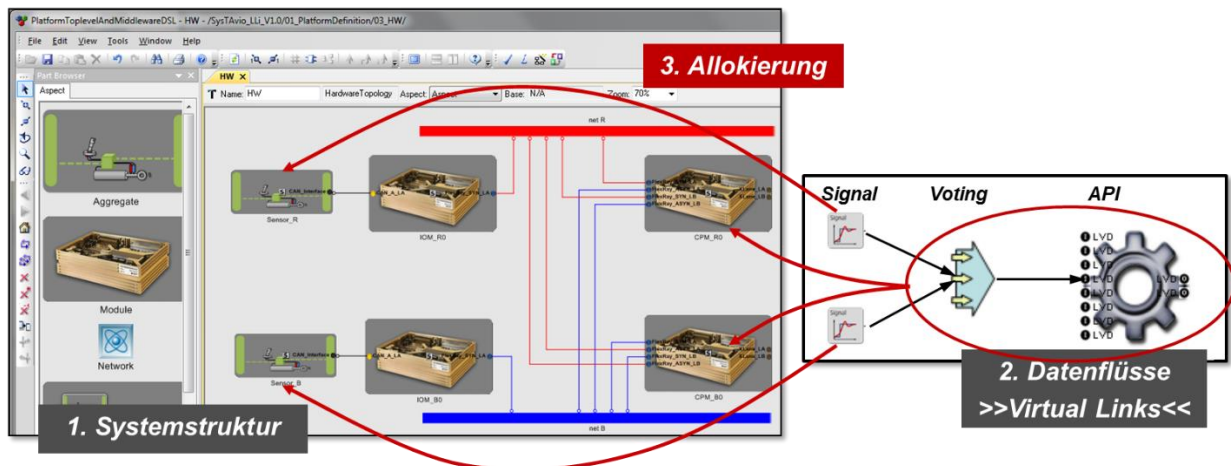
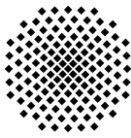


Abbildung 11: Spezialisierungs-Tool-Suite Eingabeebene

Aspekt: Eingabe neuer Synthesis-Rules Spezialisierungs-Tool-Suite

In den ersten Versionen der Spezialisierungs-Tool-Suite mussten die Synthesis-Rules als Python-Code integriert werden. Dies erforderte keine speziellen Vorarbeiten zur Eingabe von neuen Regeln, führte allerdings mit Erreichen eines gewissen Umfangs zu einer komplexen und schwer überschaubaren Repräsentanz innerhalb der Tool-Suite. Um eine Anwendbarkeit (im Sinne der Erweiterbarkeit) im industriellen Umfeld gewährleisten zu können, muss die Tool-Suite an neue Erfordernisse angepasst werden können, d.h. neue Synthesis-Rules müssen effizient integriert werden können. Hierzu wurde dieser Aspekt vollständig überarbeitet. Die Eingabe neuer Regeln erfolgt jetzt mittels graphischer Modellierung unter Verwendung von GREAT, einer Sprache zur Formulierung von regelbasiertem Wissen. Diese Lösung konnte durch Modellierung der Regeln zum überarbeiteten PLAMA(C) im Bereich SRM erfolgreich evaluiert werden.

2.1.2.6 AP1240 – Systemfunktion

Erzeugte Artefakte

- Fahrwerksmodell
- Groundeffektmodell
- Entwurfsmodelle
- Matlab-Skripte für den systematischen Reglerentwurf
- C-Software als Instanz aller entwickelten Funktionen
- Monte-Carlo-Cluster
- Dokumentation, Berichte und Präsentationen
- Abschlussarbeiten

Nicht erreichte Ziele

Alle Ziele erreicht.

Z24.2 – Erweiterung des nichtlinearen, flugmechanischen Modells und Ableitung linearer Entwurfsmodelle

Das von Diamond Aircraft zur Verfügung gestellte Simulink-Modell wurde um eine eigene Darstellung des Bodeneffekts erweitert. Dadurch konnte ein vertieftes Verständnis der nichtlinearen

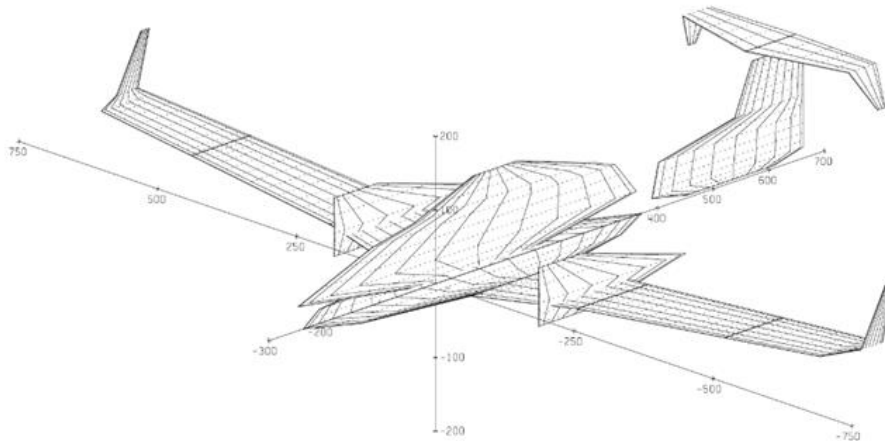


Abbildung 12: AVL-Modellierung DA42, Quelle: [25]

aerodynamischen Effekte gewonnen werden, die im Rahmen des Reglerentwurfs und der Entwicklung einer Monte-Carlo Simulation von großer Bedeutung waren. Dazu wurde das Flugzeug Diamond DA42 im Programm AVL modelliert, vgl. Abbildung 12. Diese Modellierung umfasst eine detaillierte geometrische Beschreibung des Flugzeugs (Profile, Rumpf, Triebwerke, verschiedene Konfigurationen wie Klappen und Fahrwerk) sowie Massen- und Trägheitsverteilung. Hierbei wurden aerodynamische Parameter sowohl für den Flug in großer Höhe, als auch im Bodeneffekt in verschiedenen Höhen erzeugt.

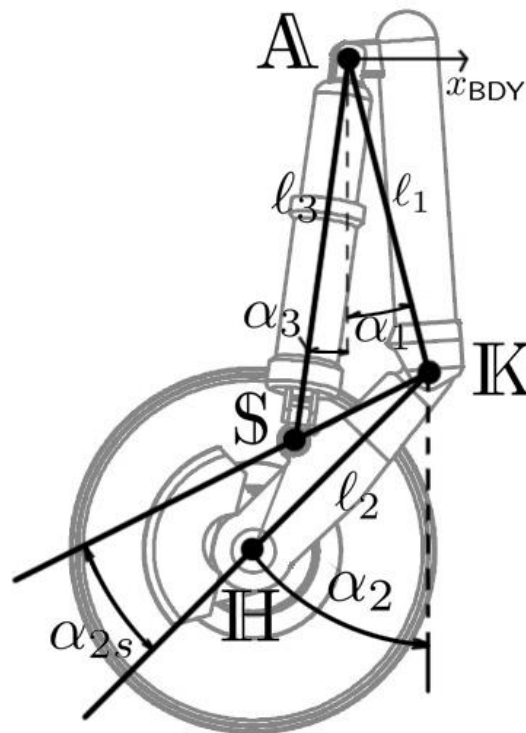


Abbildung 13: Fahrwerksmodell, Quelle: [26]



Für die Entwicklung der ATOL-Funktionen war darüber hinaus eine detaillierte Beschreibung und Parametrisierung des Fahrwerks nötig, vgl. Abbildung 13. Die Beschreibung des Fahrwerks umfasst dabei

- die Kinematik des Fahrwerkbeins modelliert als Feder-Dämpfer-System,
- Radreibung (Slipping), Rollreibung,
- Strukturdynamik der Felge und der Karkasse,
- dynamisches Verhalten des Steuergestänges,
- Bremsen.

Diese Beschreibung ermöglichte einen systematischen Reglerentwurf für das Rollen am Boden.

Neben diesen Effekten wurde eine Wind-Simulation implementiert. Diese besteht aus verschiedenen Komponenten:

- Stationärer Windanteil mit atmosphärischer Grenzschicht,
- Dryden-Wind-Modell zur Beschreibung der höhenabhängigen Turbulenz,
- Diskrete Böen.

Ausgehend von dem so entstandenen nichtlinearen Modell wurden analytisch lineare Modelle für den Reglerentwurf abgeleitet. Dies wurde für verschiedene Flugzustände und Konfigurationen durchgeführt.

Z24.3 – Entwurf eines Missionsmanagements unter Berücksichtigung eines Piloteninterfaces

Die zu fliegenden Manöver werden durch ein speziell entwickeltes Missionsmanagement determiniert, das in Form einer State-Machine aufgesetzt ist, vgl. Abbildung 14. Dabei werden die folgenden Rahmenbedingungen berücksichtigt:

- Piloteneingaben zur Freigabe spezieller Manöver, wie z.B. Einleiten des Starts oder des Landeanflugs,
- Flugzustand,
- Zustand des Flugzeugs und des Systems.

Durch die Beachtung dieser Parameter werden neben der nominalen Missionserfüllung Sicherheitsfunktionen bereitgestellt.

Das Missionsmanagement legt den Ablauf der einzelnen Prozesse und damit den funktionalen Zustand aller Prozesse fest. Darüber hinaus werden Konfigurationsänderungen initiiert.

Zusätzlich zu dieser Steuerung der funktionalen Prozesse übernimmt das Missionsmanagement die Bereitstellung von für die Piloten relevanten Informationen, die z.B. auf einem Display dargestellt werden. Beispiele hierfür sind das momentan ausgeführte Manöver und Sicherheitsfragen/Freigaben.

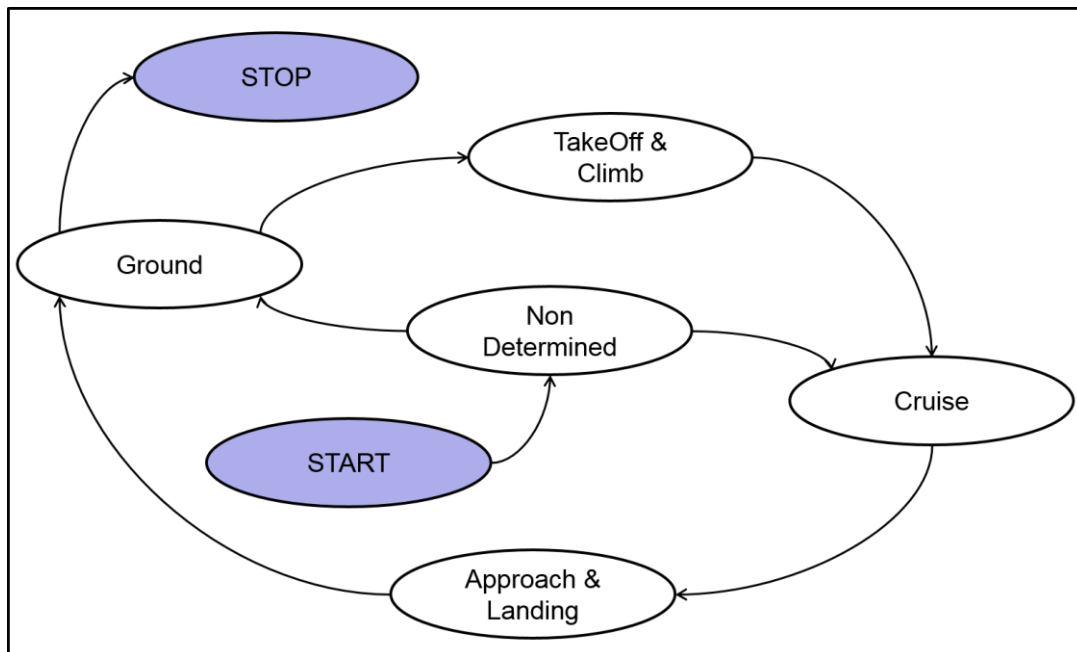
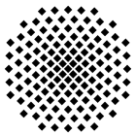


Abbildung 14: State-Machine für das Missionsmanagement

Z24.4 – Entwicklung eines Planers für eine fliegbare ATOL-Referenzmission als Positions-Geschwindigkeitsverlauf (4D)

Ziel war die Entwicklung eines Algorithmus für eine splinebasierte 4D-Solltrajektorienplanung, bestehend aus fliegbaren Positions- und Geschwindigkeitsvorgaben für einen kompletten ATOL-Missionszyklus mit weltweiter Abdeckung.

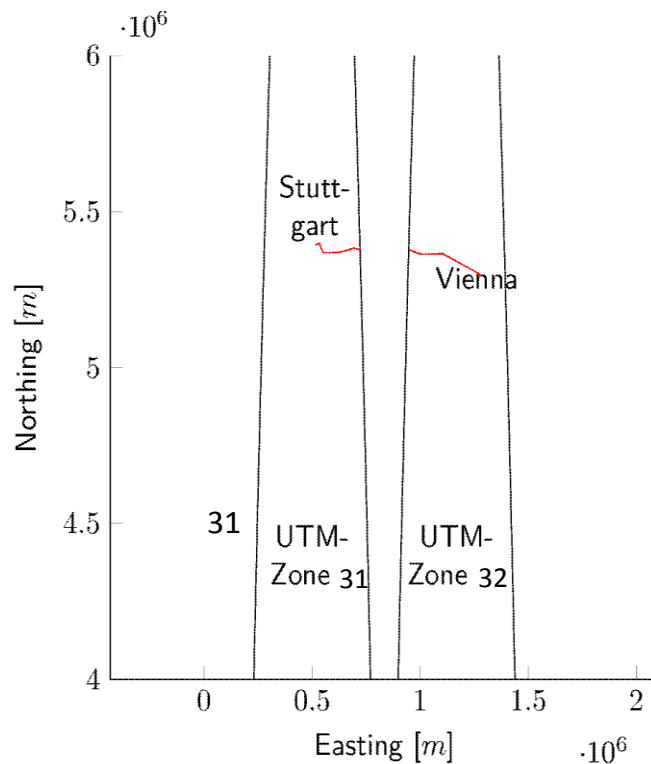
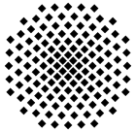


Abbildung 15: Übergang UTM-Zone



Als Eingangsgrößen stehen Wegpunkte zur Verfügung, die entweder von einem Operator oder aus einer vorgelagerten Bewegungsplanung zur Verfügung gestellt werden. Somit ist es möglich, Änderungen an der Trajektorie durch die Piloten im Cockpit vorzunehmen.

Die Trajektorienplanung ist mit Splines dritter Ordnung realisiert, wodurch die Repräsentation und Verarbeitung der Solltrajektorien relativ wenige Ressourcen auf der Bordhardware in Anspruch nehmen. Um die Fliegbarkeit der Referenztrajektorie zu garantieren, werden im Planungsprozess die fahrzeugspezifischen, dynamischen Limitierungen des Flugzeugs berücksichtigt. Beispiele hierfür sind:

- Maximale Steigleistung
- Maximaler Rollwinkel

Um eine Integration in den Luftraum zu vereinfachen, wurden im Planer Pattern mit an Standardprozeduren angelehnten Ein- und Austrittsmustern implementiert.

Die Referenzplanung erfolgt in lokalen, kartesischen Referenzkoordinatensystemen, wodurch eine hohe Planungspräzision erreicht wird. Dies ist im Zusammenhang mit ATOL von besonderer Bedeutung.

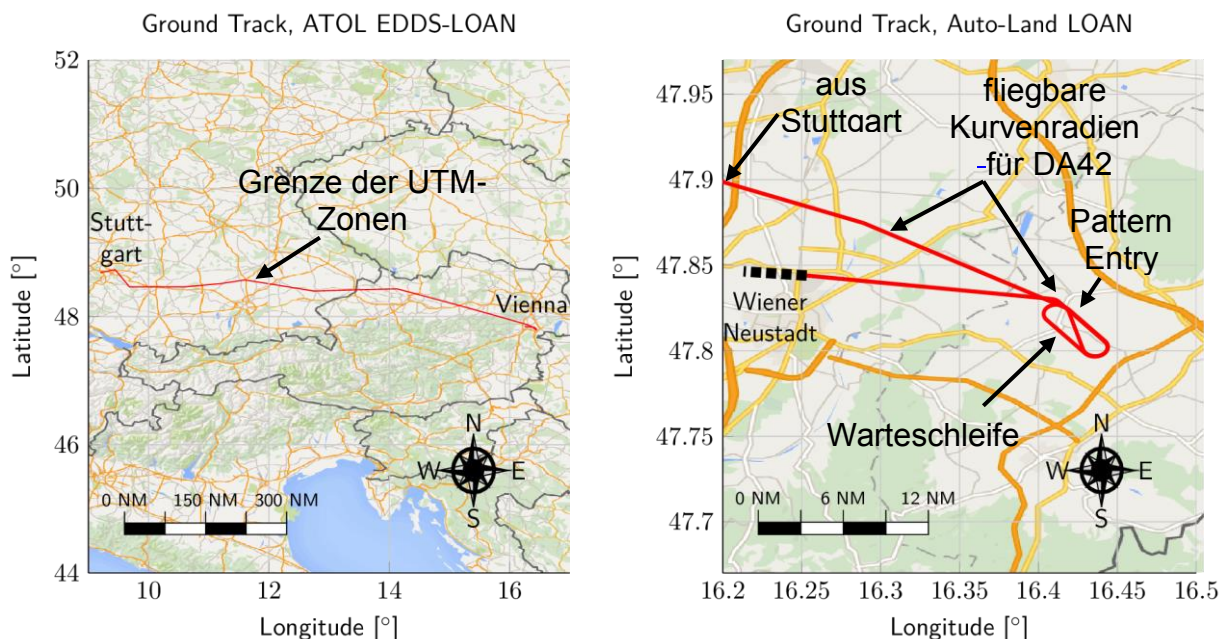
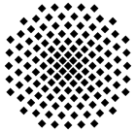


Abbildung 16: Geplante Referenztrajektorie für eine ATOL Mission von Stuttgart (EDDS) nach Wiener Neustadt (LOAN)

Jedes Referenzkoordinatensystem ist dabei einer UTM-Zone zugeordnet. Eine Mission über große Distanzen und damit mehrere Zonen wird daher in mehreren lokalen, kartesischen Koordinatensystemen geplant, vgl. Abbildung 15. Dabei müssen an den Rändern bestimmte Randbedingungen eingehalten werden, um eine konsistente, stetige Gesamtrajektorie zu erhalten. Durch eine anschließende Transformation in das UTM-System ergibt sich eine präzise, global gültige Planung. Durch dieses Vorgehen können ATOL-Missionen auch über große Distanzen hochgenau geplant werden, vgl. Abbildung 16.



Z24.5 – Entwicklung einer Guidancefunktion zur Einregelung der 4D-Trajektorie

Bei der Entwicklung des **4D-Trajektorienfolgereglers** (Position und Geschwindigkeit) waren neben einer sehr hohen Tracking Performance insbesondere im Hinblick auf die Piloten ein komfortables und angenehmes Flugverhalten zentrale Ziele. Auf Grund der ATOL-Funktionalität muss der Regler mit verschiedenen Flugzeugkonfigurationen und in einem weiten Bereich der Flugenveloppe arbeiten und für alle Bedingungen eine hohe Stabilität und Dämpfung insbesondere der schnellen Dynamiken (Anstellwinkelschwingung, Taumelschwingung, Rollbewegung) garantieren. Darüber hinaus muss der Regler gerade angesichts des bemannten Flugs in Bodennähe alle Anforderungen mit sehr hoher Zuverlässigkeit und Robustheit erfüllen.

Um die oben beschriebenen Anforderungen an den Regler möglichst optimal zu erfüllen, wurde ein hierarchischer Aufbau gewählt, vgl. Abbildung 17. Der Path Tracker berechnet aus der aktuellen Position und der Solltrajektorie Ablagen. Die Performance konnte durch den Einsatz einer vorrausschauenden Funktion, welche den zukünftigen Verlauf der Trajektorie einbezieht, erheblich verbessert werden. Die Guidance minimiert die Ablage von der Solltrajektorie und kommandiert Sollwerte für Drehraten und Fluggeschwindigkeit an die Inneren Regelkreise, welche diese mit Hilfe der zur Verfügung stehenden Aktuatorik einregelt. Der FCL Manager determiniert, wie in Z24.3 beschrieben, den Modus und das Manöver, legt darauf basierend den Zustand der Guidance und der Inneren Regelkreise fest und kommandiert Änderungen der Konfiguration. Durch die Entwicklung eines übersichtlichen Designs mit einer möglichst geringen Anzahl an Modi konnte die Implementierung sicher und effizient gestaltet werden und der Aufwand einer möglichen Zertifizierung wird somit reduziert.

Die Minimierung der Trajektorienablage durch die Guidance ist mit kaskadierten Reglern realisiert, welche eine Begrenzung verschiedener Zustände wie Lage, Raten und Geschwindigkeit ermöglichen. Durch den Einsatz von kinematischen Vorsteuerungen können Kopplungen zwischen Bewegungsachsen (z.B. Kurvenkompensation) berücksichtigt werden und die Trackingperformance verbessert werden. Um die negativen Effekte auf die Performance eines im Falle einer Sättigung auftretenden Integratorüberlaufs zu verhindern, sind alle entsprechenden Kreise mit einer Anti-Windup Logik ausgestattet. Die Grundstruktur der Guidance ist für alle Missionsabschnitte, ausgenommen des Flares, annähernd identisch. Für den Flare wurde ein spezielles Regelgesetz entworfen, welches basierend auf der gemessenen Höhe über Grund Vorgaben für die Sinkrate und den Nickwinkel berechnet. Dadurch ergibt sich ein robustes Flaremanöver, welches auch im Falle auftretender Störungen, wie zum Beispiel Böen, zu guten Ergebnissen führt.

Z24.6 – Entwicklung Innerer Regelkreise zur einheitlichen Flugregelung in der gesamten Flugenveloppe und für alle Konfigurationen

Die Inneren Regelkreise regeln die kommandierten Drehraten und Fluggeschwindigkeit mit Hilfe der aerodynamischen Steuerflächen und der Schubhebelstellung ein. Für die Regelung der Drehraten wurden Mehrgrößenregler entworfen, welche neben dem Folgeverhalten auch eine Anpassung der Dämpfung und der Frequenz der schnellen Dynamiken ermöglichen. Weiterhin werden Kopplungen zwischen den Bewegungsachsen (Roll-Gierkopplung) explizit berücksichtigt. Die Verstärkungsfaktoren werden über die Flugenveloppe und die Konfiguration gescheduled, wobei ein transientenfreier Übergang gewährleistet wird. Alle Gainsets der Ratenregler wurden systematisch, modellbasiert



entworfen. Im Falle einer Stellflächen saturation kommt ein neuartiges, optimales Mehrgrößen-Anti-Windup-Verfahren zum Einsatz.

Die Schubregelung besteht aus einem Regler und einer Vorsteuerung. Die Vorsteuerung beruht auf der ressourcenschonenden Invertierung eines relativ detaillierten und genauen Motor-Propeller-Modells, wodurch die Trackingperformance verbessert und Nichtlinearitäten explizit berücksichtigt werden können. Der Regler ist mit einem Frequenzfilter so eingestellt, dass im Normalbetrieb eine geringe Triebwerksaktivität impliziert wird und der Regler im Falle kritischer Böen trotzdem mit der notwendigen Bandbreite reagiert.

Für alle Bodenoperationen während Start und Landung wurde ein Regler entworfen, welcher die Mittellinie mit Hilfe des gekoppelten Bugradausschlags (NWS) und der Seitenruderstellung einregelt. Die durch das Feder-Dämpfersystem im Steuergestänge und der Friktion des Bugrads verursachten Verzögerungen in der Ansteuerung des Bugradausschlags wird durch den Einsatz eines Smith-Prediktors berücksichtigt. Beim Start verursachen die gleichlaufenden Motor-Propeller-Antriebe ein relativ großes Giermoment. Um die notwendige Autorität um die Gierachse sicherzustellen, wird differentieller Schub eingesetzt.

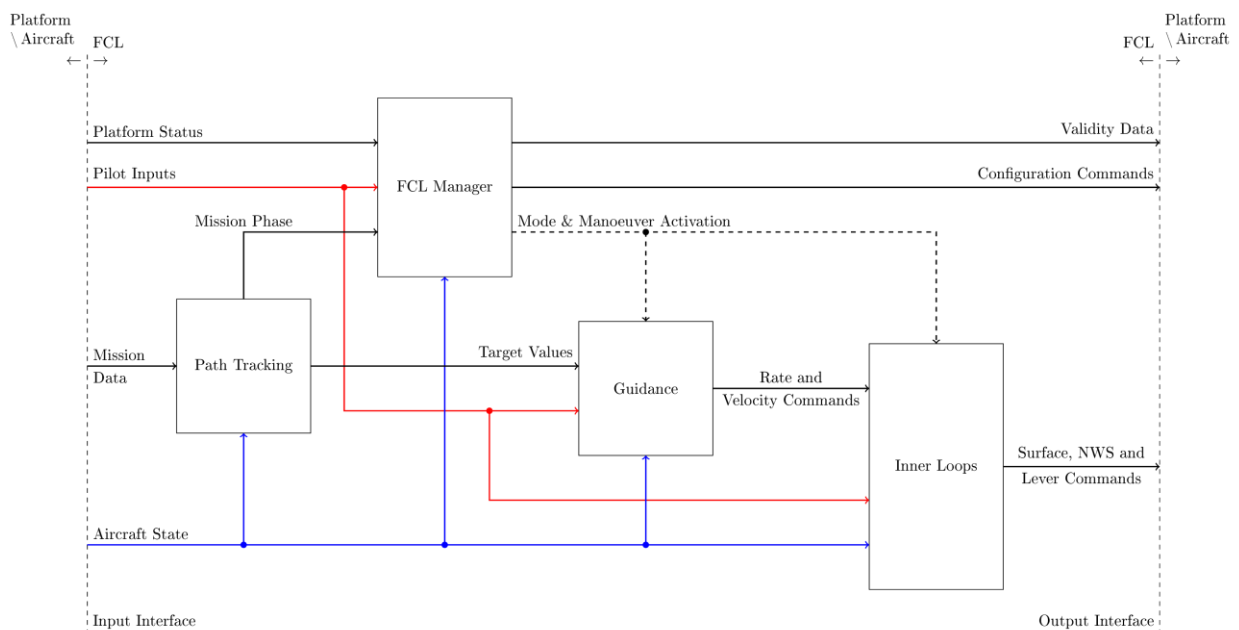


Abbildung 17: Struktur FCLs



Z24.7 – Theoretische Stabilitätsbetrachtungen und Robustheitsanalyse der Regler

Die Stabilität der Regelkreise wurde mit Hilfe der indirekten Lyapunovmethode in der gesamten Flugenveloppe und für alle Konfigurationen (Klappen, Fahrwerk) untersucht, vgl. Abbildung 18.

Durch die verwendeten Entwurfsverfahren sind Phasen- und Amplitudenreserven garantiert.

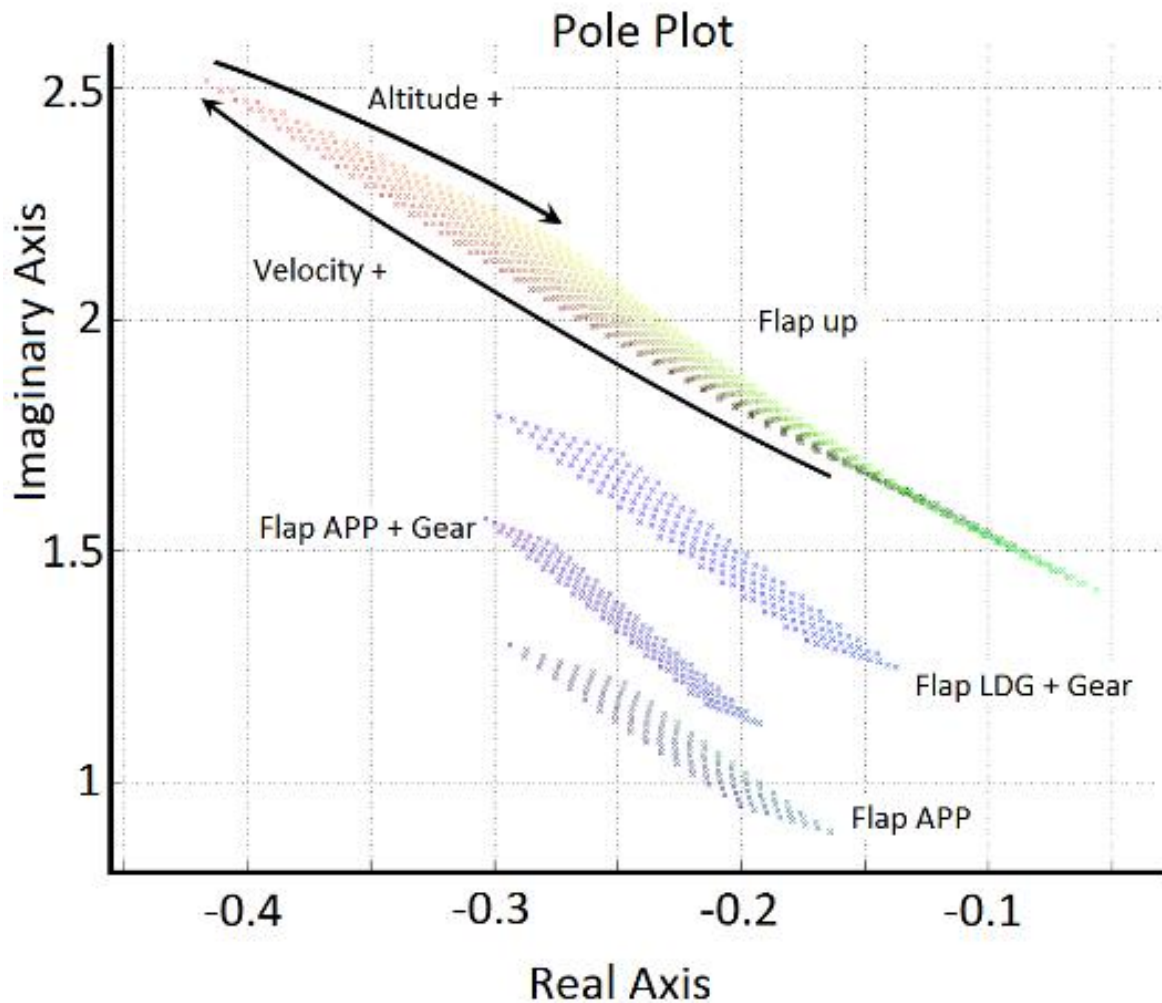


Abbildung 18: Pollagen über Fluggeschwindigkeit und Flughöhe

Z24.8 – Aufbau eines Clusters für Monte-Carlo-Simulationen, Untersuchung der Robustheit unter dem Einfluss nichtlinearer Effekte

Um die Robustheit der Regler unter dem Einfluss von Unsicherheiten zu untersuchen, wurden Monte-Carlo-Simulationen durchgeführt. In einer Vorstudie wurden Unsicherheiten in folgenden Bereichen als kritisch für sichere ATOL-Funktionen identifiziert:

- Flugzeugmasse, Massenträgheiten und Schwerpunktsverschiebungen
- Wirksamkeiten der aerodynamischen Stellflächen
- Windparameter
- Bodeneffektparameter, vgl. Abbildung 20



Im Rahmen des Projekts wurde daher ein PC-Cluster aufgebaut, welcher über Software für Fernwartung und Fernsteuerung betrieben wird, vgl. Abbildung 19. Darauf aufbauend wurde ein skalierbares Framework design und implementiert, welches systematische Monte-Carlo-Simulationen erlaubt. Dieses besteht im Wesentlichen aus vier Funktionssoftwares:

- Job-Generierung: Erstellung von Simulationsaufträgen in zwei Betriebsmodi:
 - Deterministische Variation ausgewählter Parameter in einem Intervall (Permutation)
 - Stochastische Parametervariation
- Job-Verwaltung (Master): Vergabe der Aufträge im Netzwerk zwischen mehreren Teilnehmern
- Job-Ausführung (Slave): Automatische Ausführung der Aufträge und Rückmeldung an Master
- Job-Auswertung: Automatische Auswertung der Simulationsergebnisse



Abbildung 19: iFR Monte-Carlo-Cluster

Für die Auswertung der Simulationen wurden für die General Aviation angepasste, von CS-25 Kriterien abgeleitete, quantitative Erfolgskriterien für den Erfolg einer Landung definiert. Die automatische Auswertung der Ergebnisse konnte darauf basierend kritische Parameter oder Parameterkombinationen identifizieren. Diese Ergebnisse konnten für die systematische Optimierung der Regler verwendet werden. Weiterhin wurden Kriterien zum Landeabbruch im Anflug abgeleitet. In einem weiteren Schritt können diese genutzt werden, um automatisch ein Go-Around Manöver einzuleiten.

Im Rahmen der Monte-Carlo-Studien konnte die robuste und zuverlässige Funktion der Regler im kompletten definierten Betriebsbereich gezeigt werden.

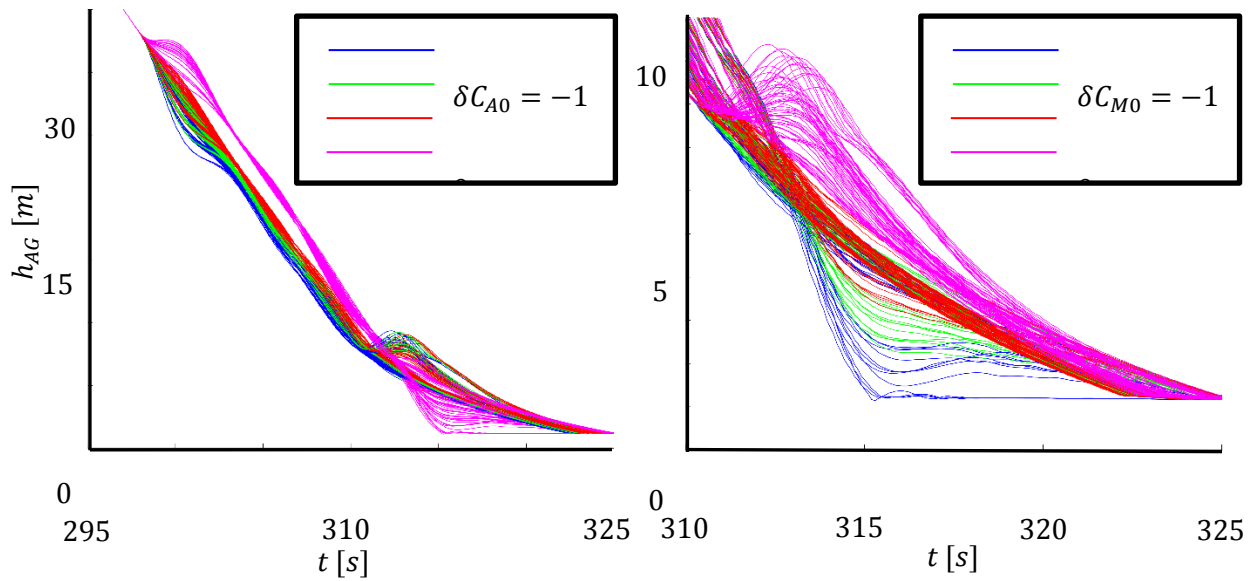


Abbildung 20: Höhe über Grund während Flare, Monte-Carlo-Simulation mit Variation der Bodeneffektparameter

Z24.9 – Implementierung aller Funktionen als effiziente und zuverlässige C-Software

Alle Funktionen wurden als C-Software mit starkem Fokus auf folgende Bereiche implementiert:

- Sicherheit
- Rechenzeit
- Speicherbedarf
- Exakte Kenntnis des Programmablaufs
- Vermeidung von Code-Bibliotheken

Eine besondere Herausforderung stellte dabei die Entwicklung von Cordic-Algorithmen z.B. zur Berechnung trigonometrischer Funktionen dar, vgl. Abbildung 21.

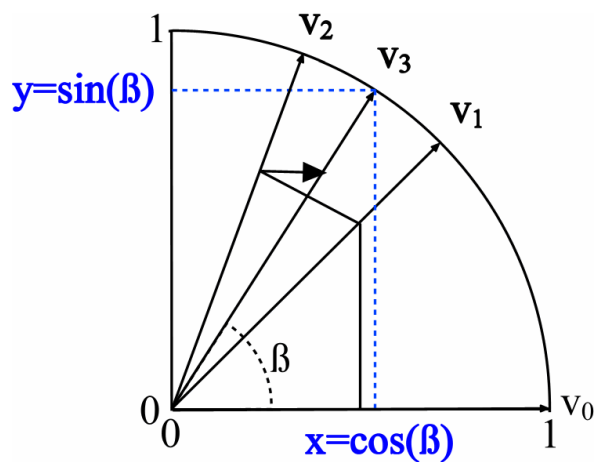


Abbildung 21: Prinzip eines iterativen Cordic-Algorithmus, Quelle: <https://en.wikipedia.org/wiki/CORDIC>



2.1.2.7 AP1320 – FCS-Plattform-Instanz (SW) / Generierung

Erzeugte Artefakte

- Dokument: Konzeption FBW-System
- Software: PLAMA(C) und PLAMA(S) der FCP23-Instanz FLYSMART
- Software: OS und Treiber der FCP23-Instanz FLYSMART
- System: FBW-System FLYSMART

Nicht erreichte Ziele

Alle Ziele erreicht.

Axx-Prozess ist vorhanden und verifiziert.

Z32.1 – Grobkonzeption FBW-System

Basis für die Konzeption des FBW-Systems sind die Anforderungen aus dem YHD. Hierzu erforderlich waren:

- Konzeption aller notwendigen Peripherie (abhängig von am Markt verfügbaren Sensoren und Aktuatoren).
- Konzeption der notwendigen Module.
- Konzeption Systemtopologie und Systemarchitektur.
- Konzeption aller virtuellen Links der FCP23-Instanz anhand der zu realisierenden Systemfunktionen, notwendigen Signale und Sicherheitsanforderungen.
- Allokierung von Systemfunktionen auf Module und Signale auf Peripherie.

Die Konzeption einer konformen Instanz erfolgt manuell und ist eine maßgebliche manuelle Tätigkeit beim Axx-Prozess (weitere manuelle Tätigkeit ist die Adaption des PLAMA(S)). Der Nachweis der Gültigkeit dieser Instanz erfolgt automatisiert (siehe AP1110).

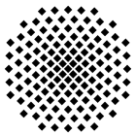
Z32.2 – Automatische SW-Generierung mittels Spezialisierungs-Tool-Suite

Anhand der Konzeption wurde die FCP23-Instanz des zu realisierenden FBW-Systems in die Spezialisierungs-Tool-Suite eingegeben. Dies erfolgte graphisch anhand der neu entwickelten DSL der Eingabeebene (AP1220). Anschließend wurden das PLAMA(C), die Konfiguration der Treiber und die Konfiguration des Betriebssystems automatisch generiert.

Dieser Schritt wurde aufgrund der inkrementellen Erweiterung der FCP23-Instanz mehrfach durchgeführt.

Z32.3 – Manuelle Adaption des modellbasierten PLAMA(S)

Das PLAMA(S) zur Realisierung der FCP23-Instanz FLYSMART basierte auf einem Modell, welches in SCADE implementiert ist. Dieses Modell muss entsprechend den Erfordernissen jeder spezifischen Instanz adaptiert werden. Diese Adaption erfolgt momentan (im aktuellen Axx-Prozess) noch manuell und wurde für FLYSMART auch manuell durchgeführt. Die Analysen zur Systematisierung des PLAMA(S) in AP1210 zeigen, dass große Teile des PLAMA(S) zukünftig auch automatisch generiert werden könnten. Die hierzu erforderlichen Forschungsarbeiten sind Teil des beantragten Projektes AAA-FCP23.



Die manuelle Adaption des PLAMA(S) wurde aufgrund der inkrementellen Erweiterung der FCP23-Instanz mehrfach durchgeführt.

2.1.2.8 AP1330 – FCS Integration & Verifikation

Erzeugte Artefakte

- Iron Bird: Closed Loop / Hardware in the Loop Testumgebung
- Dokument: Nachweis der Betriebssicherheit für ein Fly-by-Wire System einer Diamond DA42 zur Erlangung der Flugfreigabe für die gesamte Flight Envelope unter Permit to Fly

Nicht erreichte Ziele

Alle Ziele erreicht.

Z33.1 – Integration/Verifikation OS, Driver, PLAMA

Zur Verifikation der FCP23-Instanz FLYSMART wurde im Labor ein Iron Bird realisiert. Diese Closed Loop / Hardware in the Loop Testumgebung umfasst:

- Alle Module der FCP23-Instanz
- Echtzeitfähiges Testrig zur Simulation der redundanten Sensorik und Aktuatorik sowie des dynamischen Verhaltens des Validierungsflugzeuges
- Lastsimulation zur Aufprägung aerodynamischer Scharniermomente für real aufgebaute Stellantriebe
- Sichtsystem zur Visualisierung des dynamischen Verhaltens des Validierungsflugzeuges
- VITE-Testtool inklusive SPY zur Verifikation verteilter redundanter Systeme

Innerhalb dieser Umgebung wurden sämtliche Softwarekomponenten der FCP23-Instanz (OS, Treiber, PLAMA) auf die Module integriert. Das korrekte Verhalten der FCP23-Instanz wurde mittels Requirements-basierten Tests gegenüber allen Anforderungen auf Systemebene verifiziert.

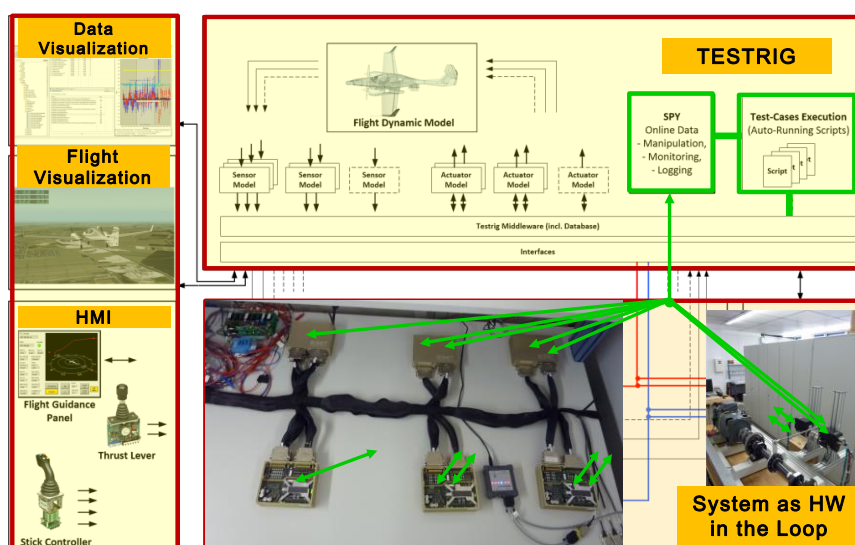


Abbildung 22: Testumgebung Iron Bird USTUTT



Z33.2 – Integration/Verifikation Laws

Bei der Integration der Automatisierungsfunktionen auf die Onboard-Hardware musste eine Anpassung der Schnittstellen vorgenommen werden. Durch Hardware-in-the-Loop-Tests auf dem Ironbird (vgl. Abbildung 23) konnte die Funktion der Software auf dem Bordrechner nachgewiesen werden. Um die Testflüge vorzubereiten, wurden alle geplanten Szenarien mehrfach in umfangreichen Testreihen auf dem Ironbird geflogen. Dadurch konnte auch die Interaktion mit den Piloten untersucht werden. Weiterhin konnten im Rahmen von Aktuator-in-the-Loop-Tests Hardwareeffekte untersucht werden, was zu weiteren Optimierungen der Regler führte.



Abbildung 23: ILS-Ironbird

2.1.2.9 AP1340 – Flugzeugintegration & PtF

Erzeugte Artefakte

- Testumgebung: Umgebung zur Fernwartbarkeit

Nicht erreichte Ziele

Alle Ziele erreicht.

Z34.1 – Zuarbeit Integration des FBW-Systems in das Validierungsflugzeug und bei der Durchführung der Integrations- und Bodentests

Die USTUTT unterstützte die Fa. CAS bei der Integration des FBW-Systems in das Validierungsflugzeug und bei der Durchführung von Tests zur Integration sowie bei Bodentests. Dies erfolgte sowohl teilweise vor Ort in Wiener Neustadt als auch direkt aus Stuttgart mit Hilfe der Realisierung einer Umgebung zur Fernwartbarkeit. Diese Umgebung ermöglichte das Laden neuer Software sowie die Durchführung von Tests direkt auf/am FBW-System, welches im Flugzeug integriert war/ist, via gesicherter Internetverbindung.

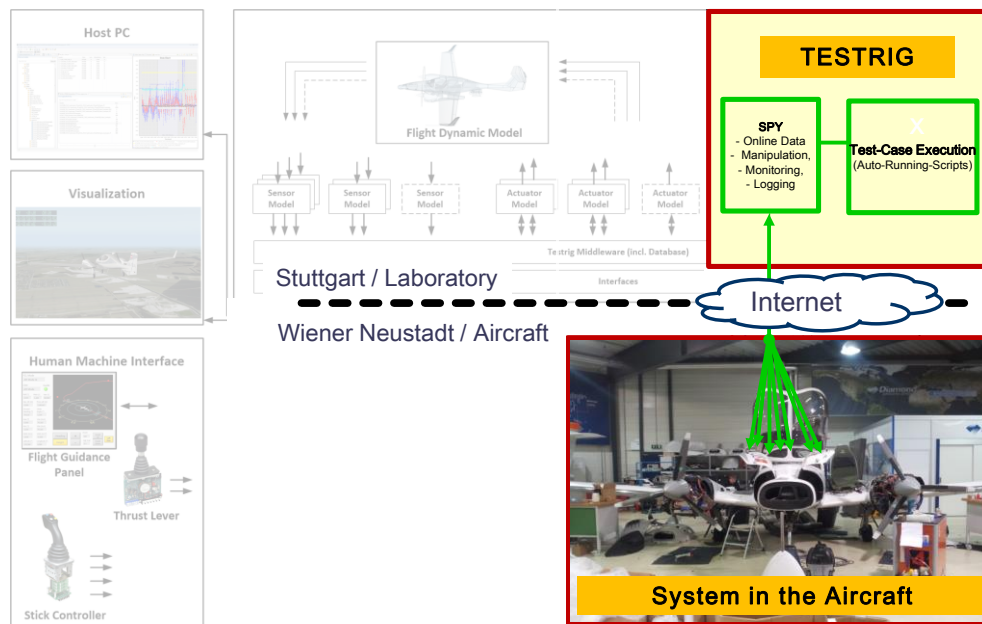
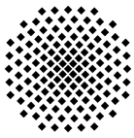


Abbildung 24: Umgebung zur Realisierung der Fernwartbarkeit

2.1.2.10 AP1350 – Flugtests und Auswertung

Erzeugte Artefakte

- Dokumentation, Berichte und Präsentationen
- Videos und Bilder

Nicht erreichte Ziele

Alle Ziele erreicht.

Z35.1 – Erprobung der generellen Funktionen im Reiseflug in großer Höhe

Für die Erprobung der generellen Funktionen (4D-Tracking, Flug in der gesamten Flugenveloppe und mit allen Konfigurationen) wurde ein Flugtestprogramm in Zusammenarbeit mit den Piloten erarbeitet. Zur Vorbereitung der Testflüge wurden Flight Cards mit Hilfe einer im Rahmen des Projekts erzeugten Software erstellt, vgl. Abbildung 26. Um die Piloten im Umgang mit dem FBW-System zu schulen und sie auf dessen voraussichtliches Verhalten vorzubereiten, wurden Briefings vor Ort oder per Telekonferenz durchgeführt.

Die gewonnenen Flugdaten wurden mit speziellem Fokus auf korrekte Funktionalität aller einzelnen Regelkreise (Path Tracker/Guidance, Innere Regelkreise) ausgewertet. Abbildung 25 zeigt einen Flug bei welchem die Guidance nach ca. 50 Sekunden aktiviert wird. Der anfängliche Regelfehler wird schnell abgebaut und die 4D-Referenztrajektorie in der Folge genau eingeregelt. In Kombination mit Pilotenfeedback konnten so gezielte Verbesserungen im Bereich Pilotenkomfort (z.B. Look-Ahead-Funktionen und ruhigere Triebwerksansteuerung) vorgenommen werden, vgl. Z24.7.

Gegen Ende dieser Kampagne wurden als Vorbereitung der automatischen Landung Simulationen des Anflugs in sicherer Höhe durchgeführt. Da alle Testflüge erfolgreich absolviert wurden, wurde anschließend mit den Tests der automatischen Landung begonnen.

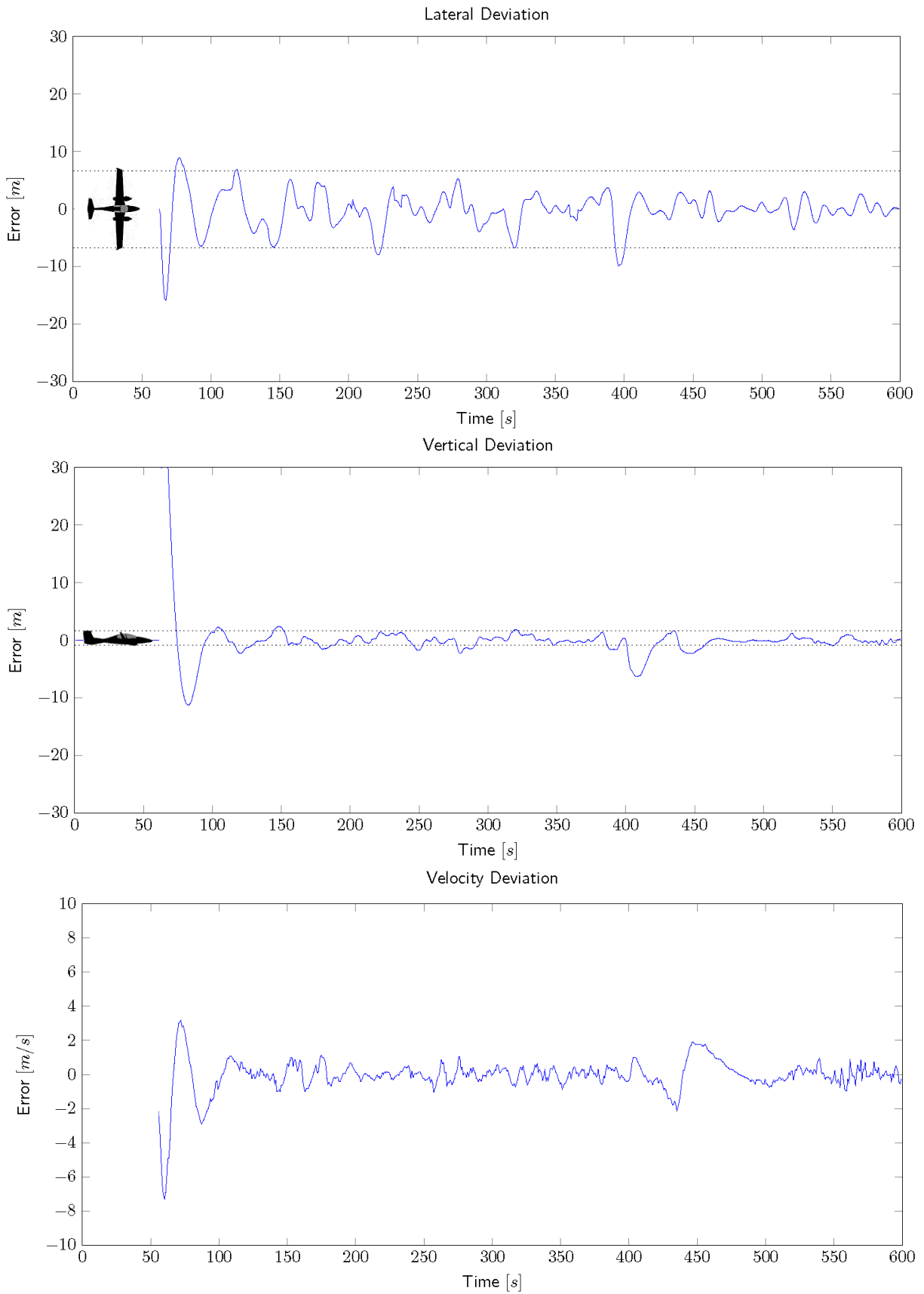
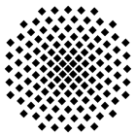


Abbildung 25: 4D Tracking Performance



Z35.2 – Erprobung der ATOL-Funktionen

Wie in der vorangegangenen Kampagne wurde ein Testprogramm definiert, Flight Cards erstellt und die Flüge mit Pilotenbriefings vorbereitet.

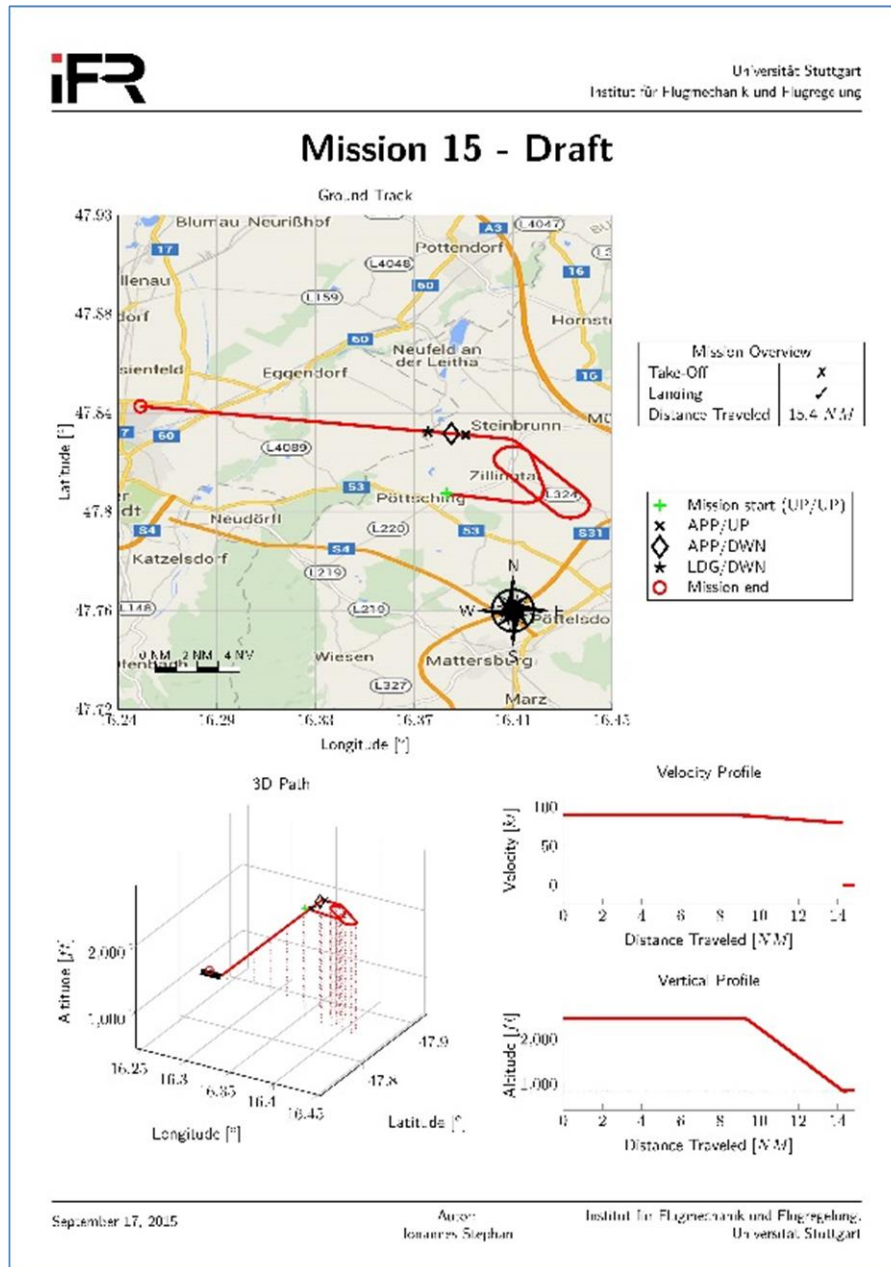


Abbildung 26: Beispiel Flight Card

Zu Beginn wurde die Steuerbarkeit am Boden mit Ground-Steeringtests untersucht. In der Auswertung dieser Versuche wurden Probleme bei der Ansteuerung des Bugrads festgestellt. Durch die relativ geringe Steifigkeit des Steuergestänges kommt es zu Verzögerungen in der Verstellung des Lenkwinkels. Als weitere Herausforderung für die Regelungsfunktionen haben sich die Asymmetrien der Triebwerke erwiesen, vgl. Z24.7. Diese Probleme konnte durch eine explizite Berücksichtigung in der Struktur des Bodenreglers behoben werden.



Am 26. August 2015 wurde die erste erfolgreiche vollautomatische Landung auf dem Werksflughafen der Firma Diamond in Wiener Neustadt durchgeführt, vgl. Abbildung 27. Zum jetzigen Zeitpunkt wurden neun automatische Landungen und zwei automatische Starts demonstriert, wobei es bisher zu keinen Abbrüchen kam. Dabei wurde auch eine komplette ATOL-Mission geflogen, bei welcher die Piloten nur zu Sicherheitszwecken an Bord waren.

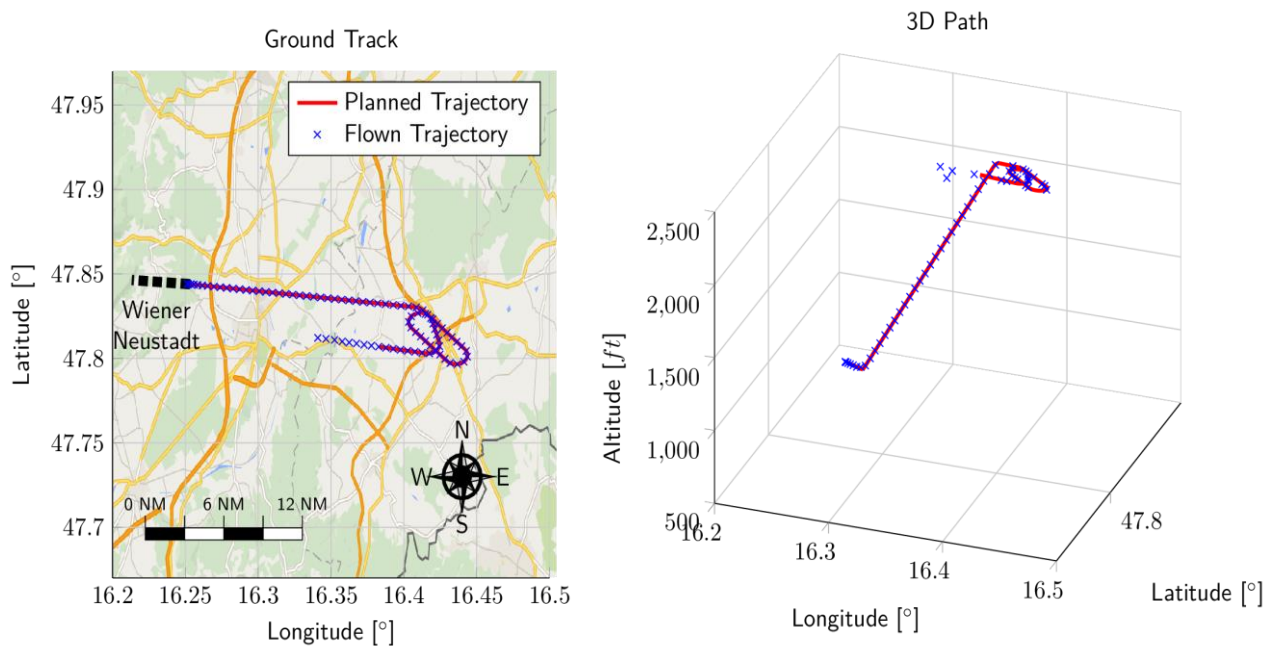


Abbildung 27: Geflogene Trajektorie

2.2 Zusammenfassung des zahlenmäßigen Nachweises

Die entstandenen Gesamtausgaben lagen ca. 1% unterhalb des Finanzierungsplans des Projektantrags. Bei den Einzelposten 0812, 0817 und 0846 kam es zu Abweichungen gegenüber dem Zuwendungsbescheid, die über alle Positionen hinweg kompensiert werden konnten.

- Aufgrund personeller Engpässe bei Beschäftigten 0812 (Beschäftigte E13) mussten Arbeiten zur Implementierung und Verifikation der Software innerhalb der Arbeitspakete AP1210 (*Generische Plattform-Middleware*) und AP1220 (*Verfahren zur Spezialisierung der generischen Plattform*) mit Beschäftigten 0817 (Beschäftigte E1-E11) durchgeführt werden. Dies führte zu einem erhöhten Mittelbedarf in Position 0817 bei zeitgleicher Verminderung des Mittelbedarfs in Position 0812 gegenüber dem Zuwendungsbescheid.
- Aufgrund erhöhter Aufwände bei der Unterstützung der Umrüstung des Validierungsflugzeuges Diamond DA42 in Wiener Neustadt (Austria) sowie der Erlangung der Flugfreigabe beim assoziierten Partner Diamond Aircraft Industries GmbH entstanden im Projekt erhöhte Ausgaben in Position 0846 (Dienstreisen). Diese konnten über Einsparungen in Position 0812 (Beschäftigte E13) kompensiert werden.



2.3 Notwendigkeit und Angemessenheit der Arbeit

Der Verlauf der Arbeit im Projekt entsprach nicht exakt der im Projektantrag formulierten Planung. Aufgrund außerordentlicher Arbeiten innerhalb eines Zertifizierungsprojektes des assoziierten Partners Diamond Aircraft Industries verzögerte sich die Flugzeugintegration, Flugfreigabe (PtF) sowie die Flugerprobung. Mit der kostenneutralen Verlängerung (angepasste Zeitplanung siehe Kapitel 1.3) konnte das Projekt erfolgreich durchgeführt werden. Alle im Arbeitsplan formulierten Ziele wurden erreicht.

2.4 Nutzen und Verwertbarkeit

2.4.1 Wirtschaftlicher Nutzen und Verwertbarkeit

Die Entwicklung (inkl. Qualifikation) komplexer Avioniksysteme bzw. Fly-by-Wire-Systeme ist sehr aufwändig. Die AAA-Plattformtechnologie hat das Potential in sich, diesen Entwicklungsaufwand deutlich zu reduzieren. Damit eröffnen sich für Industriepartner

1. unmittelbar die Möglichkeit der Entwicklung von Fly-by-Wire-Systemen oder anderer komplexer Systeme in einem neuen Umfeld, nämlich CS23. Dies könnte ab ca. 2018 der Fall sein.
2. durch die Übertragung auf CS25/27/29 bzw. die grundsätzliche Übertragung auf IMA² deutliche technologische Vorteile. So geht der AAA-FCP23-Ansatz weit über IMA2G und auch über IMA3G hinaus.

Mit den Ergebnissen dieses Teilvorhabens steht dem industriellen CS23-Entwicklungsumfeld der Axx-Prozess zur effizienten Realisierung solcher Systeme zur Verfügung. In anschließenden Projekten wird diese Technologie weiterentwickelt, um, über die automatische Erzeugung des Plattformmanagements hinaus, auch zugehörige Prozessdokumente und die Testcases automatisiert erzeugen zu können (AAA-Prozess). Darüber hinaus zeigen die Ergebnisse zu AP1210, dass neben dem PLAMA(C) auch große Teile des PLAMA(S) automatisch generiert werden könnten. Dies würde nochmals zu einer deutlichen Steigerung der Effizienz bei der Umsetzung von Systemen auf Basis des Axx-Prozesses führen. Die notwendigen Forschungsarbeiten hierzu sind Teil des beantragten Projektes AAA-FCP23 (siehe Wissenschaftliche Anschlussfähigkeit).

2.4.2 Wissenschaftlich / technischer Nutzen und Verwertbarkeit

Die im Rahmen dieses Vorhabens erarbeiteten neuen Kenntnisse werden in Form von Veröffentlichungen in international anerkannten wissenschaftlichen Medien bzw. Fachzeitschriften (Journals) sowie in Form promotions-/habilitationswürdiger Veröffentlichungen verwertet.

Erkenntnisse aus dem Projekt flossen auch in die Lehrtätigkeit ein. 21 studentische Arbeiten wurden im Zuge des Projekts ausgearbeitet. Darüber hinaus konnten mehrere Studenten als studentische Hilfskraft Erfahrungen mit der AAA-Plattformtechnologie sammeln. Außerdem ist die Fertigstellung sechs themennaher Dissertationen geplant.

² Es erscheint hier wesentlich effizienter, eine derart komplexe Technik wie die von AAA-FCP23 zunächst in einem einfacheren Umfeld zu entwickeln und erst danach in das prozesstechnisch wesentlich komplexere IMA-Umfeld zu übertragen. Aktuelle Arbeiten in diesem Zusammenhang finden im Rahmen des LUFO IV/4-Projekts SysTAvio statt oder sollen in dem parallel beantragten LUFO V/2-Projekt AVATAR ausgeführt werden.



iFR:

- Verwendung der entworfenen Automatisierungsfunktionen in der Vorlesung „Flugregelungsentwurf“
- Geplant sind 2 Dissertation
- 11 Bachelor/Masterarbeiten
- 4 studentische Hilfskräfte

2.4.3 Wissenschaftliche und wirtschaftliche Anschlussfähigkeit

Aspekt AAA-Prozess

Ergebnis des Projektes FLYSMART ist ein Axx-Prozess der im industriellen CS23-Entwicklungsumfeld effizient eingesetzt werden kann. Grundlegende Analysen zeigten, dass das bisher noch manuell zu adaptierende PLAMA(S) analog zum PLAMA(C) in großen Teilen automatisch generiert werden könnte. Dies würde nochmals zu einer deutlichen Steigerung der Effizienz des Axx-Prozesses bei der Realisierung neuer FCP23-Instanzen führen. Zur Komplettierung des AAA-Prozesses soll der generische Ansatz des PLAMA(C) und PLAMA(S) auf die automatische Generierung der Prozessdokumente und Testcases übertragen werden. Diese Themen sowie die Übertragbarkeit der AAA-Plattformtechnologie auf IMA sind Gegenstand der folgenden genehmigten bzw. beantragten Forschungsprojekte.

SeSyC (LUFO V/1) / Verbundführer Airbus Helicopters Deutschland

Im Rahmen des LUFO-Projekts SeSyC werden/wurden folgende Punkte bearbeitet:

- Ermittlung des grundlegenden Verfahrens zur Autodokumentation (xAx-Prozess) auf YRD-, SRD-, SDD-Ebene für den Bereich PLAMA(C).
- Validierung am Sensor-Redundanz-Management.

CERTT-FBW23 (LUFO V/1) / Verbundführer CAS

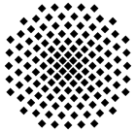
Im Rahmen des LUFO-Projekts CERTT-FBW23 werden/wurden folgende Punkte bearbeitet:

- Ermittlung des grundlegenden Verfahrens zur automatischen Testcase- und Test-Procedure-Erstellung (xxA-Prozess) auf YRD-, SRD-, SDD-Ebene für den Bereich PLAMA(C).
- Validierung am Sensor-Redundanz-Management.

AAA-FCP23 (LUFO V/2) / Verbundführer SET

AAA-FCP23, ein beantragtes LUFO V/2-Programm, umfasst:

- Überarbeitung des PLAMA(S) zur automatischen Generierung im Axx-Prozess.
- Komplettierung des AAA-Prozesses: Entwicklung der Verfahren zur Autodokumentation sowie zur automatischen Testcase- und Test-Procedure-Erstellung für alle Bereiche PLAMA.
- Gesamtvalidierung des AAA-Prozesses anhand eines repräsentativen Fly-by-Wire Systems für CS23-Flugzeuge. Die Validierung erfolgt unter Berücksichtigung aller, zusammengeführter Ergebnisse (auch aus vorangegangenen Projekten).



AVATAR (LUFO V/2) / Verbundführer DAS

In AVATAR, einem beantragten und positiv beschiedenen LUFO V/2-Programm, geht es darum

- a) den Ansatz der Flexiblen Plattform dahingehend weiterzuentwickeln, dass das PLAMA nicht mehr als Middleware-Layer und somit Plattform-Instanz umfassend realisiert ist, sondern von den einzelnen Laws (Applikationen) selektiv genutzt werden kann. Der Hintergrund dafür ist eine potentielle Erweiterung des Ansatzes der Flexiblen Plattform auf IMA, welches keine entsprechende Middleware „kennt“.
- b) den in den LUFO V/1-Projekten SeSyC und CERTT-FBW23 erarbeiteten xAA-Prozess-Ansatz grundsätzlich auf die IMA-Gegebenheiten zu erweitern.

Aspekt: Automatisierte Missionsfähigkeit

Alle im Rahmen des Projekts FLYSMART benötigten Automatisierungsfunktionen für automatischen Start, automatischen Flug und automatische Landung wurden entwickelt und erfolgreich im Flug demonstriert. Im Rahmen von Folgearbeiten sind verschiedene Weiterentwicklungen sinnvoll und nötig:

Robustifizierung und Erweiterung der Funktionen bzgl. äußerer Störungen und Unsicherheiten (Wind, Qualität der Landebahn, Verkehrsteilnehmer,...)

Robustifizierung und Erweiterung der Funktionen bzgl. Störungen und Unsicherheiten am Flugzeug (Sensorrauschen, Masse und Schwerpunktsverschiebung, Ausfall Aktuatorik,...)

Evaluierung verschiedener Interfaces zwischen Pilot und Flugzeug (Art der Pilotenvorgaben, Visualisierungen,...)

2.5 Fortschritt anderer Stellen

Trotz regelmäßiger Recherche wurde zur Laufzeit des Projekts kein wesentlicher Fortschritt anderer Stellen bekannt.

2.6 Veröffentlichungen

Die Projektergebnisse des Vorhabens werden durch folgende Beiträge auf nationalen und internationalen Konferenzen veröffentlicht:

- Geplante Veröffentlichung, Journal: Systematisierung PLAMA(S) und formaler Nachweis der Mechanismen des PLAMA(S)
- sechs Papers bei der AIAA GNC Conference bzw. bei DGLR Veranstaltungen. Themen sind:
 - Übersichtspaper Flugregelungsalgorithmen für automatisches ATOL
 - Pfadplaner mit Splines in vier Dimensionen
 - Innere Regelkreise als Mehrgrößenregelkreise und Mehrgrößen-Antiwindup-Funktion
 - Systematische Monte-Carlo Simulationen als Vorbereitung für CS23 Zulassung in Anlehnung an CS25 Kriterien
 - Ground controller design
 - Flare controller design

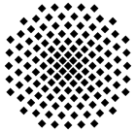


Anlagen

- Erfolgskontrollbericht
- Berichtsbeiblatt / Document Control Sheet

Referenzen

- [1] G. Bartley und B. Lingberg, „Certification concerns of Integrated Modular Avionics (IMA) systems,“ in *27th Digital Avionics Systems Conference*, 2008.
- [2] V. Hadzilacos und S. Toueg, „A Modular Approach to Fault-Tolerant Broadcasts and Related Problems,“ Ithaca, NY, USA, 1994.
- [3] J. Lewis und L. Rierson, „Certification concerns with integrated modular avionics (IMA) projects,“ in *22nd Digital Avionics Systems Conference*, 2003.
- [4] J. Rushby, „Modular Certification,“ SRI International, Menlo Park, CA, 2002.
- [5] C. Watkins, „Integrated Modular Avionics: Managing the Allocation of Shared Intersystem Resources,“ in *25th Digital Avionics System Conference*, 2006.
- [6] C. Watkins und R. Walter, „Transitioning from federated avionics architecture to Integrated Modular Avionics,“ in *26th Digital Avionics Systems Conference*, 2007.
- [7] A. Wilson und T. Preyssler, „Incremental certification and Integrated Modular Avionics,“ in *27th Digital Avionics Systems Conference*, 2008.
- [8] S. Görke, R. Reichel und S. Hesse, „A Generic Platform for Safety-Critical Applications in the General Aviation,“ in *IEEE Aerospace*, Montana, 2011.
- [9] S. Hesse, R. Reichel und S. Görke, „Platform based Realization of Safety Relevant Systems,“ in *3rd CEAS Air&Space Conference / 21st AIDAA Congress*, Venedig, 2011.
- [10] S. Korn, R. Riebeling, S. Görke und R. Reichel, „Flexible Platform Approach for CS27/29 Fly-By-Wire Systems,“ in *EU-Rotorcraft Forum*, Moskau, 2013.
- [11] R. Reichel und M. Armbruster, „X-by-Wire Platform - Concept and Design,“ *Automatisierungstechnik* 59 (9), p. 583–597, 2011.



- [12] R. Reichel und S. Görke, „A generic X-by-Wire Platform,“ in *Fachtagung Mechatronik*, Dresden, 2011.
- [13] R. Reichel, S. Görke, F. Cake, S. Polenz und R. Riebeling, „Flexible Avionics Platform,“ in *Deutscher Luft- und Raumfahrtkongress*, Bremen, 2011.
- [14] J.-B. Itier, „A380 Integrated Modular Avionics - The history, objectives and challenges of the deployment of IMA on A380,“ 2007.
- [15] M. Fervel, A. Lecanu, A. Maussion und J.-J. Aubert, „Aircraft Control Systems With Integrated Modular Architecture,“ Patent US 2012/0109424 A1, 03 05 2012.
- [16] H. Chang, „Surviving the SOC revolution: A guide to platform-based design,“ Boston: Kluwer Academic, 1999.
- [17] D. Densmore, A. Davare, „A platform-based design methodology for the electronic system level: Frameworks, designs flows, and case studies,“ Saarbrücken: VDM Verlag, 2008.
- [18] A. Sangiovanni-Vincentelli, A. Pinto, „Embedded systems design: A platform-based approach to system-level design,“ San Fransico, Oxford: Morgan Kaufmann, 2009.
- [19] M. Di Natale, A. Sangiovanni-Vincentelli, „Moving From Federated to Integrated Architectures in Automotive: The Role of Standards, Methods and Tools,“ in *Proceedings of the IEEE, vol 98, no.5*, p. 603-620, 2010.
- [20] AUTOSAR GbR, „AUTOSAR R3.0: Technical Overview,“ 2008.
- [21] Schmitt, Lorenz and Fichter, Walter, „Observability Criteria and Null-Measurement Kalman Filter for Vision-Aided Inertial Navigation,“ *Journal of Guidance, Control, and Dynamics*. Accepted for Publication June 2, 2015.
- [22] Weimer, F., Frangenberg, M., Fichter, W., "Pipelined Particle Filter with Nonobservability Measure for Attitude and Velocity Estimation," *Journal of Guidance, Control, and Dynamics*, (2014). Accessed April 14, 2014, siehe auch [doi: 10.2514/1.G000238](https://doi.org/10.2514/1.G000238).
- [23] Joos, A., Fichter, W., „Parallel Implementation of Constrained Nonlinear Model Predictive Controller for an FPGA-based Onboard Flight Computer, *Advances in Aerospace Guidance, Navigation and Control*, Springer Verlag, Berlin, 2011, pp. 273-286, ISBN 978-3-642-19816-8.



- [24] Gros M., Niendorf M., Schöttl A., Fichter, W., "Motion Planning for a Fixed-Wing MAV Incorporating Closed-Loop Dynamics Motion Primitives and Safety Maneuvers," *Advances in Aerospace Guidance, Navigation and Control*, Springer Verlag, Berlin, 2011, pp. 247-260, ISBN 978-3642198168.

- [25] Kühnert, T., „Bestimmung flugmechanischer Derivative in Bodennähe und Robustheitsanalyse eine automatische Landesystems“, Masterarbeit iFR Universität Stuttgart, Stuttgart 2015

- [26] Meschini, P., "Diamond DA-42 Physical Model of the Landing Gear for Real-Time simulation," Masterarbeit University of Glasgow, Stuttgart 2014

- [27] European Aviation Safety Agency, "CS-23 Certification Specifications for normal, utility, aerobatic and commuter category aeroplanes," 2003.

- [28] Federal Aviation Administration, "Advisory Circular 23.1309-1C," 1999.

- [29] Peter Fenelon, John A. McDermid, M. Nicolson, David J. Pumfrey, "Towards Integrated Safety Analysis and Design," in *Applied Computing Review*

Berichtsblatt

1. ISBN oder ISSN geplant	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht
3. Titel FLYSMART Leitprojekt FBW23 – FBW-Plattformtechnologie	
4. Autor(en) [Name(n), Vorname(n)] Reichel, Reinhard Fichter, Walter	5. Abschlussdatum des Vorhabens Juni 2015
	6. Veröffentlichungsdatum Oktober 2015
	7. Form der Publikation Technischer Bericht
8. Durchführende Institution(en) (Name, Adresse) Universität Stuttgart, Institut für Luftfahrtsysteme (ILS) Pfaffenwaldring 27 70569 Stuttgart	9. Ber. Nr. Durchführende Institution -
	10. Förderkennzeichen 20V1108N
	11. Seitenzahl 50
12. Fördernde Institution (Name, Adresse) Bundesministerium für Wirtschaft und Energie (BMWi) 53107 Bonn	13. Literaturangaben 29
	14. Tabellen 1
	15. Abbildungen 27
16. Zusätzliche Angaben -	
17. Vorgelegt bei (Titel, Ort, Datum) -	
18. Kurzfassung Die Entwicklung und Zertifikation von Fly-by-Wire-Systemen für automatisierte Arbeitsflugzeuge und intelligente Transportflugzeuge der Klassen EASA CS23/Class...III ist eine Herausforderung und nur über grundsätzlich neue Techniken möglich. Die AAA-Plattformtechnologie kann als ein solcher Ansatz betrachtet werden. Diese Technologie zeichnet sich durch einen streng layerhaften Aufbau der Gesamt-Software mit dem Plattform-Management als zentralem Layer aus. Damit ist für Laws die Systemkomplexität mit Redundanz und Fehlertoleranz nicht mehr sichtbar. Sie können „simplex minded“ entwickelt werden. Der AAA-Prozess ermöglicht es, für ein neu zu entwickelndes Fly-by-Wire-System das Plattform-Management weitgehend automatisch zu erzeugen, zu dokumentieren und zu verifizieren. Damit lassen sich derartige Systeme in Zukunft äußerst effizient entwickeln und qualifizieren. Zentrale Aufgabenbereiche dieses Teilvorhabens waren die Weiterentwicklung und Validierung des Axx-Prozesses zur weitgehend automatischen Erzeugung des Plattformmanagements im Hinblick auf einen Einsatz im industriellen CS23-Umfeld sowie die Entwicklung und Validierung der Funktionen (Laws) zur Erweiterung der automatisierten Missionsfähigkeit (inkl. ATOL) für zukünftige Arbeits- und Transportflugzeuge EASA CS23. Die Umsetzbarkeit des Axx-Prozesses ist durch zahlreiche Forschungsprojekte erbracht. Seine Flexibilität ist durch Demonstratoren im CS23-, Hubschrauber- und Automotivbereich grundsätzlich verifiziert. Dennoch war noch ein deutlicher Forschungsschritt erforderlich, um diese Technologie in effizienter Weise im industriellen CS23-Umfeld zum Einsatz zu bringen. Auch auf funktioneller Seite mussten grundsätzliche Lücken geschlossen werden. Dies betraf die Automatisierung möglichst vieler Flugzeugsysteme (Systemtransparenz für den Piloten) sowie die grundsätzlich vorhandene Automatic Take Off and Land (ATOL)-Fähigkeit zur Gewährleistung zentraler zukünftiger Missionsanforderungen.	
19. Schlagwörter AAA-Prozess, Flexibilität, Fehlertoleranz, Plattformansatz, Software Architektur, Systemdesign	
20. Verlag Technische Informationsbibliothek (TIB), Welfengarten 1b, 30167 Hannover	21. Preis -

Document Control Sheet

1. ISBN or ISSN planned	2. type of document (e.g. report, publication) Final report
3. title FLYSMART Leitprojekt FBW23 – FBW-Plattformtechnologie	
4. author(s) (family name, first name(s)) Reichel, Reinhard Fichter, Walter	5. end of project June 2015
	6. publication date October 2015
	7. form of publication Technical report
8. performing organization(s) (name, address) Universität Stuttgart, Institut für Luftfahrtsysteme (ILS) Pfaffenwaldring 27 70569 Stuttgart	9. originator's report no. -
	10. reference no. 20V1108N
	11. no. of pages 50
12. sponsoring agency (name, address) Bundesministerium für Wirtschaft und Energie (BMWi) 53107 Bonn	13. no. of references 29
	14. no. of tables 1
	15. no. of figures 27
16. supplementary notes -	
17. presented at (title, place, date) -	
18. abstract The design and certification of fly-by-wire systems for automated utility aircraft and intelligent transport aircraft of classes EASA CS23 / Classl...III is a big challenge and requires fundamentally new technologies. The AAA-platform technology can be seen as an appropriate approach. This technology is characterized by a strict layer structure of the overall management software with the platform management as a central layer. Thereby the complexity of the system (with redundancy and fault tolerance) is not visible to the law. The AAA process enables the automatic generation of the platform management for a new fly-by-wire system to be realized. Thus, such systems can be develop and qualify very efficiently (in the future). Central tasks of this project was the development and validation of the Axx-Process (for a widely automatic generation of platform management) to ensure the applicability within an industrial CS23 environment and the development and validation of functions (Laws) to enhance the automatic mission capacity (incl. ATOL) for future utility and transport aircraft EASA CS23. The feasibility of the Axx-Process was given numerous research projects. Demonstrators in CS23-, helicopter and automotive sector, verified its flexibility. Nevertheless, significant research activities were essential to transfer this technology in an industrial CS23 environment. This applies to the functional side too, especially to the aspects automation of aircraft system (system transparency to the pilot) and Automatic Take Off and Land (ATOL) capability.	
19. keywords AAA-Process, flexibility, fault-tolerance, platform approach, software architecture, system design	
20. publisher Technische Informationsbibliothek (TIB), Welfengarten 1b, 30167 Hannover	21. price -