Software Development of the EF2000 Flight Control Computers

Dr. H. Heusinger
EADS Deutschland GmbH
Military Air Systems
Military Air Systems

Agenda

• Overview of Flight Control System
• Overview of Flight Control Computer
• Design Process
  – System Requirements
  – Software Requirements
  – Software Design
  – Test
• Tools
Military Air Systems

EF2000 Requirements

Goal: Achieve high survivability and mission reliability

- high agility
- carefree flying
- automatic recovery
- autopilot / autothrottle
- failure tolerant
- structural protection
- easy maintenance
Military Air Systems

Aerodynamic Control Surfaces

- Single Rudder
- Spine Airbrake
- Full-Span Trailing Edge Flaperons
- 3/4-Span Leading Edge Slats
- Symmetric Foreplanes
Military Air Systems

EF2000 Flight Control System

Eurofighter Flight Control System

List of Abbreviations
ADT  Air Data Transducers
AVS  Avionic System
CCDL  Cross Channel Data Link
DECU  Digital Engine Control Unit
FCC  Flight Control Computer
FCS  Flight Control System
IMU  Inertial Measurement Unit
PSU  Pedal Sensor Unit
SSICA  Stick Sensor and Interface Control Assembly
UCS  Utility Control System

Primary Actuators
1  Foreplane L
2  Foreplane R
3  O/B Flaperon L
4  O/B Flaperon R
5  I/B Flaperon L
6  I/B Flaperon R
7  Rudder
8  Leading Edge
9  Airbrake
10  Airintake L
11  Airintake R
12  Nose Wheel

Secondary Actuators

Engine L
Engine R
DECU L
DECU R

Analog Wiring

FCS-Bus STANAG 3838

AVS-Bus STANAG 3910

17 PSU
17 PSU
16 SSICA
18 Throttle Boxes

Preconditioned Power

UCS-Bus STANAG 3838

14 IMU
13 FCCs

ADTs

CCDL
Military Air Systems

Flight Control Computer Architecture

Flight Control Computer

Analogous inputs
Excitation
External Discrete I/Ps
FCS Bus Controller
STANAG 3910 I/F
AVS/UCS RT

CCDL Tx/Rx
I/O µP
I/O Sequencer

Air Data µP
Actuator µP
Analogue O/Ps

Primary DDV
Leading Edge System
Secondary EHSV

PS External
Actuator Discrete O/Ps

CL3 µP
CL1 µP
CL2 µP

PS Internal
PS External

SSICA
IMU
ADT
Inter Processor Communication

Common Datastore

IO, ACT, AIR, CL1, CL2, CL3

Bus Arbiter

IO Sequence

T-Bus
Military Air Systems

General FCS Requirements

- A single failure must not endanger the mission success
- A second failure must not generate a safety hazard
- All FCCs shall be interchangeable
- All FCCs shall generate the same actuator demands at all times (even with failures)
- There must be an instantaneous reaction to the pilot’s inputs
Interchangeable FCCs

Into which slot fits this FCC?

• FCCs must be interchangeable
• FCCs must have identical software
• FCS must verify the HW / SW standard
• FCC must take over stored data from others
Generate Identical Actuator Demands

General Approach
Military Air Systems

Synchronisation

- Each FCC generates a master interrupt every 12.5 ms.
- The FCC tries to keep the timer interrupts within 100 μs.

General Housekeeping

Send Sync Marker A

Sync Window +/- 50 μs

Receive SyncB
Receive SyncC
Receive SyncD

Adjust Timer:
\[ A = A - \frac{1}{4}(\Sigma \text{Times B/C/D received}) \]

If Sync is lost then the Frame Number drives the adjustment
Military Air Systems

Channel Exclusion (part 1)

FCC2 detects CCDL fail from FCC1 due to missing checkword

FCC2 tells FCC4 that it cannot see FCC1 but all others; FCC4 assumes FCC1 transmitter faulty
FCC2 detects CCDL fail from FCC1 and FCC4

FCC2 tells FCC4 that it can’t see FCC1 and FCC2; FCC4 assumes FCC2 receiver faulty and works quadruplex
FCC1

FCC2 detects CCDL fail from all others
FCC2 isolates itself

FCC3

FCC4 detects CCDL fail from FCC2

Page 15
Military Air Systems

Software Design Process

SW Requirements Analysis

FCC Algorithms Timing Requirements

SW Top Level Design

Ada Package Descriptions:
• Allocation to Processors
• Scheduling of Tasks

SW Detailed Design / Code

Ada / Assembler Code
• Implementation of functions
• Optimisations
Software Requirement (Example 1)

The actuator inner loop shall implement the following:

Iteration Rate: 320 Hz
Delay from Valve Feedback to Motor Demand $\leq t_1$ ms
Software Requirement (Example 2)

The pilot’s inceptor demands shall flow through the system as:

- **Consolidation**
- **Arbitration**
- **Control Law Execution**
- **Actuator Outer and Inner Loop**

Iteration Rates: 40 Hz; 160 Hz; 320 Hz

Fastest Delay from SSICA/RSSICA to Motor Demand ≤ t₂ ms
Military Air Systems

Software Top Level Design

Tasks:

- Define IO Sequence
- Allocate the functions to the processors
- Define the Ada Package Specifications
- Define the communication between the processors
- Estimate the worst case run time, the program store and the needed RAM for each module
- Generate the schedulers for the modules
Software Top Level Design (IO Sequence)

Principle: Just in time is in time

SSICA to Actuator

Bus Message

CCDL Pilot Dmd

Engineering Judgement

Delay Req

CCDL Actuator Dmd

Iteration Rate

DDV Demand

Actuator Inner Loop

Iteration Rate

Analogue Input

Valve Position

Delay Req

Just in time is in time
Military Air Systems

Software Top Level Design (Communication)

Worst Case Time when BufferToCom is finished < t
Military Air Systems

Software Top Level Design (Scheduling)

Functional Chain with Delay Requirement

640 Hz
Act Inner Loop
SSICA to Act

without Delay Requirement

Iteration Rate
1.25 Hz

Workload (Excel Sheet)

ACT
IO
CL1

Define the buffer modules now
Military Air Systems

Software Coding (Standard)

- SafeAda must be used
  - No tasking
  - No predefined exceptions
  - No pointers
  - No unchecked conversion
  - No recursive calls
  - etc.

- Assembler should not be used
  - runtime system
  - highly used components (e.g. voter / monitors, filters, interpolations, etc.)
  - allocation of memory
Waits can speed up the program!!

Access to COM slow:
- slow T-Bus
- 6 bus clashes

Waits guarantee that at most 3 bus clashes occur
Count Clock Cycles for each procedure taking into account:
- branches
- addressing modes
- memory wait states
- misalignment of data
- parallelism between main- and coprocessor

Analyse Transport Delays
Software Coding (Hazard Analysis)

Prove that no hazard is introduced due to the coding

- Divide by zero Analysis
- Overflow Analysis
- Denormalisation Analysis
- Memory Usage Analysis
- Ada / Assembler Interface Analysis
- Iteration Rate Analysis
- Illegal Instruction Analysis
- Recursion Analysis
- Compiler Defect Analysis
- etc.
Software Coding (Limited Change Capability)

Difference of Object Code drives regression test

small change of source  small change of linked program

We must be able to fix code and data in memory

Sometimes it’s better to move unchanged data rather than changed data

Changes of the runtime and data flow must be handled locally
Software Testing

Unit Tests
Verify that each procedure implements the design given in Top Level / Detailed Design (Statement, Branch and MC/DC coverage on target)

Software Integration Tests
Verify that all calls to procedures are correct according to the Top Level / Detailed Design
Military Air Systems

Software Testing

Hardware / Software Integration Tests
Verify that each software requirement is correctly implemented in the actual

Test Equipment
- Bus Stimulation/Monitoring
- Actuator Model
- RS232 Commands
- CCDL Substitution/Monitoring

FCC
- RS232 Interpreter
- Flight Test Output

CCDL SubBox
Military Air Systems

Software Testing (Timing)

Signal Generator and Analyser

Analogue Input

Analogue Output

FCC
Military Air Systems

Tool Support

Generator for Schedulers

- Functional Chains (inc. Iteration Rates)
- WCRT Estimates
- Deadlines
- Delay Requirements
- IO Sequence
- Accuracies

Schedulers
Buffer Modules
Spare Time
Accuracies
Military Air Systems

Tool Support (Validated)

Worst Case Runtime Analyser
- new processor MPC 565
- cache enabled
- runtimes with parameters

Transport Delay Analyser
- graphical interface to data flow diagrams
- automatic input of IO sequence
- automatic bus clash analysis