AIRBUS FLY-BY-WIRE: A TOTAL APPROACH TO DEPENDABILITY

Pascal Traverse, Isabelle Lacaze and Jean Souyris
Airbus, 316, route de Bayonne, 31060 Toulouse, France
{pascal.traverse, isabelle.lacaze, jean.souyris}@airbus.com

Abstract: This paper deals with the digital electrical flight control system of the Airbus airplanes. This system is built to very stringent dependability requirements both in terms of safety (the systems must not output erroneous signals) and availability. System safety and availability principles are presented with an emphasis on their evolution and on future challenges

Key words: dependability, fault-tolerance, safety, proof, human factors, system design, airplane, fly-by-wire, flight controls

1. INTRODUCTION

1.1 Background

The first electrical flight control system (a.k.a. Fly-by-Wire) for a civil aircraft was designed by Aerospatiale and installed on Concorde. This is an analogue, full-authority system for all control surfaces and copies the stick commands onto the control surfaces while adding stabilizing terms. A mechanical back-up system is provided on the three axes.

The first generation of electrical flight control systems with digital technology appeared on several civil aircraft at the start of the 1980's including the Airbus A310. These systems control the slats, flaps and spoilers. These systems have very stringent safety requirements (in the sense that the runaway of these control surfaces is generally classified as Catastrophic and must then be extremely improbable). However, loss of a function is permitted, as the only consequences are a supportable increase in the crew's workload.

The Airbus A320 was certified and entered into service in the first quarter of 1988. It is the first example of a second generation of civil electrical flight control aircraft, which is now a full family (A318, A319, A320, A321, A330, A340). The distinctive feature of these aircraft is that high-level control laws in normal operation control all control surfaces electrically and that the system is designed to be available under all circumstances.

This family of airplane has accrued a large and satisfactory service experience with more than 10000 pilots operating a Fly-by-Wire Airbus, and more than 40 million flight hours. Nevertheless, system architecture is permanently challenged to take benefit of technical progress and of this large in-service experience. Indeed, on top of the architecture level reached by A340[1,2], A340-600, A380, and A400M are going steps further.

The A340-600 is the first significant change compared to the A320/A330/A340 baseline. It entered into service mid of 2002, introducing structural modes control, a full rudder electrical control and integration of autopilot inner loop with manual control laws. The full rudder electrical control is now part of all A330 and A340 definition.

A380 and A400M will be the first in-service aircraft with electrical actuation of control surfaces (a.k.a. Power-by-Wire). Additionally, new avionics principle are applied and a full autopilot and manual control integration is performed.

Other architectures are possible[3]. The family of architectures we have designed has the merit of having been built step-by-step, together with our products development and experience.

1.2 Fly-by-wire principle

On a conventional airplane, the pilot orders are transmitted to the actuators by an arrangement of mechanical components. In addition, computers are modifying pilot feels on the controls, and autopilot computers are able to control servo actuators that move the whole mechanical control chain.

The A320/A330/A340 Airbus flight control surfaces are all electrically controlled, and hydraulically activated.

The side-sticks are used to fly the aircraft in pitch and roll (and indirectly through turn co-ordination in yaw). The pilot inputs are interpreted by the flight controls computers that move the surfaces as necessary to achieve the desired flight path modification. In autopilot mode, the flight controls computers take their orders from the autopilot computers. With this respect, the flight controls are composed of five to seven computers, and the autopilot of two.

The aircraft response to surfaces movement is fed back to both autopilot and flight controls computers through specific sensors (Air Data and Inertial Reference Units - ADIRU, accelerometers, rate-gyro).

1.3 On failure and dependability

Flight control systems are built to very stringent dependability requirements both in terms of safety (the system must not output erroneous signals) and availability. Most, but not all, of these requirements are directly coming from Aviation Authorities (FAA, EASA, etc. refer to FAR/JAR 25[4]).

Remaining of the paper is structured around threat to safety and availability of the system[5], namely:

- Failures caused by physical faults such as electrical short-circuit, or mechanical rupture
- Design and manufacturing error
- Particular risks such as engine rotor burst
- Mishap at Man-Machine Interface

Interestingly, means against these threats to dependability are valuable protection against malicious faults and attacks, on top of classical security measures.

For each of these threats, the applicable airworthiness requirements are summarized; the solutions used on Airbus Fly-by-Wire are described, along with challenges to these solutions and future trends.

2. SYSTEMS FAILURES DUE TO PHYSICAL FAULTS

FAR/JAR 25.1309 that requires demonstrating that any combination of failures with catastrophic consequence is Extremely Improbable typically addresses failures. “Extremely Improbable” is translated in qualitative requirements (see § 3 to 5) and to a 10^{-9} probability per flight hours. Specifically for flight controls, FAR/JAR 25.671 requires that a catastrophic consequence must not be due to a single failure or a control surface jam or a pilot control jam. This qualitative requirement is on top of the probabilistic assessment.

To deal with the safety issue (the system must not output erroneous signals), the basic building blocks are the fail-safe command and monitoring computers. These computers have stringent safety requirements and are functionally composed of a command channel and a monitoring channel.

To ensure a sufficient availability level, a high level of redundancy is built into the system.

2.1 Command and monitoring computers

2.1.1 Computer architecture

Functionally, the computers have a command channel and a monitoring channel (see figure 1.a). The command channel ensures the function allocated to the computer (for example, control of a moving surface). The monitoring channel ensures that the command channel operates correctly. This type of computer has already been used for the autopilot computers of Concorde, and the Airbus aircraft.

These computers can be considered as being two different and independent computers placed side by side. These two (sub) computers have different functions and software and are placed adjacent to each other only to make aircraft maintenance easier. Both command and monitoring channels of one computer are active simultaneously, or waiting, again simultaneously, to go from stand-by to active state. When in stand-by mode, computers are powered in order to activate potential dormant faults and isolate them. The monitoring channel acts also on associated actuator: when deselecting the COM order, it switches off the actuator solenoid valve to set it in stand-by mode (figure 1.b).

Two types of computers are used in the A320 flight control system: the ELAC's (ELevator and Aileron Computers) and the SEC's (Spoiler and Elevator Computers). Each computer has a command channel and a monitoring one. Thus, four different entities coexist: command channel of ELAC computer, monitoring channel of ELAC computer, command channel of SEC computer, and monitoring channel of SEC computer. This leads to four different software packages.

Two types of computers are also used on the A340 and A380: the PRIM's (primary computers) and the SEC's (secondary computers). Although these computers are different, the basic safety principles are similar and described in this part of the paper.

In addition to the ELAC's and SEC's of the A320, two computers are used for rudder control (FAC). They are not redundant to the ELAC's and SEC's. On other Airbus, these rudder control functions are integrated in the PRIM's and SEC's.

2.1.2 Computer channel architecture

Each channel (figure 1.a) includes one or more processors, associated memories, input/output circuits, a power supply unit and specific software. When the results of one of these two channels diverges significantly, the channel or channels which detected this failure cut the links between the computer and the exterior.

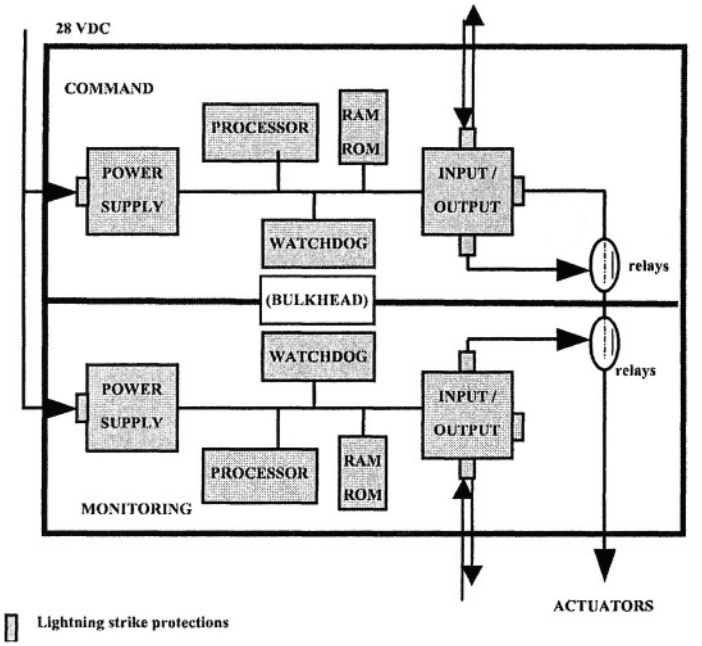

Figure 1.a: computer global architecture

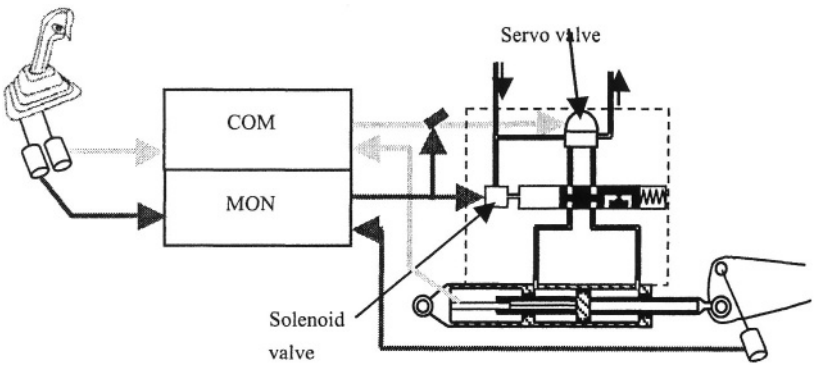

Figure 1.b: computer monitoring architecture

The system is designed so that the computer outputs are then in a dependable state (signal interrupt via relays). Failure detection is mainly achieved by comparing the difference between the command and monitoring commands with a predetermined threshold. This schema therefore allows the consequences of a failure of one of the computer's components to be detected and prevents the resulting error from propagating outside of the computer. This detection method is completed by monitoring for good execution of the program via its sequencing and memory encoding.

Flight control computers must be especially robust. They are protected against over voltages and under voltages, electromagnetic aggressions and indirect effects of lightning. They are cooled by a ventilation system but will operate correctly even if ventilation is lost.

2.1.3 Redundancy

The redundancy aspect is handled at system level. This paragraph only deals with the computer constraints making system reconfiguration possible. The functions of the system are divided out between all the computers so that each one is permanently active at least on one subassembly of its functions. For any given function, one computer is active the others are in standby ("hot spares"). As soon as the active computer interrupts its operation, one of the standby computers almost instantly changes to active mode without a jerk or with a limited jerk on the control surfaces. Typically, duplex computers are designed so that they permanently transmit healthy signals and so that the signals are interrupted at the same time as the "functional" outputs (to an actuator for example) following the detection of a failure.

2.1.4 Failure detection

Certain failures may remain masked a long time after their creation. A typical case is that of a monitoring channel made passive and detected only when the monitored channel itself fails. Tests are conducted periodically so that the probability of the occurrence of an undesirable event remains sufficiently low (i.e., to fulfill FAR/JAR 25.1309 quantitative requirement). Typically, a computer runs its self-tests and tests its peripherals during the power-up of the aircraft and therefore at least once a day.

2.2 Components redundancy

2.2.1 Power supplies

Primary power is coming from the engines to pressurize hydraulic systems and to generate electricity. Also, an auxiliary generator, batteries and a Ram Air Turbine (RAT) are available. If all engines shut down, the RAT is automatically extended. It then pressurizes a hydraulic system that drives a third electrical generator. The computers are connected to at least two electrical power supplies. The aircraft has three hydraulic systems (identified by a color, Green, Blue, and Yellow on figure 2 for A340-600) one of which is sufficient to control the aircraft.

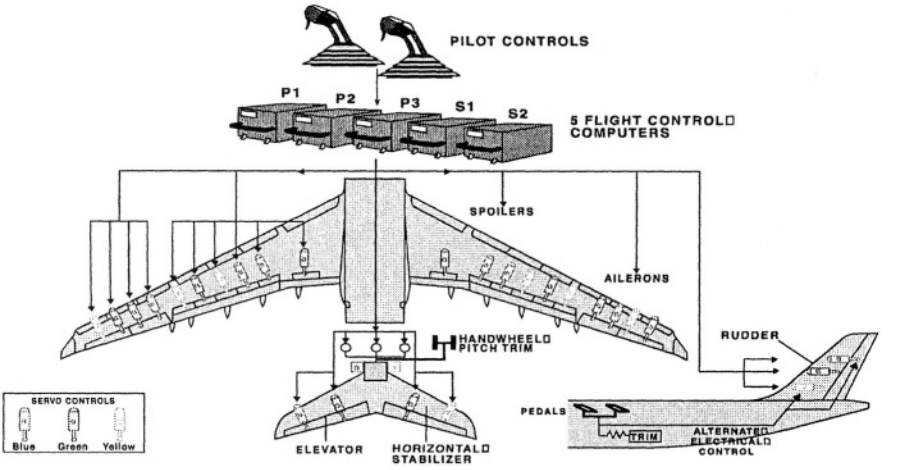

Figure 2: A340-600 system architecture

As a new technology of actuators is now available[6] (Electro Hydrostatic Actuator – EHA – see figure 3.a, compared to conventional servocontrol, figure 3.b) it is possible to take benefit of them. This is done on A380 and A400M. The 3 hydraulic power supplies are replaced by 4, 2 hydraulic ones and 2 electrical ones. RAT is providing directly electrical power. This provides a weight and cost saving along with an increased redundancy and survivability, which was the primary reason for the introduction of this technology.

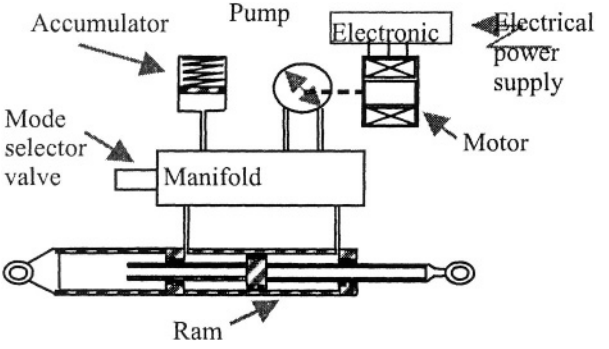

Figure 3.a: Electro-hydrostatic Actuator

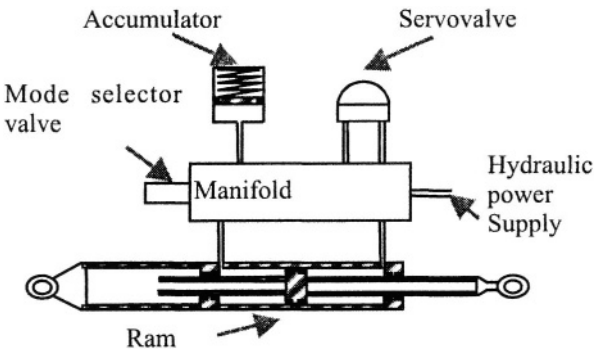

Figure 3.b: Hydraulic Servocontrol

2.2.2 Computers

The computers and actuators are also redundant. This is illustrated by the A340-600 pitch control (left and right elevator, plus Trimable Horizontal Stabilizer - THS). Four command and monitoring computers are used, one is sufficient to control the aircraft. In normal operation, one of the computers (PRIM1) controls the pitch, with one servocontrol pressurized by the Green hydraulic for the left elevator, one pressurized by the Green hydraulic on the right elevator, and by electric motor N°1 for the THS. The other computers control the other control surfaces. If PRIM1 or one of the actuators that it controls fails, PRIM2 takes over (with the servocontrols pressurized by the Blue hydraulic on left elevator, yellow on right side, and with THS motor N°2). Following same failure method, PRIM2 can hand over control to SEC1. Likewise, pitch control can be passed from one SEC to the other depending on the number of control surfaces that one of these computers can handle. Note that 3 computers would be sufficient to meet the safety objectives. The additional computer is fully justified by operational constraints: it is desirable to be able to tolerate a take-off with one computer failed. This defines the Minimum Equipment List (MEL).

2.2.3 Reconfiguration of flight control laws and flight envelope protections

Note that the laws are robust as designed with a sufficient stability margin[7-10]. Also, if the input vector of the system is far outside the maximum certified envelope, only a simple law, using the position of the sticks and the position of the control surfaces at input, is activated (this law is similar to the type of control available on a conventional aircraft).

The laws must be reconfigured if certain sensors are lost (in particular, the ADIRU's). The crew is clearly warned about the status of the control law. If the three ADIRU's are available (normal case), the pilot has full authority within a safe flight envelope. This safe flight envelope is provided by protections included in the control laws, by addition of protection orders to the pilot orders. Flight control is in G-load factor mode.

If only one ADIRU is available, it is partially monitored by comparison with other independent information sources (in particular, an accelerometer). In this case, the safe flight envelope is provided by warnings, as on a conventional aircraft. Flight control is still in G-load factor mode. If all ADIRU's are lost, the flight envelope protections are also lost and the flight control law is in a degraded mode: direct mode. This law has gains, which are a function of the aircraft configuration (the position of the slats and the flaps), and allows here again flight control similar to that of a conventional aircraft.

2 .3 Challenges and trends

On computer side, there is no major change in sight, apart from physically cutting a COM/MON computer into two units. This coupled with an increase self-test capability could provide a reduction of spare needs. This will be applied on A380 PRIM. Another trend is to design fully portable software. This could be used to get exactly the same software on simulators as on airplane.

In term of communications between computers, a step has been done on A380 and A400M by using a deterministic Ethernet network, for non-critical data and functions. Next step could be to use more smart actuators, and thus a digital network between them and computers.

3. DESIGN AND MANUFACTURING ERRORS

These errors are addressed by FAR/JAR 25.1309 that mandates to follow a stringent development process, based on following guidelines:

- ARP4754/ED79[11] for aircraft system development
- DO178/ED12[12] for software development
- DO254/ED80[13] for hardware development

There is no clear requirement that a design must be design-fault-tolerant, except if the applicant wishes to reduce its development assurance effort.

On Airbus EFCS, both ways are used:

- Error-avoidance with a stringent development process
- Error-tolerance as well.

3.1 Error avoidance

Aviation guidelines are applied, with the highest level of Development Assurance Level (level A). A340-600 EFCS is even likely to be the first system to be certified according to ARP 4754 level A.

3.1.1 On computer functional specification

The specification of a computer includes, on the one hand, an "equipment and software development" technical specification used to design the hardware and, in part, the software, and, on the other hand, an "equipment functional specification" which accurately specifies the functions implemented by the software.

This functional specification is a key element in the Fly-by-Wire development process. It is designed by engineers skilled in automatic control and aircraft system sciences and used by software engineers. Although system and software engineers are knowledgeable in each other field, and are working in the same company with the same objective, it is mandatory that the functional specification be non-ambiguous for each discipline. It is written using a graphic computer-assisted method. Specification language is named SCADE, a derivative of a previous one: SAO. All of the computer functions are specified with this method: flight control laws, monitoring of data, actuators, slaving of control surfaces, reconfigurations, etc. Timing of these functions is very simple. Scheduling of operations is fixed and run continuously at a fixed period. One of the benefits of this method is that each symbol used has a formal definition with strict rules governing its interconnections. The specification is under the control of a configuration management tool and its syntax is partially checked automatically.

Hence, validation and verification activities are addressed in this paper in three steps: system architecture and integration, computer functional specification, computer software.

For the translation of functional specification into software, the use of automatic programming tools is becoming widespread. This tendency appeared on the A320 and since A340-600 both PRIM and SEC are programmed automatically for a significant part. Such a tool has as input the functional specification sheets, and a library of software packages, one package for each symbol utilized. The automatic programming tool links together the symbol packages.

The use of such tools has a positive impact on safety. An automatic tool ensures that a modification to the specification will be coded without stress even if this modification is to be embodied rapidly (situation encountered during the flight test phase for example). Also, automatic programming, through the use of a formal specification language, allows onboard code from one aircraft program to be used on another. Note that the functional specification validation tools (simulators) use an automatic programming tool. This tool has parts in common with the automatic programming tool used to generate codes for the flight control computers. This increases the validation power of the simulations.

3.1.2 System architecture and integration V&V

The system validation and verification proceeds through several steps:

- Peer review of the specifications, and their justification. This is done with the light of the lessons learned by scrutinizing incidents that occur in airline service
- Analysis, most notably the System Safety Assessment which, for a given failure condition, checks that the monitoring and reconfiguration logics allow to fulfill the quantitative and qualitative objectives, but also analysis of system performances, and integration with the structure
- Tests with a simulated system, taking credit to the automatic coding of the functional specification, with a coupling with a rigid aircraft model
- Test of equipment on a partial test-bench, with input simulation and observation of internal variables (for computers)
- Tests on iron bird and flight simulator. The iron bird is a test bench with all the system equipment, installed and powered as on aircraft. The flight simulator is another test bench with an aircraft cockpit, flight controls computers, and coupled with a rigid aircraft model. The iron bird and the flight simulator are coupled for some tests.
- Flight-tests, on up to four aircraft, fitted with a "heavy" flight test instrumentation. More than 10000 flight controls parameters are permanently monitored and recorded.

The working method for these tests is twofold. A deterministic way is used, based on a test program, with a test report answering. In addition, credit is taken of the daily use of these test facilities for work on other systems, for demonstration, or test engineer and pilot activity. If the behavior of the system is not found satisfactory, a Problem Report is raised, registered and investigated.

3.1.3 Verification and validation of functional specifications

Certain functional specification verification activities are performed on data processing tools (e.g., the syntax of the specification can be checked automatically). A configuration management tool is also available and used.

The specification is validated mainly by means of proofreading (in particular, during the safety analysis), analysis, and ground or flight tests (see § 3.1.2). Analyses are more or less aided by tools, and address topics such as uncertainties propagation and timing for robustness. Our target is validation at earliest possible stage. To achieve this, various simulation tools exist and this because the specifications were written in a formal language making the specification executable.

This makes it possible to simulate the complete flight control system: computers, actuators, sensors, and aircraft returns (OCASIME tool). It is also possible to inject with this tool some stimuli on data that would not be reachable on the real computer. The signals to be observed can be selected arbitrarily and are not limited to the inputs/outputs of a specification sheet. The test scenarios thus generated can be recorded and rerun later on the next version of the specification, for example. A global non-regression test is in place, allowing for each new standard of computer specification, to compare the test results of the previous version, and of the new version. This comparison allows detecting modification errors.

Also, the part of the specification that describes the flight control laws can be simulated in real time (same Ocasime tool) by accepting inputs from a real sidestick controller (in fact, simpler than an aircraft stick), and from the other aircraft controls. The results are provided on a simulated Aircraft Primary Flight Display for global acceptance, and in more detailed forms, for deep analysis. The Ocasime tool is coupled to an aerodynamic model of the aircraft.

Test scenarios are defined based on the functional objectives of the specification, including robustness and limit tests. Some formal proofs are performed too, but still on a very limited basis.

3.1.4 Software

The software is produced with the essential constraint that it must be verified and validated. Also, it must meet the world's most severe civil aviation standards (currently level A software to DO178B). The functional specification acts as interface between the aircraft manufacturer's world and the software designers' world. The major part of the flight control software specification is a copy of the functional specification. This avoids creating errors when translating the functional specification into the software specification. For this "functional" part of the software, validation is not required as covered by the work carried out on the functional specification.

Actually, the whole software is divided in five programs plus one library. The programs are: the applicative program, automatically produced from the functional specification, as mentioned above; the self-tests; the initialization and applicative tasks sequencing; the download function; the input/output software. The library is the set of basic code components that implement the graphical SCADE – or SAO – basic components (OR, AND, FILT, etc) of the functional specification.

With respect to the applicative (functional) program, checking that the applicative tasks are schedulable must be performed "at software level". Indeed, to make software verification easier, the various tasks are sequenced in a predetermined order with periodic scanning of the inputs. Only the clock can generate interrupts used to control task sequencing. This sequencing is deterministic. A part of the task sequencer validation consists in methodically evaluating the margin between the maximum execution time for each task (worst case) and the time allocated to this task.

Lets now focus on the non-applicative software parts. Their development (called *life-cycle* by DO 178B) requires to successively specify, design and write the code. The verification techniques used for getting confident in the results of each activity and on the whole program are traditionally based on tests, readings and intellectual analyses.

In A380 software development, tool-aided software proof techniques were introduced into the verification workbench.

Lets take a example of one of the most important software verification: Unit verification, which is used for demonstrating that the software components (like C routines), once coded, conforms their definition, made at design time.

An important criterion of the quality of a verification process is its *functional coverage,* regardless of the verification technique used. In Unit Verification, satisfying this criterion consists in making sure that for each design component, there exists a code component which is verified by a "verification entity" allowing for the checking of all the Low Level Requirements (DO 178B terminology) expressed for this component at design time.

When the verification technique is the *test,* these Low Level Requirements are verified by applying the so-called "equivalent class" method. Adequate *functional coverage* of the Low Level Requirements must be obtained for the range of values of the inputs of a code component. The term "adequate" does not mean that the tests are assumed to be exhaustive, which is practically impossible to achieve, but means that "equivalent classes" are defined for covering all expected behaviors. The test cases actually performed are the most representative of each "equivalent class".

When a tool-aided *proof* method is used for Unit Verification[14], the *functional coverage* of the Low Level Requirements is a lot more directly obtained. Indeed, the Low Level Requirements are expressed formally by

first order predicates at design time, and the verification consists of applying the tool-aided proof method for demonstrating that these predicates hold on the code component, i.e., for all possible behaviors.

When the verification technique is the *test,* an additional criterion has to be fulfilled: the *structural coverage.* It consists, for each software component, of checking that 100% of the instructions, 100% of the decisions and 100% of the Modified Conditions/Modified Decisions are exercised during tests. These *structural coverage* criteria are completely specified by DO 178B.

Beyond Unit Verification, the following other verification activities benefit from tool-aided proof techniques: the safe stack maximum usage computation and the safe Worst Case Execution Time computation[15] for all functional tasks. This kind of automatic demonstration that a whole program actually possesses some characteristics is of great interest with respect to dependability properties.

The verification techniques, like those described above, and a possible additional verification effort have the approval of the various parties involved (aircraft manufacturer, equipment manufacturer, airworthiness authorities, designer, quality control).

The basic rule to be retained is that the software is made in the best possible way. This has been recognized by several experts in the software field both from industry and from the airworthiness authorities. Dissimilarity is an additional precaution that is not used to reduce the required software quality effort.

3.1.5 Challenges and trends

With respect to error-avoidance we are faced with the challenge to get the system right the first time. This leads more and more to move V&V upstream and to partially automate it. We have also an opportunity that is the level of formalism of functional specification language. This should make more way to prove formally properties of the system and to measure the structural coverage of the tests performed.

Applied to software verification, this leads to use formal verification (tool-aided proof methods, static analysis) widely. As stated in section 3.1.4, a first set of proof techniques has been introduced in the verification workbench, i.e., for Unit Verification, safe maximum stack usage and Worst Case Execution Time computations.

These first applications cover a small subset of all software verification objectives whereas the underlying theoretical framework, i.e., the Abstract Interpretation theory[16], makes it possible, in the future, to get other applications (as automatic tools) like: the proof of absence of Run Time Error[17]; the analysis of the quality of floating point calculus[18]; the proof of properties (predicates) on whole programs (not limited to Unit Verification).

Moreover, more assurance about the system will be obtained earlier in the development process by using Abstract Interpretation based verification tools for proving dependability properties by analysis of the formal functional specification.

The objective is the effective application of the Product Based Assurance concept in which the confidence in the program is not only based on the quality of its development (Process Based Assurance) but also on its properties, as a product.

3.2 Error tolerance

3.2.1 Dissimilarity

The flight control system was subjected to a very stringent design and manufacturing process and we can reasonably estimate that its safety level is compatible with its safety objectives. An additional protection has nevertheless been provided which consists in using two different types of computers: for example, A320's ELAC is based on 68010 microprocessors and the SEC on the 80186; A340's PRIM on 80386, and the SEC on 80286; A380's PRIM on Power PC and the SEC on Share processor. Automatic coding tools are different too.

Functional specification and hence the software are different too; ELAC and PRIM run the elaborate functions while SEC is simpler (less functions, less stringent passenger comfort requirements) and thus more robust.

Within a computer, COM and MON hardware are basically of a same design, but with different software.

We therefore have two different design and manufacturing teams with different microprocessors (and associated circuits), different computer architectures and different functional specifications (ELAC vs. PRIM on A320; PRIM vs. SEC on A330/A340/A380). At software level, the architecture of the system leads to use 4 software packages (ELAC/COM, ELAC/MON, SEC/COM, SEC/MON) when, functionally, one would suffice. This is still applicable to PRIM and SEC of A330/A340 and A380.

3.2.2 Data diversity

As part of a struggle against single point of failures, the system is loosely synchronized. Computers are synchronizing their data both internally (command/monitoring) and between them (PRIM1, 2 ...) but not their clocks. Hence, for a given piece of information computers are using different data, sampled at different time. This is felt as an additional robustness margin.

3.2.3 Challenges and trends

A challenge to error tolerance is the reduction of electronic component suppliers: it becomes more and more likely that if two design teams (one for PRIM, one for SEC) choose independently their components, they will end up with some in common. Hence, we have moved from this kind of "random" dissimilarity to a managed one, such that both computer design teams decide in common to take different components.

In-service experience has shown that PRIM/SEC dissimilarity is fully justified. Indeed, two cases showed that this dissimilarity was beneficial for system availability. During one A320 flight, both ELAC were lost following an air conditioning failure and the subsequent abnormal temperature rise. It appears that a batch of these computers was fitted with a component whose temperature operating range did not match exactly the specified range. During one A340 flight, a very peculiar hardware failure of a single component trapped all three PRIM logic temporarily (reset was effective).

EHA are also an opportunity to get dissimilar actuation power supplies: indeed, A380 and A400M will be able to tolerate a complete loss of hydraulic power.

4. PARTICULAR RISKS

Particular risks are spread within FAR/JAR. ARP 4761[19] tends to regroup most of them.

Basically, the concern with this type of event is that it can affect several redundancies in a single occurrence.

Airbus addresses this concern by building a robust system and qualifying its components accordingly (against vibration, temperature...). Additionally, emphasis is put on separating physically the system resources, segregating them, and by providing an ultimate back-up redundant to the EFCS.

4.1 Segregation

The electrical installation, in particular the many electrical connections, comprises a common-point risk. This is avoided by extensive segregation: in normal operation, two electrical generation systems exist without a single common point. Computers are divided in two sets associated to these two electrical generation systems. The links between computers are limited, the links used for monitoring are not routed with those used for command. We end up with at least four different electrical routes: COM of electrical system 1, MON of electrical system 1, COM of electrical system 2, MON of electrical system 2. This proved useful when a case of electrical arc tracking occurred: all the wires in a single bundle have been destroyed, but other

located elsewhere were sufficient to ensure continued safe flight and landing, with margin.
The destruction of a part of the aircraft is also taken into account: the computers are placed at three different locations, certain links to the actuators run under the floor, others overhead and others in the cargo compartment. Power supplies are also segregated. It is worth noting here again the benefit of EHA, as electrical power cables are easier to install and thus it is possible to get more space between all the power transmission lines (electrical cables and hydraulic pipes).

4.2 Ultimate back-up

In spite of all these precautions, a mechanical standby system has been conserved on A320 to A340. This mechanical system is connected to the trimmable horizontal stabilizer allowing the pitch axis and the rudder to be controlled providing direct control of the yaw axis and indirect control of the roll axis. The safety objectives for the fly-by-wire part of the system (PRIM's plus SEC's) have been defined without taking credit of this mechanical back-up.

A340-600 needs a precise rudder control to damp structural vibration. This is difficult to get with an ageing mechanical control, prone to threshold and freeplay. Hence, A340-600 rudder control is fully electrical (like an elevator or an aileron on A320 or basic A340). A new ultimate back-up has thus been designed, which is electrical with an autonomous power converter (from hydraulic to electricity), completely independent from the basic system of PRIM's and SEC's, integrating a yaw rate-gyro, pedals sensors, rudder servocontrol servoloop.

On A380 and A400M the last step is done: the mechanical linkage from cockpit control wheel to the actuator of the horizontal stabilizer is cancelled. Ultimate back-up is thus similar to A340-600 rudder one, but controlling rudder, one pair of elevators, and one pair of ailerons, based on pedals and sidesticks order. Technology is currently analog.

4.3 Challenges and trends

Fiber optics is used on A340-600 and A380 for the "Taxi Aid Camera System". This non-critical system is partially installed in the fin and in non-pressurized area. It should demonstrate that fiber optics can be used in this kind of difficult area and are compatible with standard airline maintenance practices. This will open the door to introduce this technology on civil fly-by-light systems. Current systems are sufficiently immune to electromagnetic interferences, and flight control system communication network needs a rather low bandwidth. Hence, optical fiber are not needed, nevertheless this could give some more installation margin.

5. HUMAN FACTOR IN FLIGHT CONTROL DEVELOPMENT

Since Human Factor is identified as important as a contributive factor in accidents and incidents[20], Airbus flight control system takes it into account in its process development.

This issue is extensively addressed by the aviation regulation with respect to aircraft stability and control and related issues (warning, piloting aid). Maintainability is also addressed in broad terms.

Airbus flight control system offers piloting aids such as flight envelope protections, some of them are available on non fly-by-wire airplane while others are specific, along with maintainability helping devices. Note that errors introduced by the designers are addressed in §3.

5.1 Human Factor in design development

The automation in Airbus fly-by-wire contributes to safety enhancement by reducing the crew workload, the fatigue, and providing situation awareness and a better survivability to extreme situations, not to mention better robustness to crew error.

5.1.1 Comfort

One of the constraints to optimize the control laws is the crew and passengers comfort, in order not to have too much oscillations or excessive G-load factor variation[8-10].

This optimization contributes to mitigate crew fatigue[21].

5.1.2 Situation awareness

The Airbus flight control system provides also information to the crew, in order to increase his situation awareness to an adequate level. On top of this information, the aircraft systems can provide warnings, with aural and visual cues.

The information displayed on PFD / FMA / ECAM (such as which AP mode is engaged or the stall speed indication on speed scale or the status of flight control on ECAM page) provide tools to the crew to interpret the situation and to maintain him in the automation loop (crew is not excluded of the aircraft control).

Another level of information is the warnings (visual or audio). Flight control system provides the necessary information to the Flight Warning Computer.

For instance, the T.O. CONFIG memos allow checking the good configuration of the aircraft before take-off (spoiler retracted, flap / slat in take-off configuration, etc.).

5.1.3 Reconfiguration

The auto-diagnostic of a failure and the automatic reconfiguration after this failure (see paragraph 2.2.3) contributes to reduce the crew workload.

For instance, in case of a servo-control control loss, the failure is automatically detected by monitoring of discrepancy between feedback loop and command loop. Then, the redundant servo-control of the impacted surface takes over from the failed one, with a totally transparency for crew.

5.1.4 Specific flight envelope protection

Several avionic equipments are already dedicated to flight envelope protection, providing information to the crew as:

- Audio alert on Traffic Collision Avoidance System (TCAS) in case of collision risk with another A/C, on Terrain Avoidance Warning System (TAWS) in case of terrain collision risk but also in case of too excessive sink rate.
- Situation awareness on meteorological radar with the display of storming area on Navigation Display.

The electrical flight control system contributes also to the safety enhancement of the aircraft through the set of protections[8, 22], which is an integral part of the flight control laws.

Structure protections are provided during normal flying (extreme G-load factor, excessive speed).

Another protection, called high angle-of-attack, prevents the aircraft from stalling. Airbrakes are also set to 0° in case the pilot commands full thrust on the engines or flight a high angle of attack regime.

These protections lighten the pilot's workload, in particular, during avoidance manoeuvres whether for an obstacle (near miss) or windshear. A pilot who must avoid another aircraft can concentrate on the path to be followed without worrying about the structural limits of the aircraft or a possible stall.

5.2 Human factor in maintainability

Electrical flight control system uses sensors all over the aircraft and inside the actuators. As a side effect, most system failures are readily detectable and a rather precise diagnostic can be done. Thus, hundreds of precise maintenance messages are targeting the exact Line Replaceable Unit.

This contributes to decision-making in case of a failure; by crew if a dispatch is proposed in MEL document, by maintenance team otherwise.

The flight control system is designed to propose the maximum of availability.

5.3 Human Factor in certification

The aviation rules (in particular FAR/JAR 25.1302) have been reviewed for A380 to put emphasis on the human error impact in system failure.

Through this new rule, the flight control design will be demonstrated to be adequate to the effects of crew errors, to the workload, and to provide an adequate feedback to the crew on aircraft situation.

That means that the flight control design, the interface with crew, the procedures in case of failure (Flight Crew Operating Manual - FCOM) and the training are adapted:

- Not to increase the crew workload
- To provide safety barriers which prevent a single human error to transform a minor or major failure into catastrophic failure.

5.4 Challenges and trends

A difficulty has been to fine-tune all the failure detection mechanism. A basic Airbus fly-by-wire choice is to prefer immediate failure detection by on-line monitorings to off-line tests during scheduled maintenance. This reduces the level of hidden failure when the aircraft is dispatched. Unfortunately, this can be a burden to the operator when such a monitoring is too "talkative". Challenge is thus to get that all these monitorings be perfectly matured when the airplane enters into service.

The trend is also to more integrate the system, to have more interaction with avionics systems and all surveillance systems. For instance, flight control system could automatically react to a collision risk, better control could be provided on ground[23].

On certification point of view, the Human Factor Working Groups have also proposed some recommendations on Airworthiness rules FAR/JAR 25.1301 and 25.1302, specifically on:

- *Error-tolerance:* The objective is to explicitly address design-related pilot error, to make errors detectable and reversible. The error effects must be apparent for flight crew.
- *Error-avoidance:* This rule would formally address design characteristics that lead to or contribute to error. For instance, the controls and system logic required for flight crew tasks must be provided in accessible usable and unambiguous form and must not induced pilot error. The integration within systems must also be addressed.

Airbus cockpits are already designed this way; the new rule adds formalism in the exercise.

6. CONCLUSION

Experience has shown that Airbus fly-by-wire is safe, and even features safety margins. Research has also shown that new technologies can be both cost effective and provide additional safety margins. Such technical improvements, when mature, are incorporated in aircraft design, such as Electrical Actuation on A380 and A400M.

Acknowledgements

The authors are indebted to Dominique Brière for continuous guidance and inspiration.

REFERENCES

1. D. Brière, and P. Traverse, *Airbus A320/A330/A340 electrical flight controls – a family of fault-tolerant systems,* Proc. 23rd IEEE Int. Symp. On Fault-Tolerant Computing (FTCS-23), Toulouse, France, pp. 616-623 (1993).
2. D. Brière, and P. Traverse, *Airbus electrical flight controls – a family of fault-tolerant systems,* Proc. RTO/SCI Symp. on Challenges in Dynamics, System Identification, Control and Handling Qualities for Land, Air, Sea and Space Vehicles, Berlin, Germany, RTO-MP-095, paper 29 (2002).
3. Topical Days on *Fault Tolerance for Trustworthy and Dependable Information Infrastructure,* IFIP World Computer Congress, Toulouse, France, Kluwer, (2004).
4. FAR/JAR 25, *Airworthiness Standards: Transport Category Airplane,* published by FAA, title 14, part 25, and *Certification Specifications for Large Aeroplanes,* published by EASA (former JAA), CS-25.
5. A. Avizienis, J.C. Laprie, and B. Randell, *Fundamental Concepts of Dependability,* LAAS report no. 01-145 (2001).
6. D. van den Bossche, *EHA application to commercial transports – the Aerospatiale approach,* Proceedings on conference on Recent Advances in Aerospace Hydraulics, Toulouse, France (1998) and *More electric control surface actuation,* Proceedings of Royal Aeronautical Society conference on More-Electric Aircraft, London, UK, 2004.
7. J. Farineau, *Lateral electric flight control laws of a civil aircraft based upon eigen structure assignment technique,* Proc. AIAA Guidance, Navigation and Control Conference, Boston, MA, USA (1989).
8. C. Favre, *Fly-by-wire for commercial aircraft: the Airbus experience,* International Journal of Control, vol. 59, No. 1, pp.139-157 (1994).
9. F. Kubica, T. Livet, X. LeTron, and A. Bucharles, *Parameter-robust flight control system for a flexible aircraft,* Control Engineering Practice, Vol. 3, No. 9, pp.1209-1215 (1995).
10. T. Livet, D. Fath, and F. Kubica, *Robust autopilot design for a highly flexible aircraft,* Proc. 13th IFAC World Congress, Vol. P, San Francisco, CA, USA, pp.279-284 (1995).

11. ARP 4754/ED79, *Certification Considerations for Highly-Integrated or Complex Systems,* published by SAE, no. ARP4754, and EUROCAE, no. ED79 (1996).
12. DO178B/ED12, *Software Considerations in Airborne Systems and Equipment Certification,* published by ARINC, no. DO178B, and EUROCAE, no. ED 12, 1992.
13. DO254/ED80, *Design Assurance Guidance for Airborne Electronic Hardware,* published by ARINC, no. DO254, and EUROCAE, no. ED80 (2000).
14. F. Randimbivololona, J. Souyris, P. Baudin, A. Pacalet, J. Raguideau et D. Schoen. *Applying Formal Proof Techniques to Avionics Software: A pragmatic Approach.* FM99. LNCS 1709, Vol II.
15. S. Thesing, J. Souyris, R. Heckmann, F. Randimbivololona, M. Langenbach, R. Wilhelm, and C. Ferdinand, *An Abstract Interpretation-Based Timing Validation of Hard Real-Time Avionics,* Proc. Int Conf. on Dependable Systems and Networks (DSN) (June 2003).
16. P. Cousot. *Interprétation abstraite,* Technique et Science Informatique, *Vol. 19, Nb 1-2-3.,* Hermès, Paris, France, pp. 155-164 (2000).
17. B. Blanchet, P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival.*A Static Analyzer for Large Safety-Critical Software,* Proc. PLDI 2003 - ACM SIGPLAN SIGSOFT Conf. on Programming Language Design and Implementation, Federated Computing Research Conference, San Diego, CA USA, pp. 196-207 (2003).
18. E. Goubault, M. Martel, and S. Putot. *Asserting the Precision of Floating-Point Computations: a Simple Abstract Interpreter* (Demo Paper), ESOP'2002, LNCS.
19. ARP 4761, *Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems,* published by SAE, no. ARP4761 (1996).
20. *Human Factors for Civil Flight Deck Design,* published by Ashgate (2004).
21. I. Lacaze, *Prise en compte du confort vibratoire dans la conception,* Paris V University Report (2002).
22. D. Chatrenet, *Les qualités de vol des avions de transport civil à commandes de vol électriques,* Proc. AGARD Conf. on Active Control Technology, Turin, Italy, AGARD-CP-560, paper 28 (1994).
23. J. Duprez, F. Mora-Camino and F. Villaume, *Robust control of the aircraft on ground lateral motion,* Proc. 24th ICAS Conf., Yokohama, Japan (2004).