

System Dependency Analysis as a Common Cause Search Engine for Complex Aircraft Systems

Klaus Fritz

Copyright © 2009 SAE International

ABSTRACT

The system dependency analysis for complex aircraft systems is a model-based methodology and tool for analyzing availability and minimum acceptable control requirements for failures or event scenarios to support the aircraft and system safety analyses (SAE ARP4761) required to show compliance to 14CFR/CS §25.1309, §25.671 and other, related requirements. Aspects of the system such as functional interaction and dependencies to supply systems, physical items (equipment, wiring and tubing) and installation aspects are included in the analysis.

This paper describes additional steps to enable the search for potential common cause failure conditions for the system of interest or airplane level systems based on the system model. Common cause analysis (CCA) procedures using the system dependency analysis rely on a systematic and checklist-based approach to determine potential common cause failure conditions. The model-based approach of the system dependency analysis can be converted to search for critical functional failure combinations and therefore for sensitive installation areas and potential common mode failure conditions.

This paper is based on the system dependency analysis described in the SAE papers 2007-01-3852 and 2009-01-3107 which provide the modeling approach and the analysis of system dependencies to support the aircraft and system safety analyses including examples for the common cause analyses. An example of a quad-redundant flight controls system for a regional jet aircraft serves to show the system model and the analysis.

INTRODUCTION

SYSTEM DEPENDENCY ANALYSIS – The system dependency analysis provides a model-based systematic approach to determine fault propagation within supply systems and sub-systems up to the main system functions. It is based on the general principle to detect erroneous behavior near the failure source with subsequent fail-safe reaction deactivating the affected function. Deactivation may occur on any functional level depending on severity of the fault leading to a "loss of function". The function loss either propagates through the system, e.g. loss of an electric power supply leads to loss of a control function, or does not propagate if redundancy is available, e.g. a dual redundant power input prevents loss of a control function after loss of one power input.

For aircraft level analyses, the system model needs to consider end-to-end functionality, e.g. from the system sensors to the effective control actuation, as well as the integration with supply systems needs to consider source-to-end, i.e. from the power generation to the supply of power to control and actuation. All functions and the links between the functions are being considered. In addition, the system model includes the physical representation of the functions and links, such as equipment, wiring, hydraulic tubing or mechanical parts as well as their specific installation into the aircraft. To allow determination of main function availability with failures present, performance criteria (minimum acceptable control criteria or MACC) are also integrated into the system model. A performance criterion may, for example, determine the capability of continued safe flight and landing of the aircraft after occurrence of a failure condition.

The inclusion of the integration aspects (supply systems/functions), the physical implementation and the installation aspects into the system model allows not only to support the single and multiple failure analyses, but also provides capability to analyze event scenarios to be analyzed for the common cause analyses (e.g. PRA, ZSA and CMA) as per SAE ARP4761. Those typically create event scenarios to be analyzed for their functional effect, i.e. multiple failures of physical items, such as equipment, wiring bundles, hydraulic tubing or mechanical parts, of the system itself, but also of the supply systems. This paper provides examples for using the system dependency analysis especially for the common cause analyses.

MODELING TECHNIQUE – The system model for the system dependency analysis integrates all model items, representing functions, sub-functions, function links, performance criteria (MACC) and all physical items including their installation aspects equally into a database structure independent of their hierarchical level. The model for each item consists of a list of the associated dependencies and an analysis failure injection. The model result function combines the status of all dependencies and the fault injection, i.e. a model function is available, if all inputs to the function are available and if it is not set failed itself. The fault injection is used to set failure conditions for the analyses performed on the system model. The positive-fail logic uses "1" or "TRUE" representing a function "failed" and uses "0" or "FALSE" representing a function "ok" or "available". The result function may include a logic "AND"-function to represent redundancies. The logic "OR"-function is used to list dependencies leading directly to loss of the function. By integrating the model items into a system model, the database structure using lookup functions automatically connects the items to an overall system function "network" enabling visualization of the failure propagation.

The main aspects include, but are not limited to, the functional, the physical implementation and the installation levels (see figure 1). The modeling technique allows introduction of as many hierarchical levels and sub-levels as required to perform the analyses and to ensure a structured modeling approach. This technique also allows grouping of physical items and installation aspects for analysis purposes.

ANALYSIS TECHNIQUE – By extending the system model database with an analysis management connected to the model items failure injection, the system model can be subjected to various sets of failure conditions. The analysis management stores the fault injection for the individual analysis cases and scenarios and logs the associated results. The analysis management enables the use of a single system model for a wide variety of analyses and analysis cases. Storing the results allows documentation in order to support the traditional suite of safety analyses, but also enables determination of the impact of changes to system function, implementation, integration or installation without manually repeating all individual analyses.

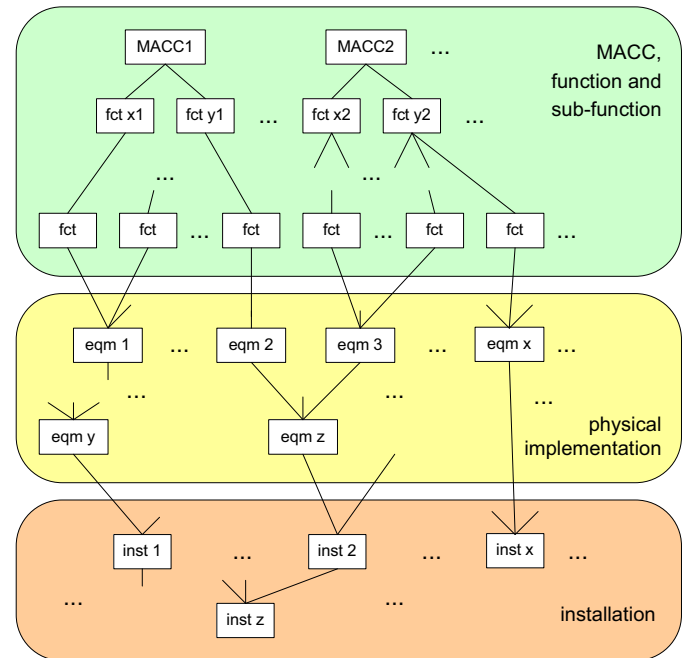


Figure 1: System Model Hierarchy

RESULT VISUALIZATION – The system dependency model and analysis database allows the user to visualize the failure propagation throughout the system up to the performance criteria. This is important not only for model generation, model debugging and verification, but most important to show analysis results at various level of detail. The visualization within the model items, the overall system model and optional visualization at aircraft system level allows the user to trace deficiencies and root causes in the system function architecture, especially for integration and installation aspects including the supply systems.

Resolving these issues allows the creation of integration and installation concepts and requirements to ensure sufficient availability of system resources for all failure conditions or events.

EXAMPLE SYSTEM – A typical quad-redundant fly-by-wire flight control system to control the aircraft pitch axis is shown in figure 2. This simplified system will be used as a demonstration example for the system dependency model and analysis. The aircraft pitch axis is controlled by the flight crew operating the control columns installed on the flight deck. The movement of the columns is sensed by a position sensor package (SP). The pitch control computer (PCC) uses the position of the columns to control the elevator surface actuators (PA). Position sensors integral to the actuators allow closed loop control of the pitch actuator. The pitch control function is comprised of 4 redundant control channels, with 2 actuators in an active-active configuration moving each one of the two elevator surfaces (LH and RH ES). For this example, the minimum acceptable pitch control requirement for continued safe flight and landing is fulfilled, if at least one control channel is available to operate an elevator surface. The fly-by-wire computer

has a dual redundant electrical power supply. The surface actuator is hydraulically powered.

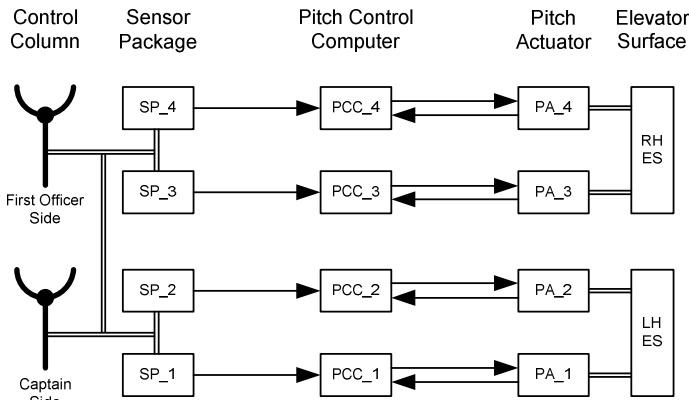


Figure 2: System block diagram of a quad-redundant pitch control function

The SAE paper "System Dependency Analysis for Complex Aircraft Systems" (2007-01-3852) describes the modeling and analysis technique in detail. An example model and analysis database implementation uses the desktop computer software Microsoft Excel © on the example fly-by-wire flight control system.

The SAE paper "System Dependency Analysis supporting Common Cause Analyses of Complex Aircraft Systems" (2009-01-3107) provides examples to use the system dependency analysis to support the common cause analyses (CCA), such as the particular risk analysis (PRA), zonal safety analysis (ZSA) and the common mode analysis (CMA) as defined per SAE ARP4761.

The examples focus on system dependency analysis methodology creating failure conditions from predetermined events, which are defined in certification regulation or industry guidance checklists, created from system/aircraft characteristics or known by experience. To engage additional value from the system model for potential, not predetermined failure conditions (failure cause not predefined) the system dependency analysis can be converted to be used as a common cause search engine for complex aircraft systems.

This paper shows the process, additional model preparation and analysis steps to use the system dependency analysis as a search engine for potential common cause hazards highlighting system or airplane level control capability for continued safe flight and landing.

COMMON CAUSE SEARCH ENGINE

The system dependency analysis as described in the previous papers cannot be directly used as a search engine for potential common cause hazards. Basis for use as a search engine is the system model which has to be refined concerning its modeling standards. The system model remains compatible with the general

purpose of the system dependency analysis, i.e. if the model complies with the search engine model demands it can be used for all purposes. Based on the same model, the search engine and the associated analysis are separate from the general analysis management.

The strategy using the system dependency analysis as a search engine is based on the following steps:

- Create or refine model (if applicable)
- Determine critical dependencies between functions
- Assign analysis subjects to the critical combinations of functions,
 - For the ZSA, assign installation aspects
 - For the CMA, assign implementation aspects
- Search for commonalities
- Evaluate acceptability

MODELING REFINEMENTS – The following aspects have to be considered for use of the system model also for the search engine:

System Breakdown – The expression "system" is defined to be "a combination of inter-related items arranged to perform a specific function(s)" (SAE ARP4761), i.e. generally a mix of functions and their physical implementation. In preparation of a suitable system modeling enabling search for potential common cause faults with the SDA, a structured breakdown of the system is required, specifically a distinction between functions, implementation and installation aspects (see figure 1). This distinction between physical items and their installation is common practice. The distinction between functions and their implementation or physical representation is commonly universalized or abbreviated. For example, a switch can be a function to select inputs, but is also used for a physical item which performs this function. For this example, the switch function is implemented using an electronic component. The switch, in terms of a physical item, is installed in a dedicated installation area within the aircraft.

Each item to be considered for the system model has to be broken down into its specific function and the associated implementation by creating 2 individual model items. In some cases, this may be controversial to efficient modeling. The functions may be hierarchically structured to as many levels required to represent the system to be analyzed.

The same applies to the links between the functions. A function link connects functions and automatically establishes a dependency between the functions, for example, transmitting a data item from one function to another. Those links, for example, may be implemented by means of a memory device if both functions are implemented in software or they may be a piece of wire installed in the aircraft or they may be a mechanical lever arm, etc. Both function and implementation should be modeled by individual items.