Madrid. 23 y 24 de noviembre de 2016



Organiza:





APPLICATION OF MBSA FOR A FLIGHT CONTROL EMA

Ricardo de Arriba

Elena García

Daniel García

RAMS & ILS Engineer

RAMS & ILS Engineer

RAMS & ILS Intern





Outline



- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion







1. Introduction

- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion





1. Introduction CESA Presentation





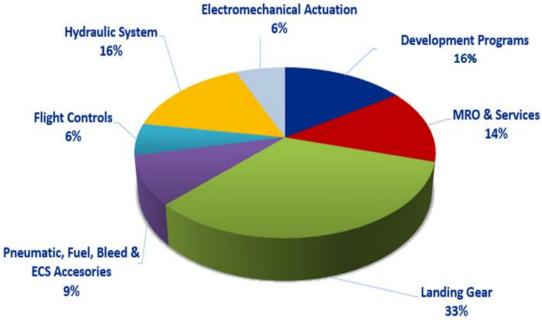
60%

40%

















1. Introduction CESA Products









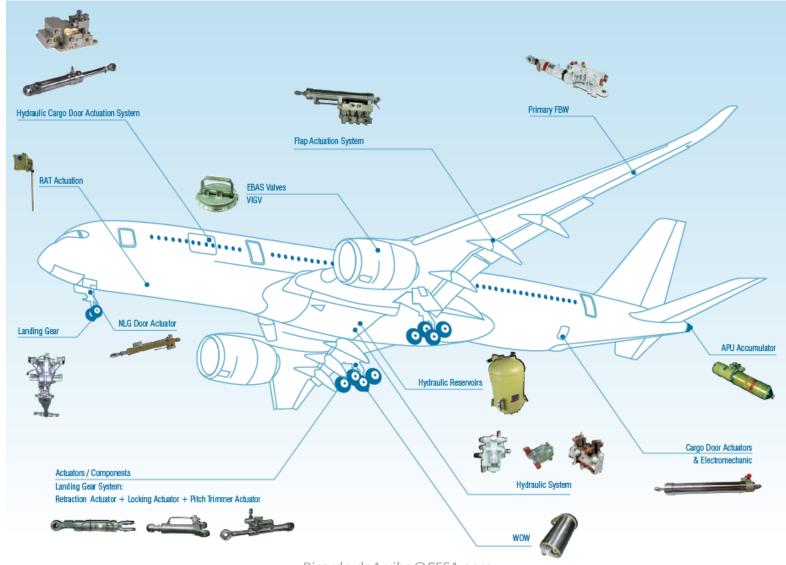










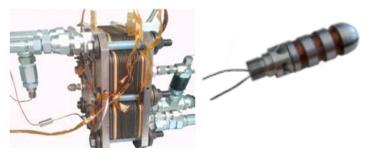


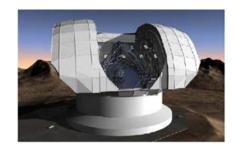
1. Introduction CESA Research & Development























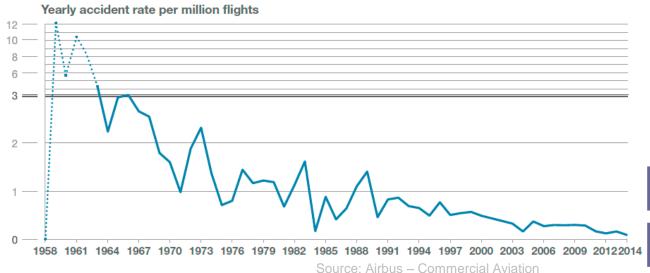






1. Introduction System Safety Assessment

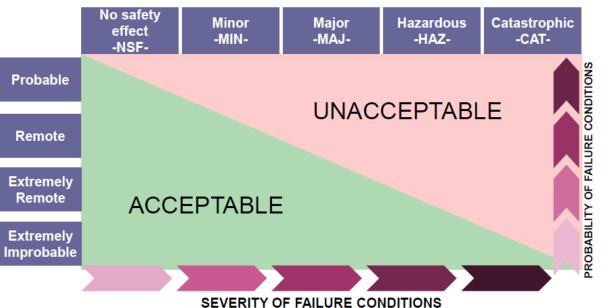




PONNINISTRATION



Accidents 1958-2014 - A Statistical Analysis



Source: Safety Assessment MASI





1. Introduction System Safety Assessment

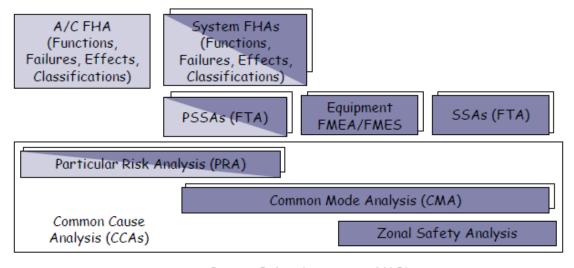


• SAE-ARP-4754

Product Validation Customer Final Needs **Product** Top Level Requirements System Integration Concept System Verification System System **Testing** Requirements Requirements Design Subsystem are validated Verification Subsystem against upper level Sub-System requirements Testing Requirements Hardware/Software Development Time

Source: www. sarel-consult.de

SAE-ARP-4761



Source: Safety Assessment MASI







2. MBSA

- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion



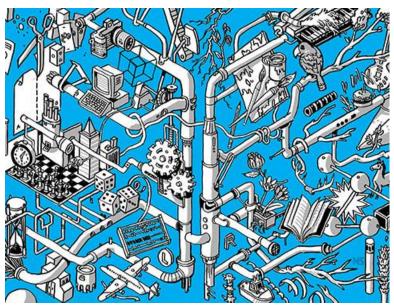


2. MBSA Why?





Classical Safety Methods



Complex Systems



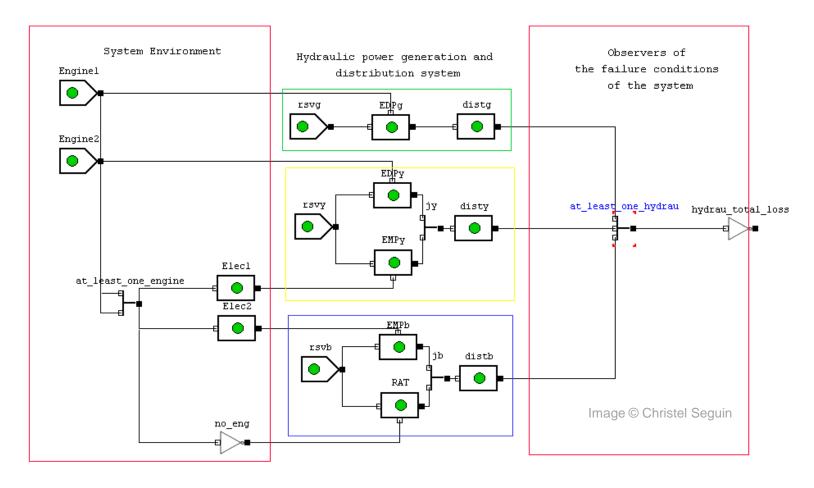
Model Based Safety Analysis





2. MBSA What is?





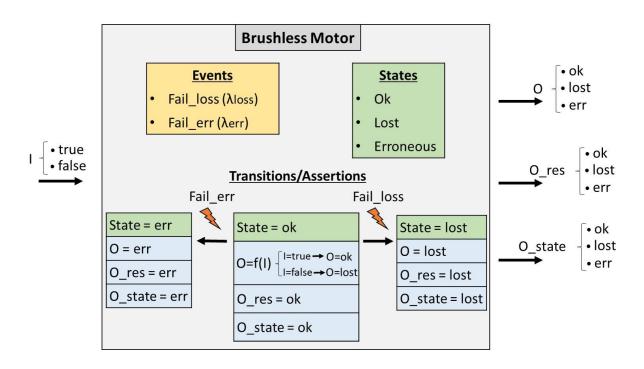




2. MBSA

Elements

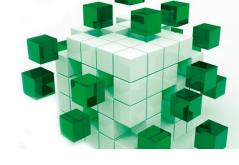




```
node block Motor
 flow
 icone : [1,3] : out ;
 O_res : COMPUTATION_COMPUTATION_FailType : out ;
                                                         Interfaces
 O : COMPUTATION COMPUTATION FailType : out ;
                                                         Declaration
 I : bool : in ;
 O state : COMPUTATION COMPUTATION FailType : out ;
 s : COMPUTATION_COMPUTATION_FailType ; --- Component State Declaration
 fail loss, fail err; ---- Faulty Events
   s := ok ; ---- Initial State value
                                                         Transitions
trans
S=ok |- fail loss -> S:=lost;
S=ok |- fail err -> S:=err;
0 = case { (I=true and S=ok) : ok,
          (I=true and S=err) : err,
          (I=false and S=err) : err,
         else lost };
                                                      Assertions
O res = case { S=ok : ok,
          S=lost : lost,
         else err };
O state = S;
icone = case { S=ok : 1,
          S=lost : 2,
          else 3);
```







3. EMA for Aileron

- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion

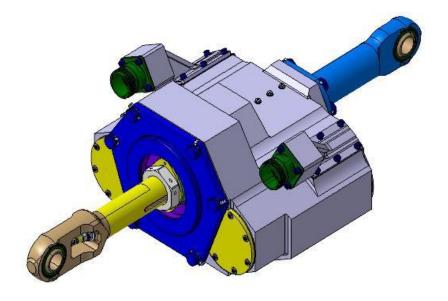


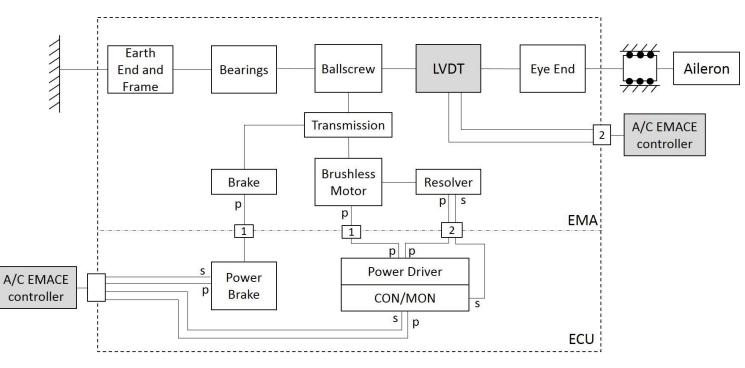


3. EMA for Aileron

Functional Description







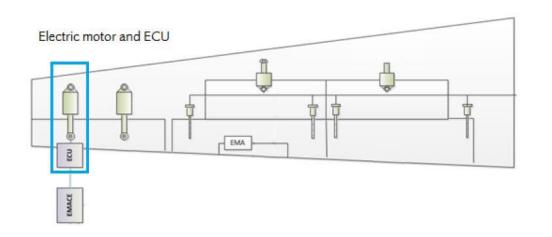




3. EMA for Aileron

Failure Conditions

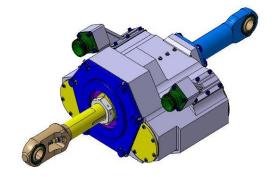






Aileron Runaway

Aileron Jamming



EMA Jamming







4. Application of MBSA for EMA Flight Controls

- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion

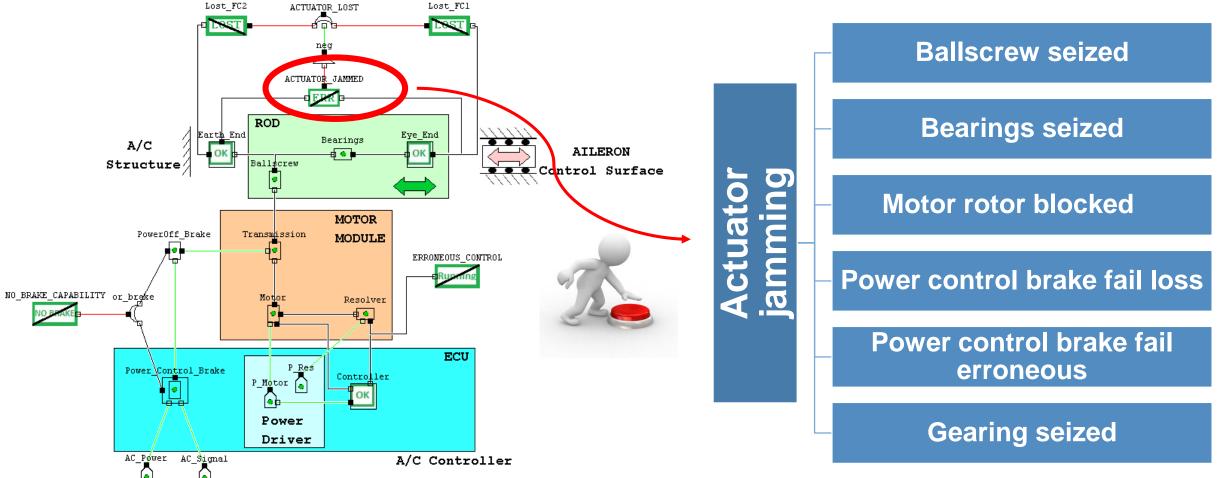




4. MBSA for EMA

Failure Conditions



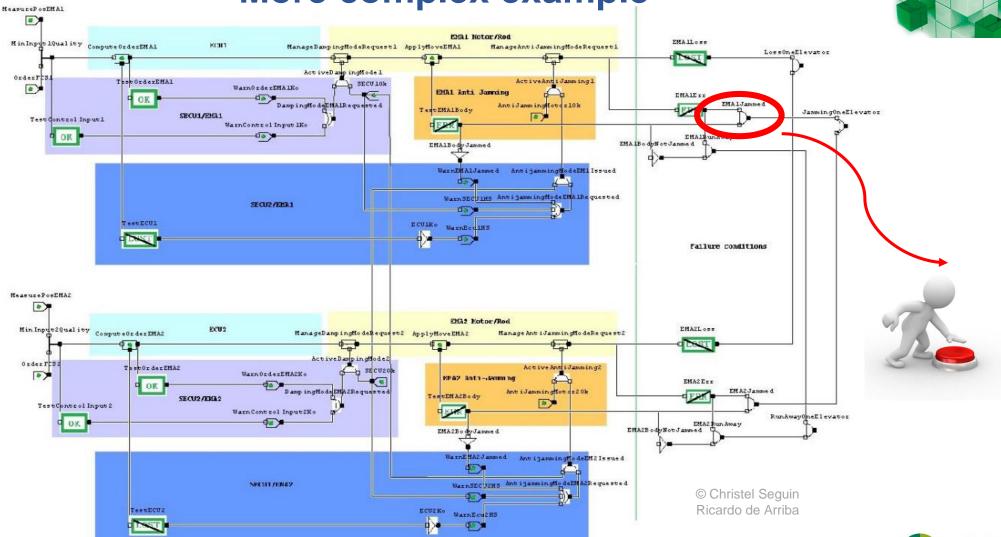






4. MBSA for EMA

More complex example





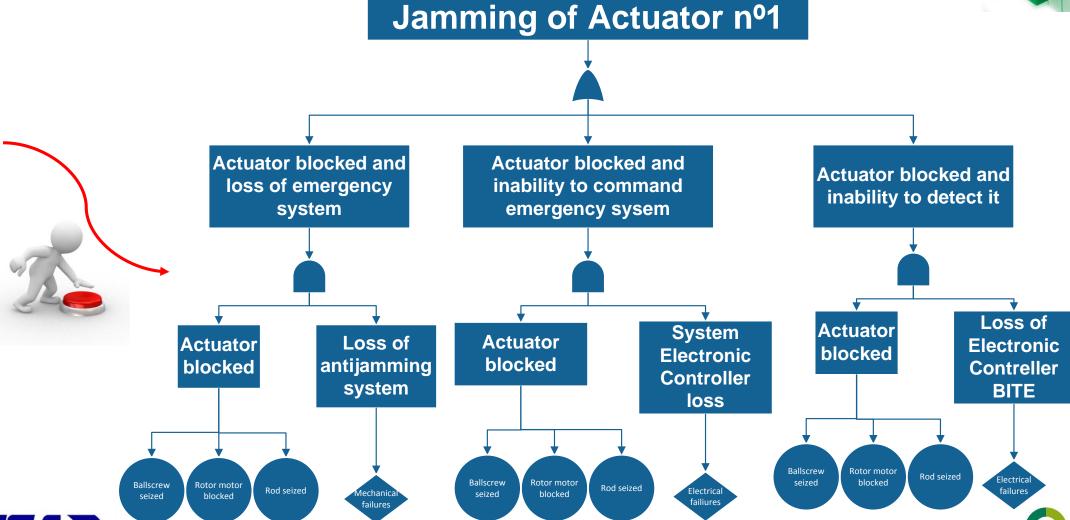


4. MBSA for EMA

More complex example



ASOCIACIÓN ESPAÑOLA PARA LA CALIDAD





5. Conclusion

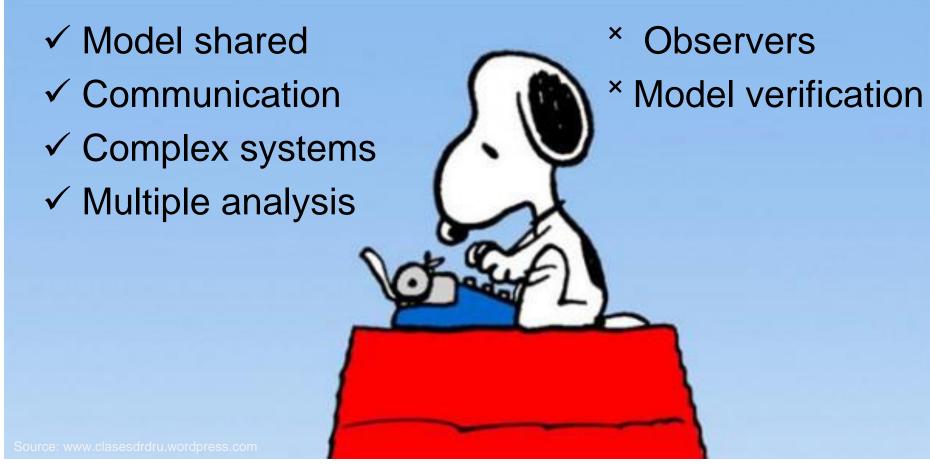
- 1. Introduction
- 2. MBSA
- 3. EMA for Aileron
- 4. Application of MBSA for EMA Flight Controls
- 5. Conclusion





5. Conclusion









Acknowledgements











Thank you!







