Rapid Deployment of Aerospace Flight Controls

Edward L. Burnett

Lockheed Martin Aeronautics Company - Palmdale

Lockheed Martin Senior Fellow

Modeling, Simulation and Controls

Presented to:

The MathWorks Aerospace and Defense Conference

14 June, 2006

Rapid Deployment of Aerospace Flight Controls

- The Historic Problem
- A Possible Solution
- A Case Study
- Summary

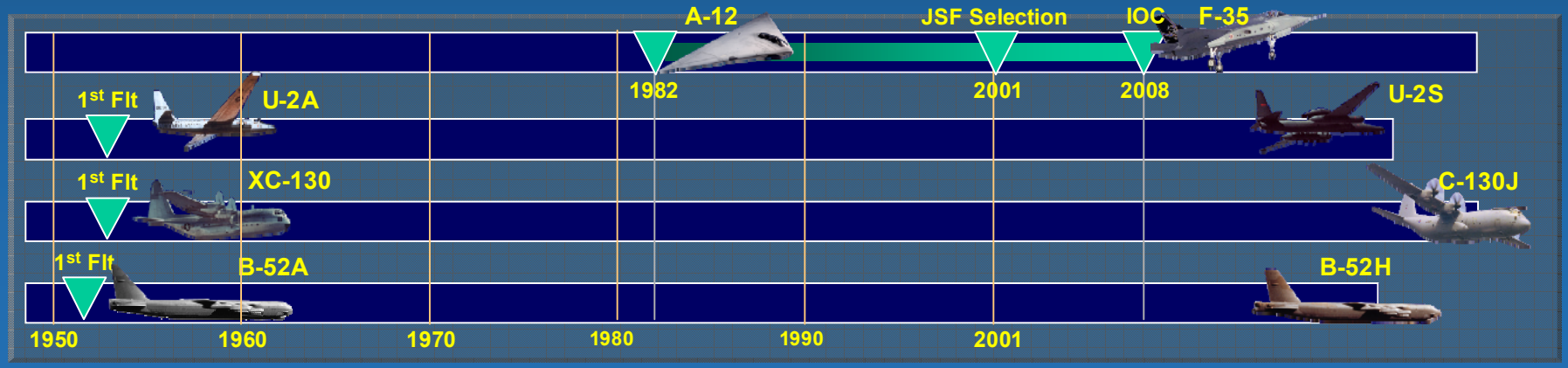
The Problem

- **Augustine's Laws**
  - ***(Law XVI)***
    - "In the year 2054, the entire defense budget will purchase just one aircraft"
  - ***Software's Part (Law XVII)***
    - "Like Entropy"
      - *"Weight nothing"*
      - *"Obeys the 2nd Law of Thermodynamics; i.e., its always increasing"*

The Problem

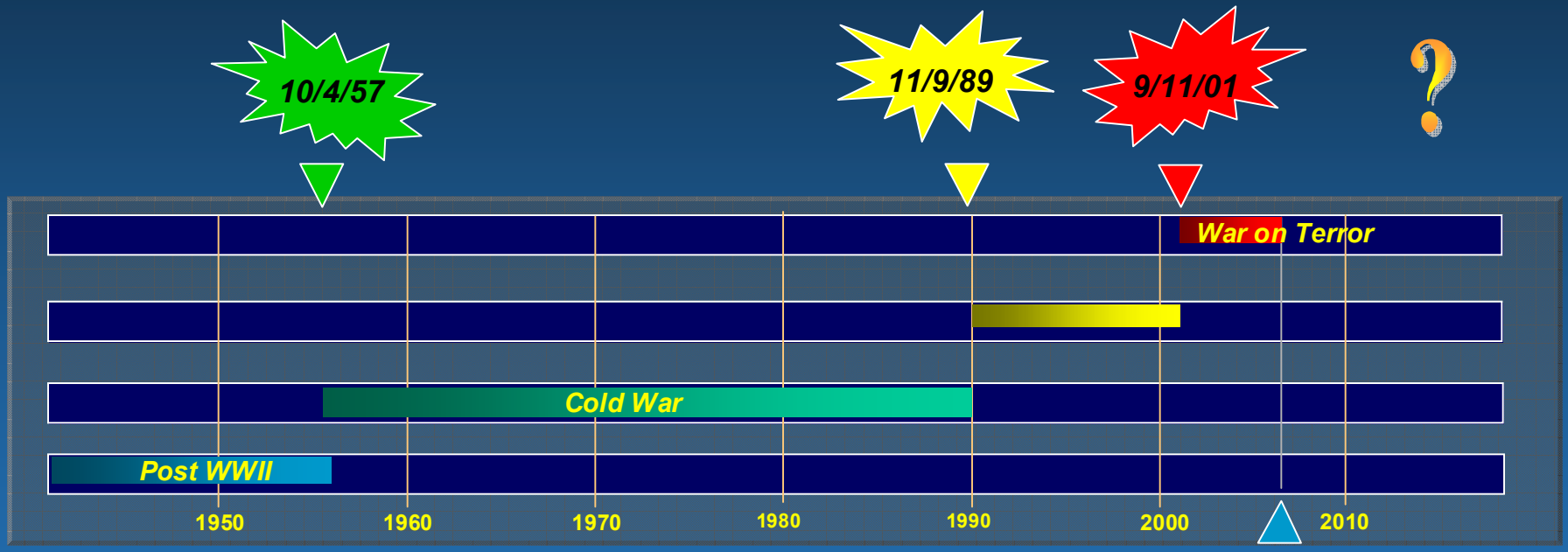
Major Acquisition Process

Airframe Service Life



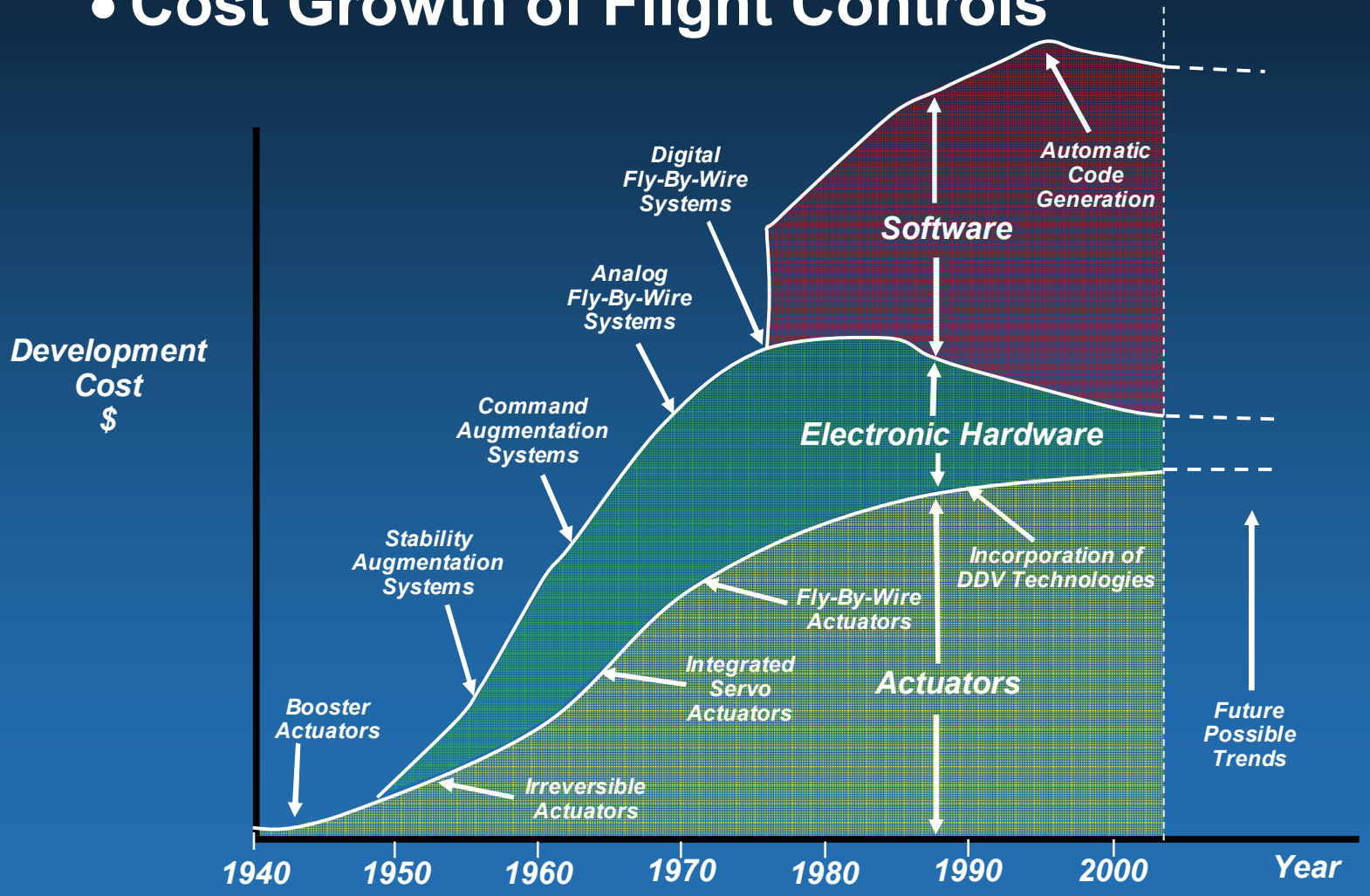
The Problem

- *However, We Live in a Rapidly Changing World*



The Problem

- Cost Growth of Flight Controls

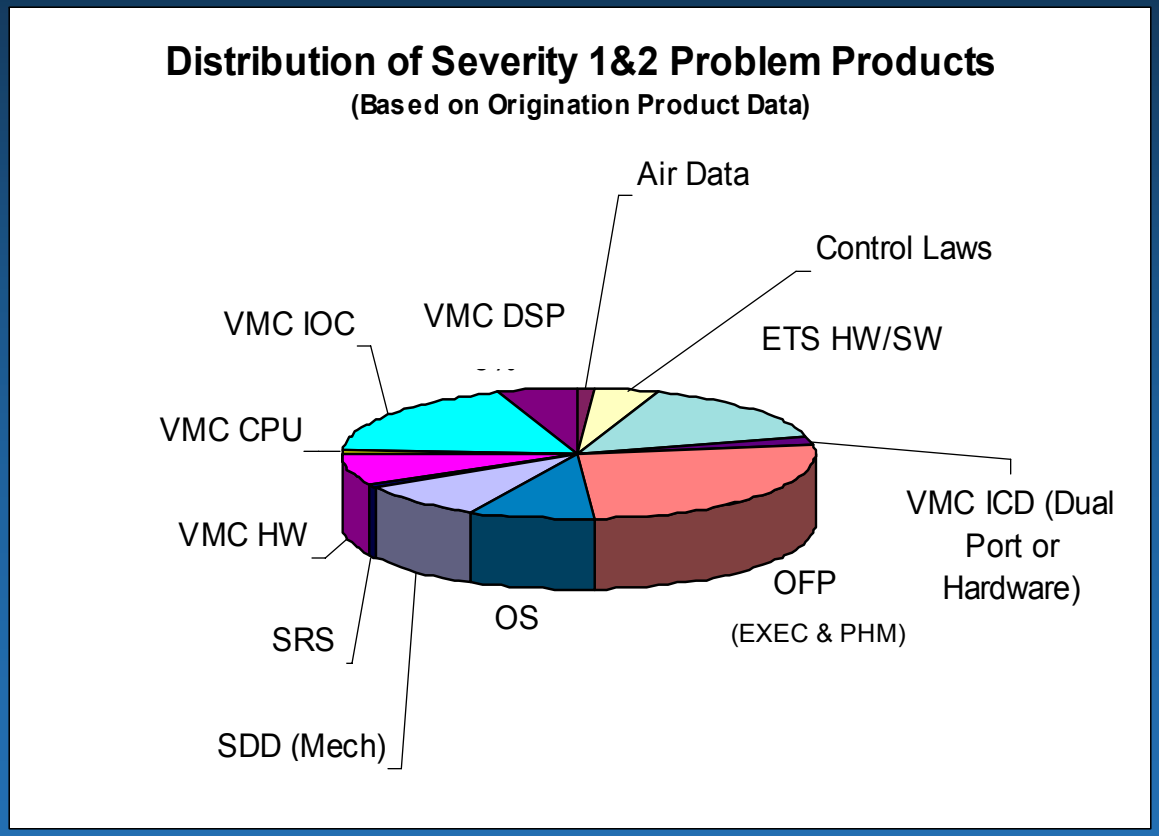


The Problem

- **Integration Issues**
  - ***Interface Control Documents (ICDs)***
    - Variable Name
    - Data Type
    - Units
    - LSB, MSB
    - Update Rates
    - …

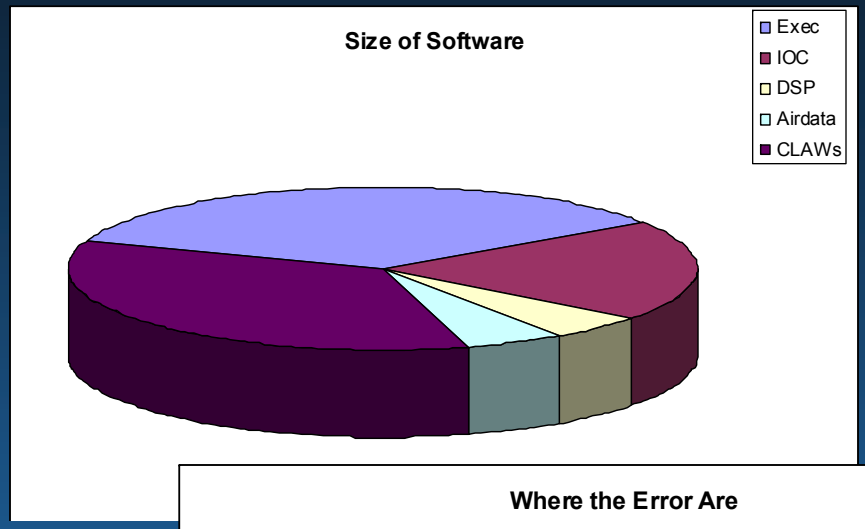
The Problem

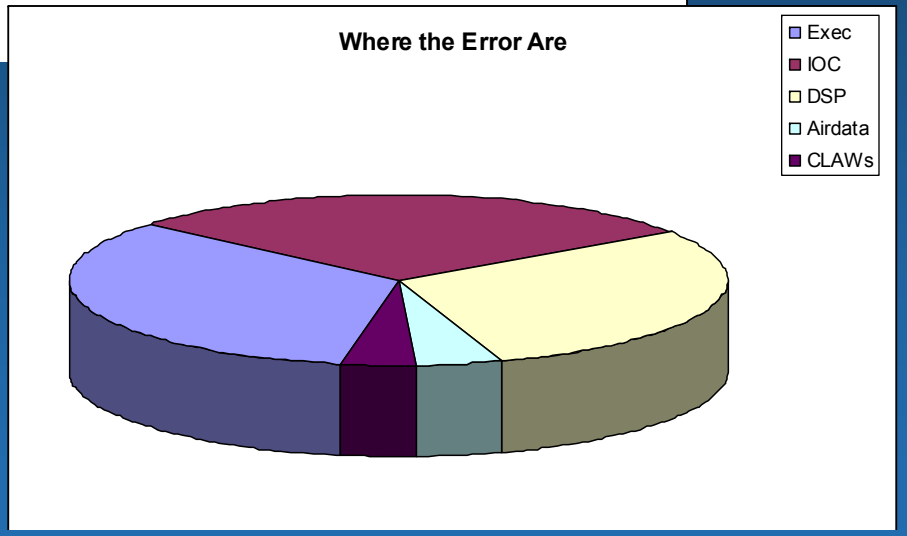
- **Looking For Trends in History**



The Problem

- **Software Only**
  - ***Exec, IOC, DSP***
    - Classical
  - ***CLAWS, Air Data***
    - MBD

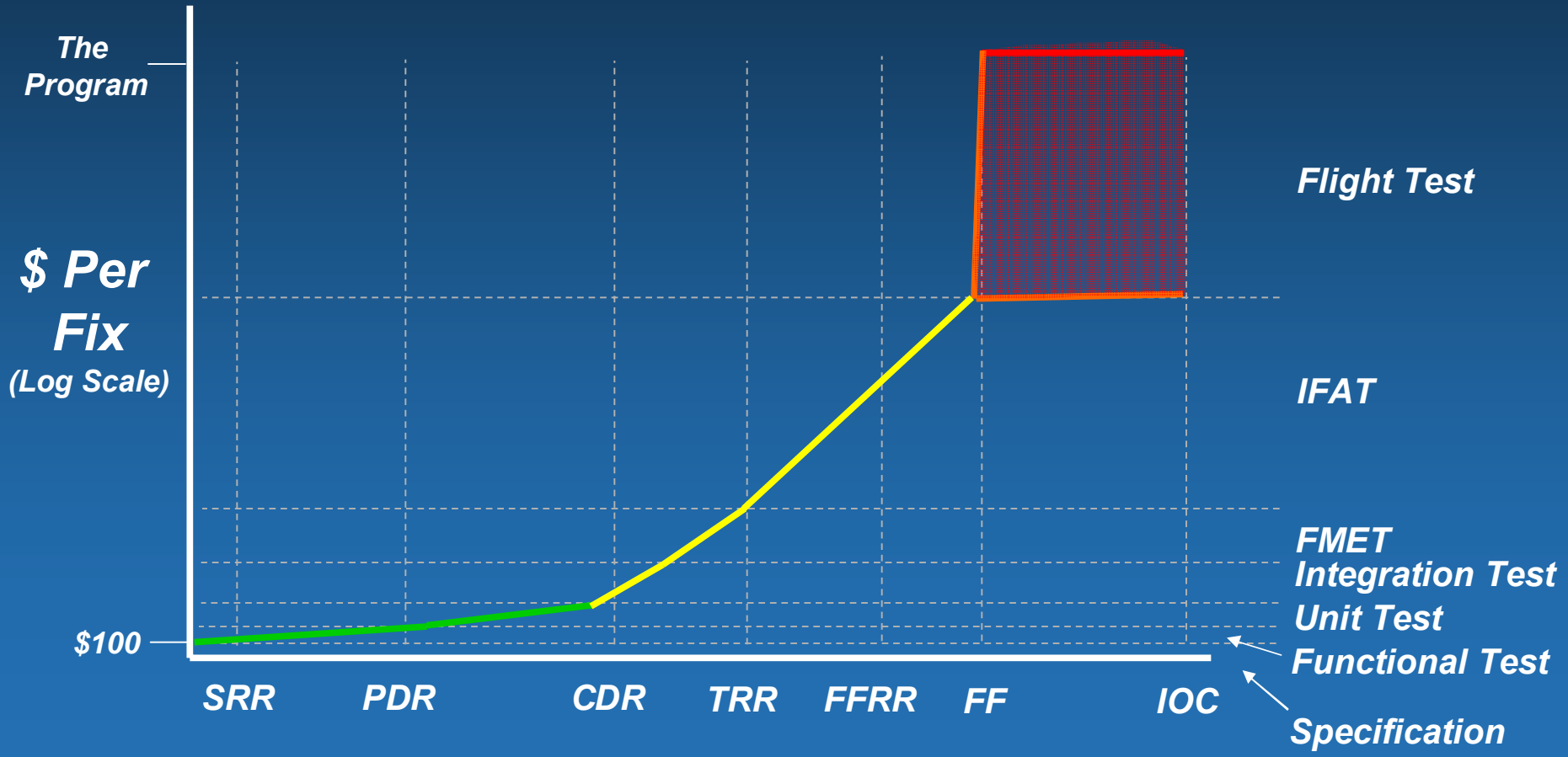




Factor of 7 Times More Errors From Classical Process

The Problem

- Cost of Software and ICD Fixes



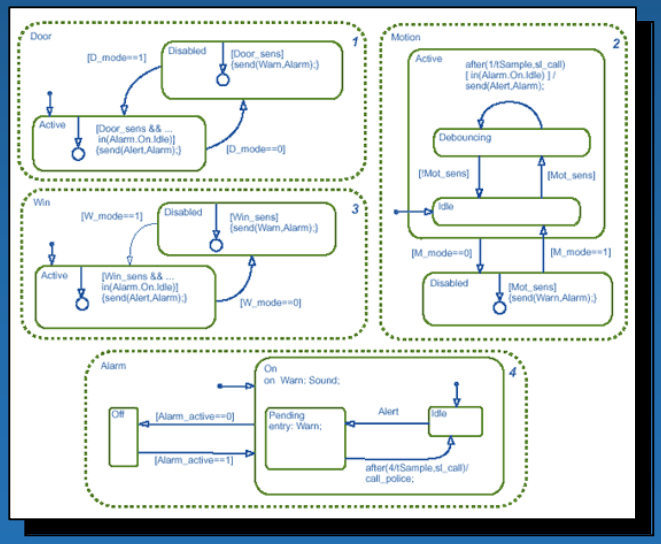
The Problem

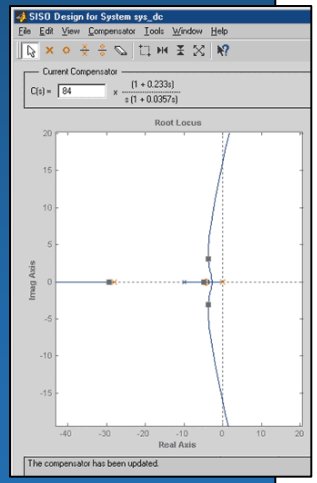
- **New Challenges –**
  - ***FAA***
    - Reliability
      - *Space and Time Partitioning*
    - UAVs in the NAS
      - *See and Avoid*

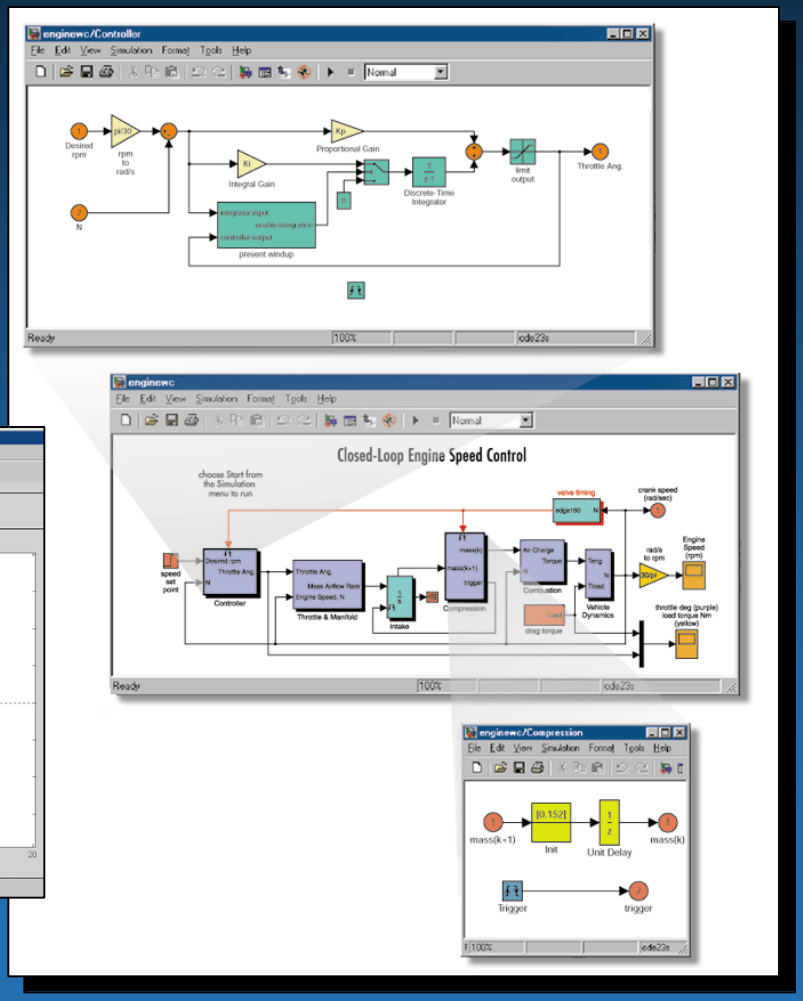
A Possible Solution

A Possible Solution

- **Model Based Design +**
  - *Design*
  - *Analyze*
  - *Deploy*
  - *Integrate*
  - *Test*
  - *Document*

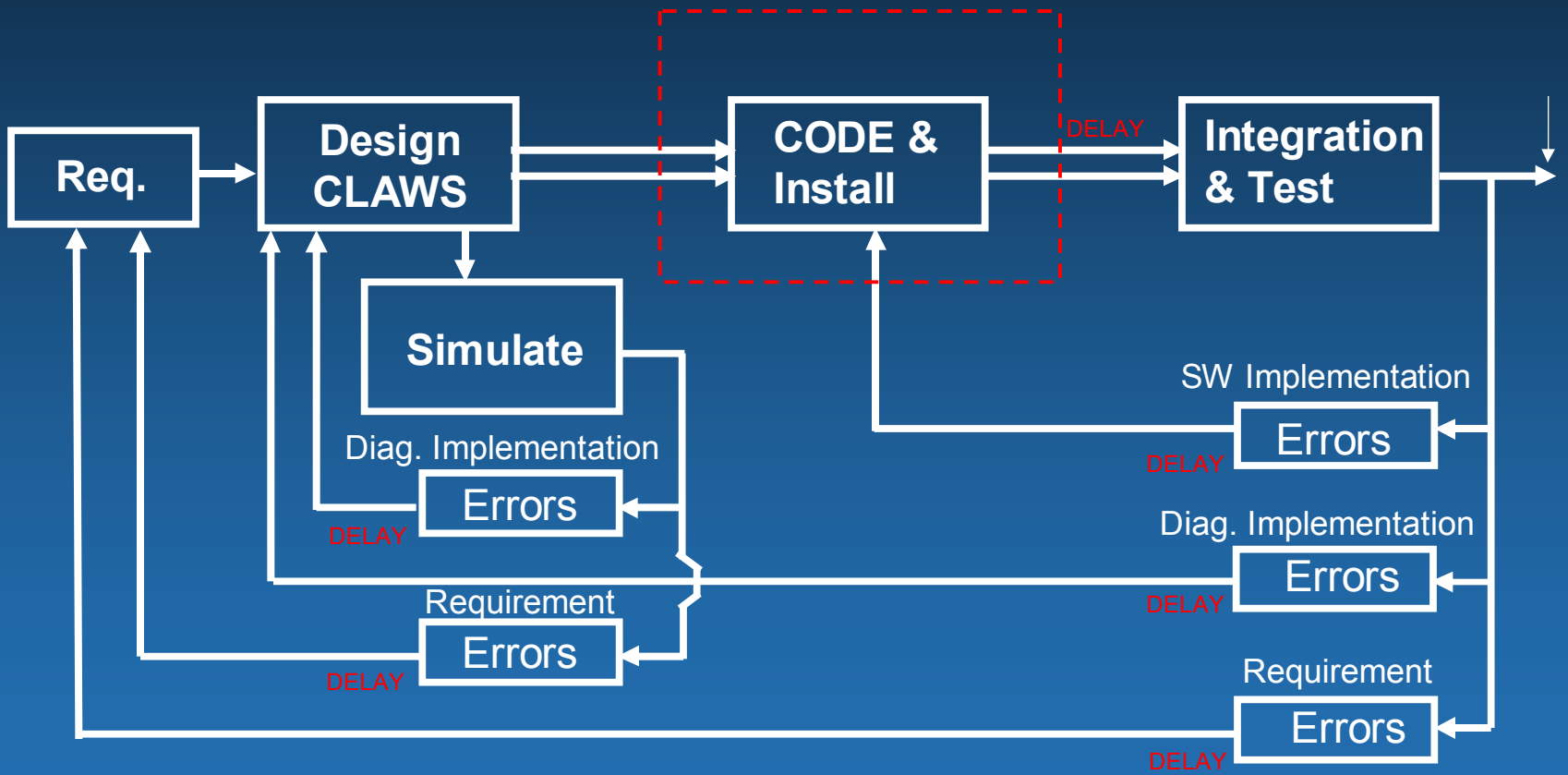






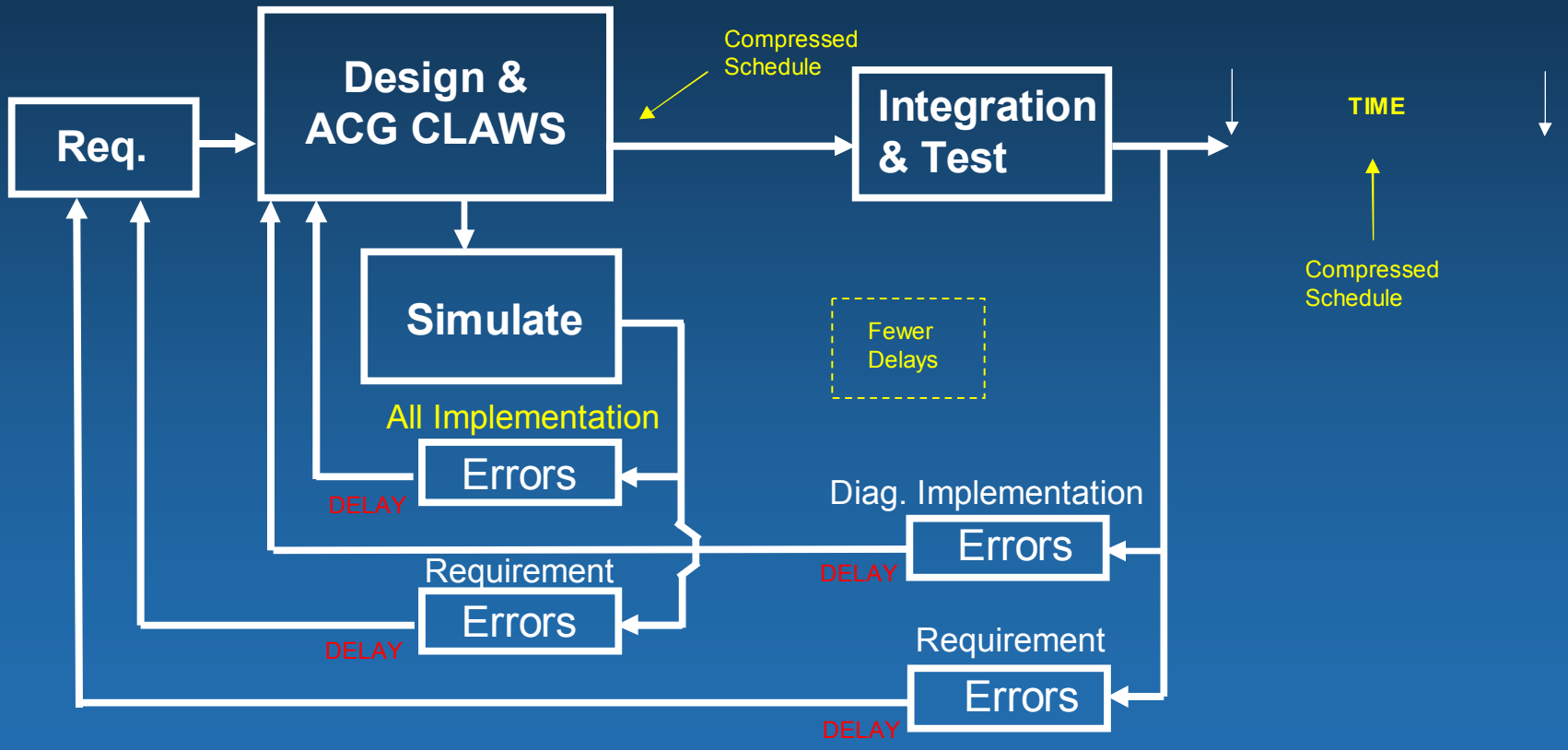
A Possible Solution

- **Classical VMS System Development Process**



A Possible Solution

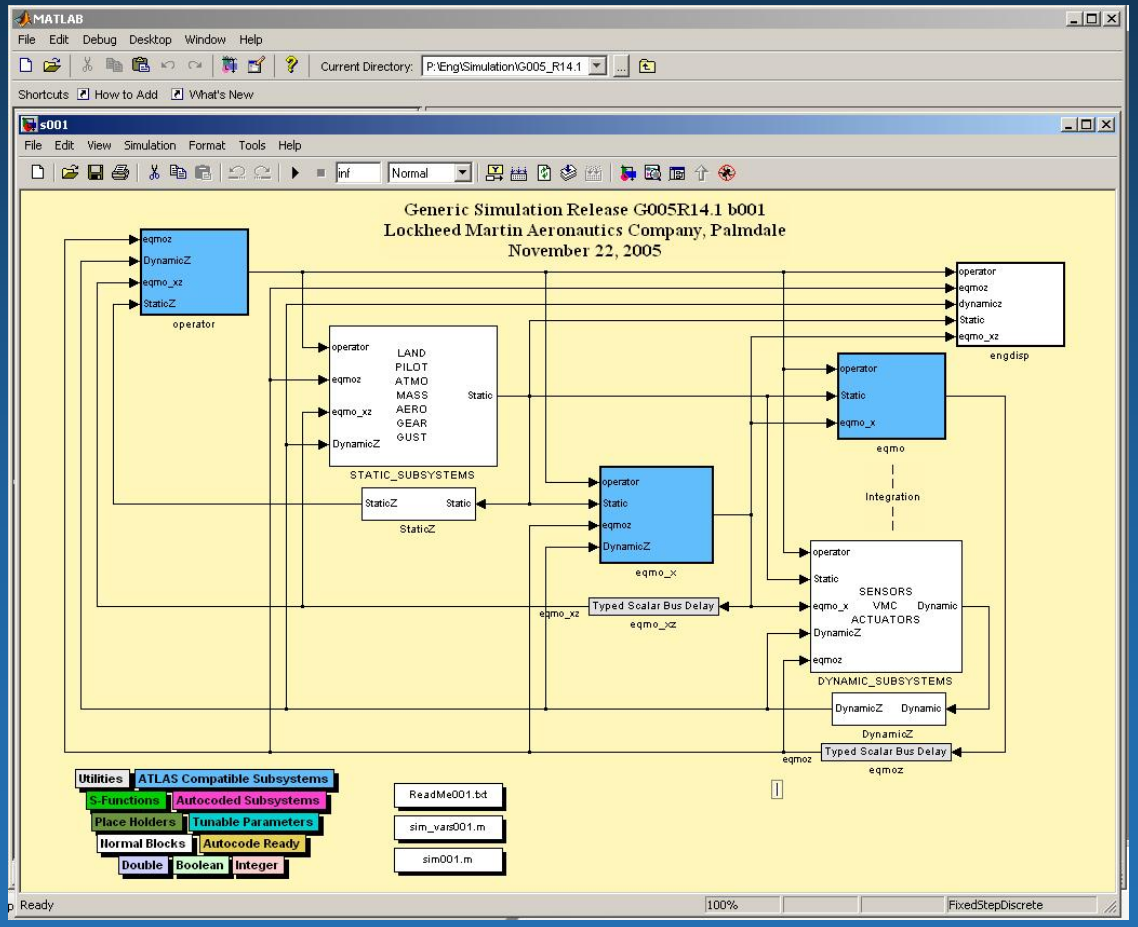
- **Model Based Design VMS System Development Process**



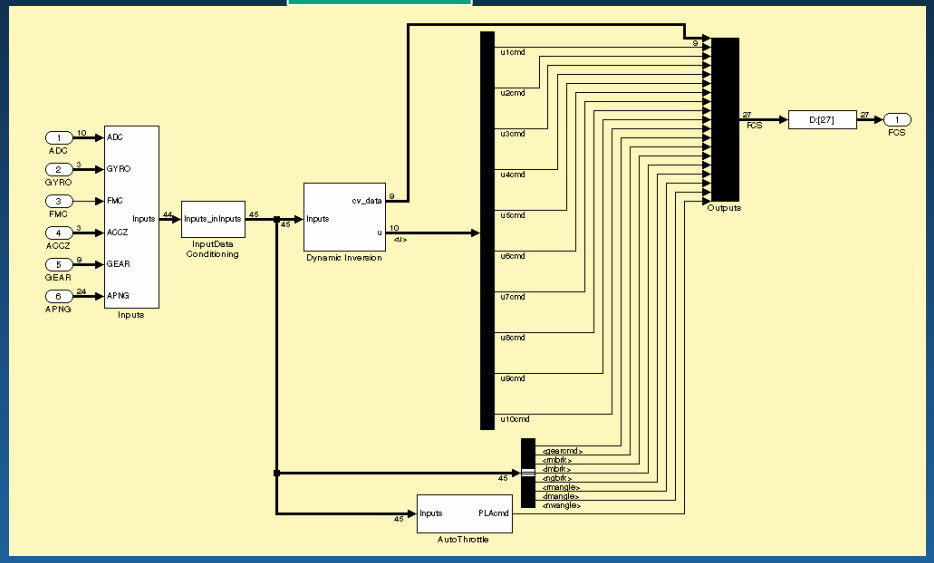
ACG = Automatic Code Generation

A Possible Solution

- **Model Drawing Standards**
  - ***The Enabler***
    - Learning Curve
    - Model Reuse
    - Scripts
    - Tools
    - Automated Test



A Possible Solution



Automatic Code Generation:

Template Files, Scripts

Create C-code and wrappers

FCS_fwrp.f

FCS_wrp_.c

FCS.c

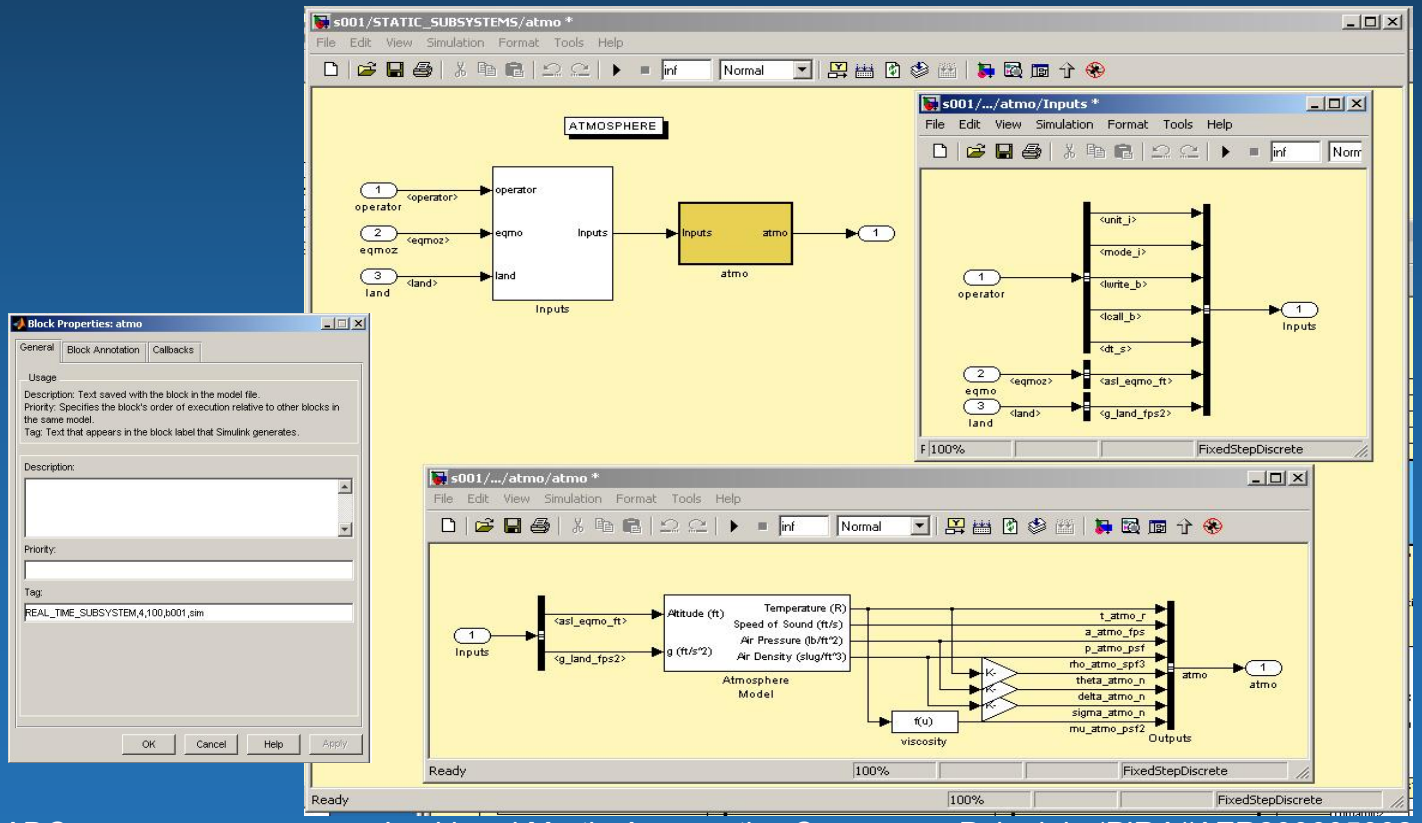
```
/* Real-time model */
RT_MODEL_s002fcs s002fcs_M_;
RT_MODEL_s002fcs *s002fcs_M = &s002fcs_M_;

void sf_m0_c25_s002fcs(void)

{
  int32_T i;
  int32_T ok;
  int32_T j;
  real_T temp;
  i = 0;
  ok = 0;
  j = 0;
  temp = 0.0;
  j = 0;
  s002fcs_B.ilimit_c = 1;
  s002fcs_B.scale_c = 1.0;
  while(j < s002fcs_B.neff) {
```

A Possible Solution

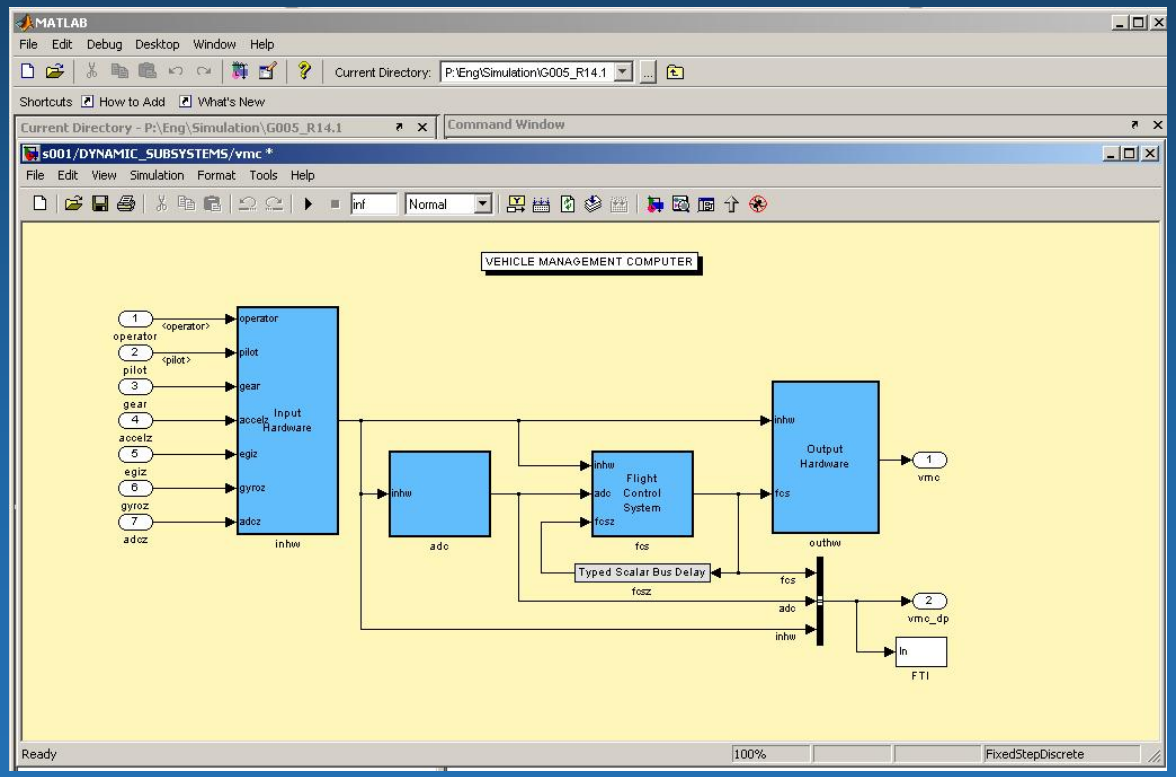
- **Use of Buses and Property Tags**
  - ***Model Based ICD***
  - ***Script Testing of Interfaces***



A Possible Solution

- **Control of Input and Outputs of Sub-Systems**
  - ***Allows for Automated Test Vector Creation***
    - Simulation (PIL)
    - HIL

Iron Bird Testing



A Case Study

- **Lockheed Martin – A Recent Example**
  - ***Independently Funded Concept Demonstrator***
  - ***Rapid Design-Build-Fly Program***
  - ***Very Small Team***

A Case Study

- **Rapid Simulation and Flight Software Development**
  - ***Developmental / Analysis Simulation – <1 Month***
  - ***Real-Time Piloted Simulation – <2 Months***
  - ***HIL & Engineering Test Stand – 7 Months***
  - ***SCOs – 9 Months***
  - ***Taxi – 12 Months***
  - ***1st Flight – 13 Months***

A Case Study

- **Only 2 Flight Controls Software Changes**
  - ***Calibration Tables (Scheduled Update)***
    - During SCOs
  - ***Flight Test Data Output Update***
    - Ethernet to RS422

Summary

- **Model Based Design +**
  - ***Reduces Design Process Delays***
  - ***Model Drawing Standards***
    - Reduces Learning Curve
    - Increases Model Reuse
  - ***Automatic Code Generation***
    - Reduces Manpower Required
    - Reduces Errors Early
  - ***Embedded ICD in Model***
    - Reduces Errors in Documentation
    - Reduces Errors in Integration
  - ***Built in Test and Data Pump***
    - Increases Test Efficiency

Reduces Total Time and Cost to Deploy

Questions?

Never Forget Who You are Working For !