MATLAB EXPO 2019

Ingeniería de sistemas: de los requisitos a la arquitectura y a la simulación

Luis López
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design

REQ 3.1 ENABLING CRUISE CONTROL
Cruise control is enabled when ......

ENABLE SWITCH DETECTION
If the Enable switch is pressed ......

Implemented By

Implemented By

Derives

reqMode.Cruise
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design
- Connected environment for designing and analyzing architectures and designs
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design

- Connected environment for designing and analyzing architectures and designs

- Integrated platform for analyzing all parts of your architecture in one multi-domain environment

Dynamic Systems  State Machines  Discrete-Event  Physical Modeling
What does that mean?

**Early in the Process**

Concepts/Descriptions

**Later in the Process**

Models
What is the Gap?

Early in the Process
Concepts/Descriptions

Later in the Process
Models

Digital Thread
Connected Environment
Analysis & Simulation Platform
1. Functional Requirements

1.1. Normal Mode of Operation
During the normal mode of operation, the Fault Tolerant Fuel Control System shall determine the fuel rate which is injected at the valves.

1.1.1. Stoichiometric mixture ratio
During normal model of operation, the System shall maintain the stoichiometric mixture target ratio of 14.6.

1.1.2. Oxygen Sensor (EGO)
MathWorks Solution: System Composer R2019a and

✓ Be Intuitive
✓ Facilitate Analysis
✓ Tackle Complexity
✓ Enable Implementation

Simulink

Requirements Coverage Reporting and Impact Analysis

Simulink Requirements
Now let’s see it in action
How to run the Helivolland model:
1. See the Aerospace Blockset User’s Guide for instructions to set up FlightGear.
2. Install the dch2 geometry model to FlightGear’s data/Aircraft directory. The geometry is downloadable from www.flightgear.org.
3. To start FlightGear, execute the script in a powershell window.

Requirements - UAS_reference_architecture

<table>
<thead>
<tr>
<th>Index</th>
<th>Summary</th>
<th>Implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>Communications</td>
<td></td>
</tr>
<tr>
<td>1.3</td>
<td>Payload Capabilities</td>
<td></td>
</tr>
<tr>
<td>1.4</td>
<td>Construction</td>
<td></td>
</tr>
<tr>
<td>1.4.1</td>
<td>Modularity</td>
<td></td>
</tr>
<tr>
<td>1.4.2</td>
<td>Propulsion Power</td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>Flying Qualities</td>
<td></td>
</tr>
</tbody>
</table>

VariableStepAuto
Propulsion Power Subsystem

Gas Engine: Nine-cylinder, air-cooled, radial aircraft engine
Fuel type: 80/87 grade aviation gasoline
Dry weight: 290 kg
Power output: 400 hp (298 kW) at
Propulsion Power Subsystem

Engine: Nine-cylinder, air-cooled, radial aircraft engine
Fuel type: 80/87 grade aviation gasoline
Dry weight/lb (1.03 kW/kg): 290 kg
Power output: 400 hp (298 kW) at 2,200 RPM up to 5,000 ft (1,500 m)
Electrical Subsystem

#35: Propulsion Power

Propulsion Power Subsystem
< Engine_Power_Nm_s >

apControls

EnvBus

enginePower

Actuator Power Subsystem
< Actuators >

actuatorDeflectors

uaPropulsionConfiguration
<table>
<thead>
<tr>
<th>Instances</th>
<th>Mass</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>UAS_reference_architecture Electric_budgetRollup</td>
<td>392.33</td>
<td>1756143000</td>
</tr>
<tr>
<td>BVLOS Navigation</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Ground Station</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Communication Box</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Ground Station GPS Interface</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>USB Serial Converter</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Wireless Communication Subsystem</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>GPS receiver</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Guidance and Navigation Computer</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Flight Commands</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Payload Computer</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Vehicle</td>
<td>392.33</td>
<td>1756143000</td>
</tr>
<tr>
<td>Communications Subsystem</td>
<td>2.63</td>
<td>58050</td>
</tr>
<tr>
<td>Automatic Dependent Surveillance-Broadcast</td>
<td>0.05</td>
<td>5000</td>
</tr>
<tr>
<td>C-Band Radio Modem</td>
<td>0.05</td>
<td>2000</td>
</tr>
<tr>
<td>KU-Band Radio TX/RX</td>
<td>2.5</td>
<td>50000</td>
</tr>
<tr>
<td>On-Board GPS</td>
<td>0.01</td>
<td>50</td>
</tr>
<tr>
<td>Radio RX PPM/PWM</td>
<td>0.02</td>
<td>1000</td>
</tr>
<tr>
<td>Electrical Subsystem</td>
<td>143.15</td>
<td>1753550900</td>
</tr>
<tr>
<td>Actuator Power Subsystem</td>
<td>8</td>
<td>300000</td>
</tr>
<tr>
<td>Power Distribution</td>
<td>10</td>
<td>1000</td>
</tr>
<tr>
<td>Power Monitor</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>Power Source</td>
<td>20</td>
<td>1000</td>
</tr>
<tr>
<td>Propulsion Power Subsystem (Electric)</td>
<td>100</td>
<td>175000000</td>
</tr>
<tr>
<td>Vehicle Power Subsystem</td>
<td>5</td>
<td>50000</td>
</tr>
<tr>
<td>apitRegulator</td>
<td>0.05</td>
<td>20</td>
</tr>
<tr>
<td>commnRegulator</td>
<td>0.05</td>
<td>1000</td>
</tr>
<tr>
<td>pIRegulator</td>
<td>0.05</td>
<td>2000</td>
</tr>
<tr>
<td>Environment</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Simulink Requirements
Digital Thread from Requirements to Architecture and Design

Author requirements or view from external source

Link requirements, architectures, design, code and test

Identify gaps in architecture or design

Identify impact of requirement changes

MATLAB EXPO 2019
System Composer
Intuitively design system and software architectures

Description

= =

Architecture
System Composer
Perform trade studies based on data driven analysis to optimize architectures

Add custom data
Create analysis model
Calculate mass roll-up data
System Composer
Tackle Architecture complexity with spotlight views
Generate Simulink models from architecture components

Link Simulink models to architecture components

Autogenerated by System Composer on March 25, 2019 2:00 pm EST
Simulink: A Multi-Language Simulation Environment

Dynamic Systems

State Machines

Discrete-Event Systems

Physical Modeling

Object-Oriented
Learn More

- Simulink Requirement Webpage
- System Composer Webpage
- System Modeling and Simulation Webpage
- Trial