Desktop real time flight simulator for control design

By

T Vijeesh, Technical Officer, FMCD, CSIR-NAL, Bangalore
C Kamali, Scientist, FMCD, CSIR-NAL, Bangalore
Prem Kumar B, Project Assistant,,FMCD, CSIR-NAL , Bangalore
A A Pashilkar, Group head Flight simulation, FMCD, CSIR-NAL, Bangalore

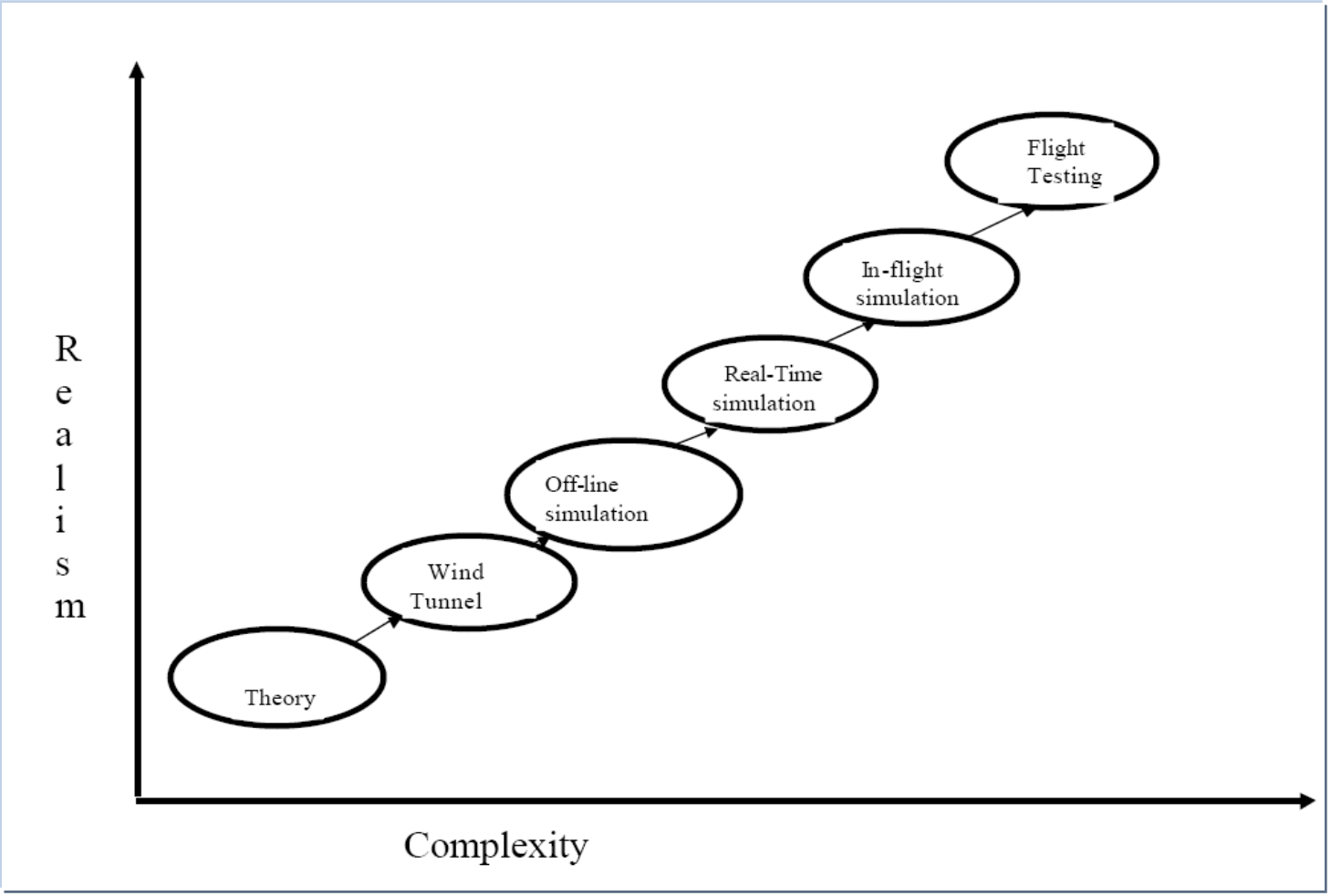
Overview

- Introduction
- Aircraft model
- Control/AP design
- Pilot inputs and Code generation
- GUI design
- Visualization, data logging and plotting
- Integrating the components

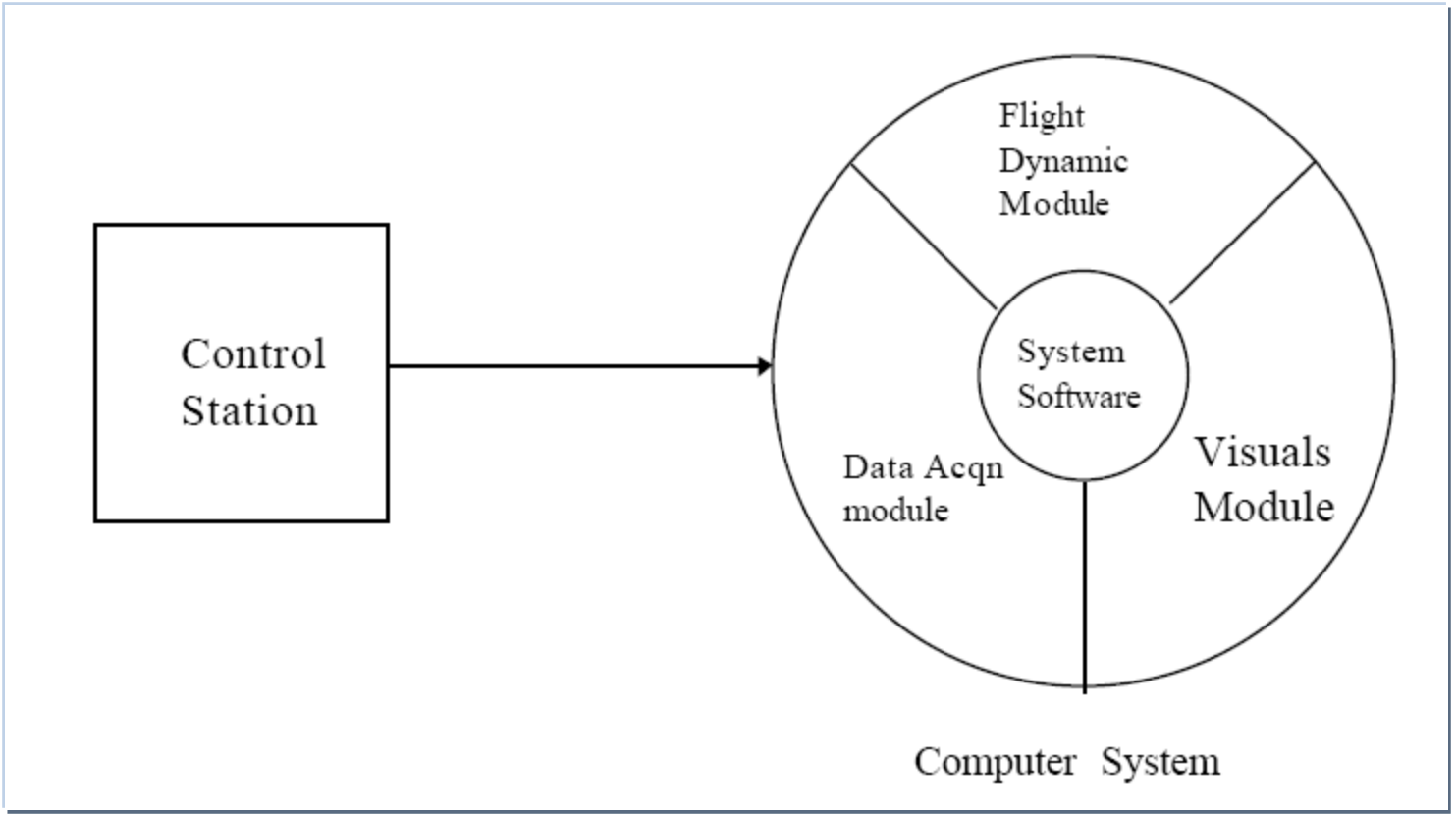
Introduction

- Flight simulation is an essential tool in any aircraft development program.
- It is inevitable for control design and algorithm developments.
- It can also be used for pilot training purposes.
- Desktop based flight simulation eases implementation and validation of flight control strategies.
- Users can work within the same environment from the requirement analysis to the flight simulation and controller design and implementation
- This presentation brings the highlights of how a desktop flight simulator is developed at NAL for control design.

Role of flight simulation in aircraft design



Composition of flight simulation

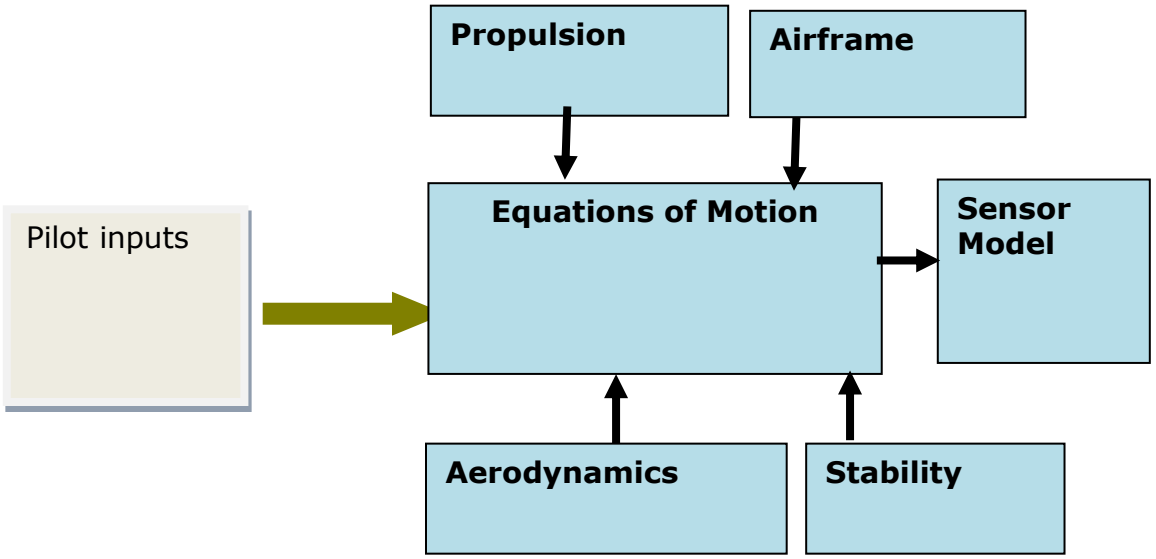


DFS Applications

- Affordable aircraft simulator in all educational institutions imparting aerospace studies
- Aircraft autopilot control design and evaluation
- Guidance and navigation algorithm development
- Terrain data integrity studies
- Enhanced synthetic vision studies
- Human factor studies

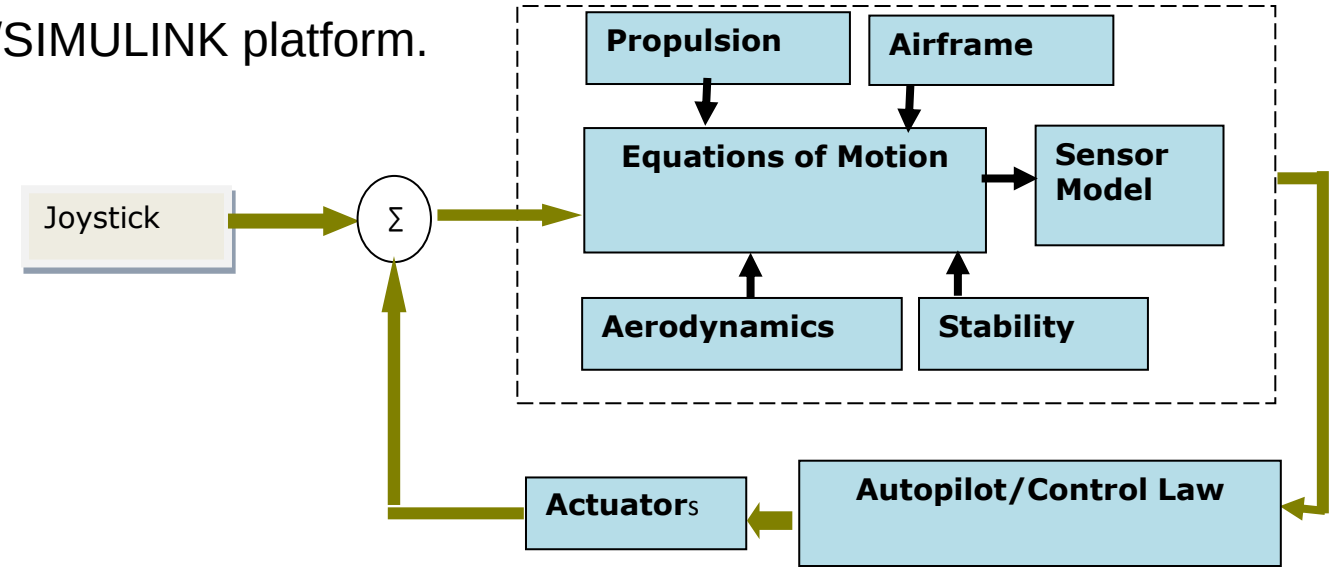
DFS Aircraft Model

- Aircraft Simulation model is developed on MATLAB/Simulink platform.
- This includes:
  - Six degrees of freedom equations of motion for aircraft dynamics
  - Propulsion model
  - Aerodynamic model
  - Structural model
  - Atmospheric model
  - Sensor model
  - Pilot inputs



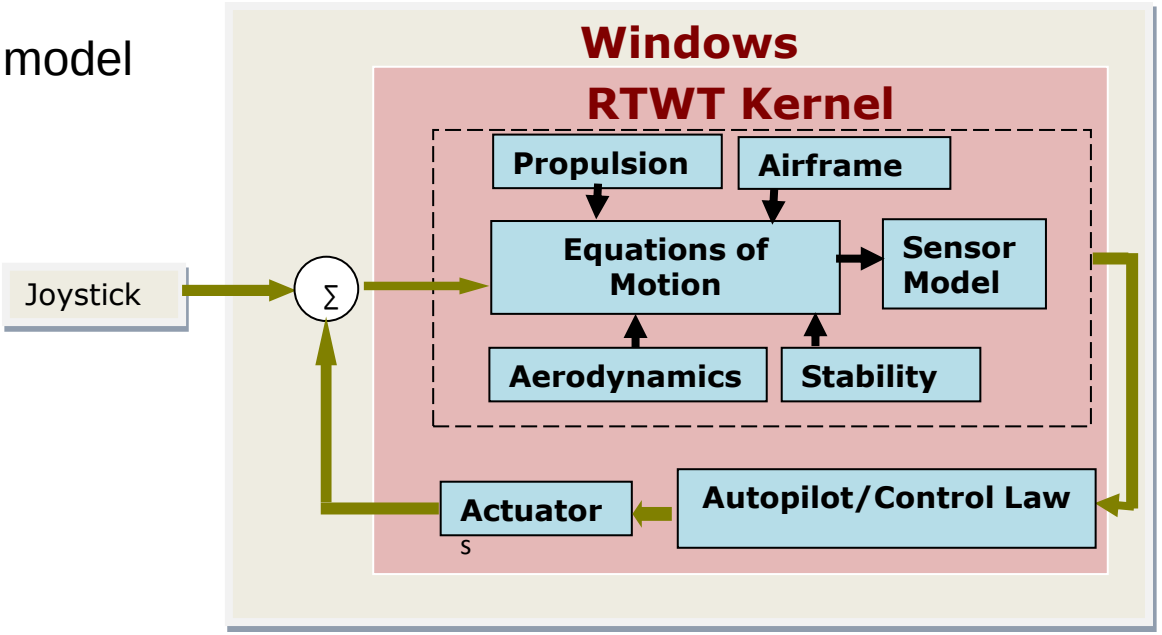
DFS Control/Autopilot design

- Control/Autopilot design is carried out using the linearised models obtained from the aircraft open loop simulation model.
- The control addressed in this presentation involves only proportional, proportional-integral and proportional-integral –derivative control.
- The design and implementation of all autopilot modes are carried out in MATLAB/SIMULINK platform.

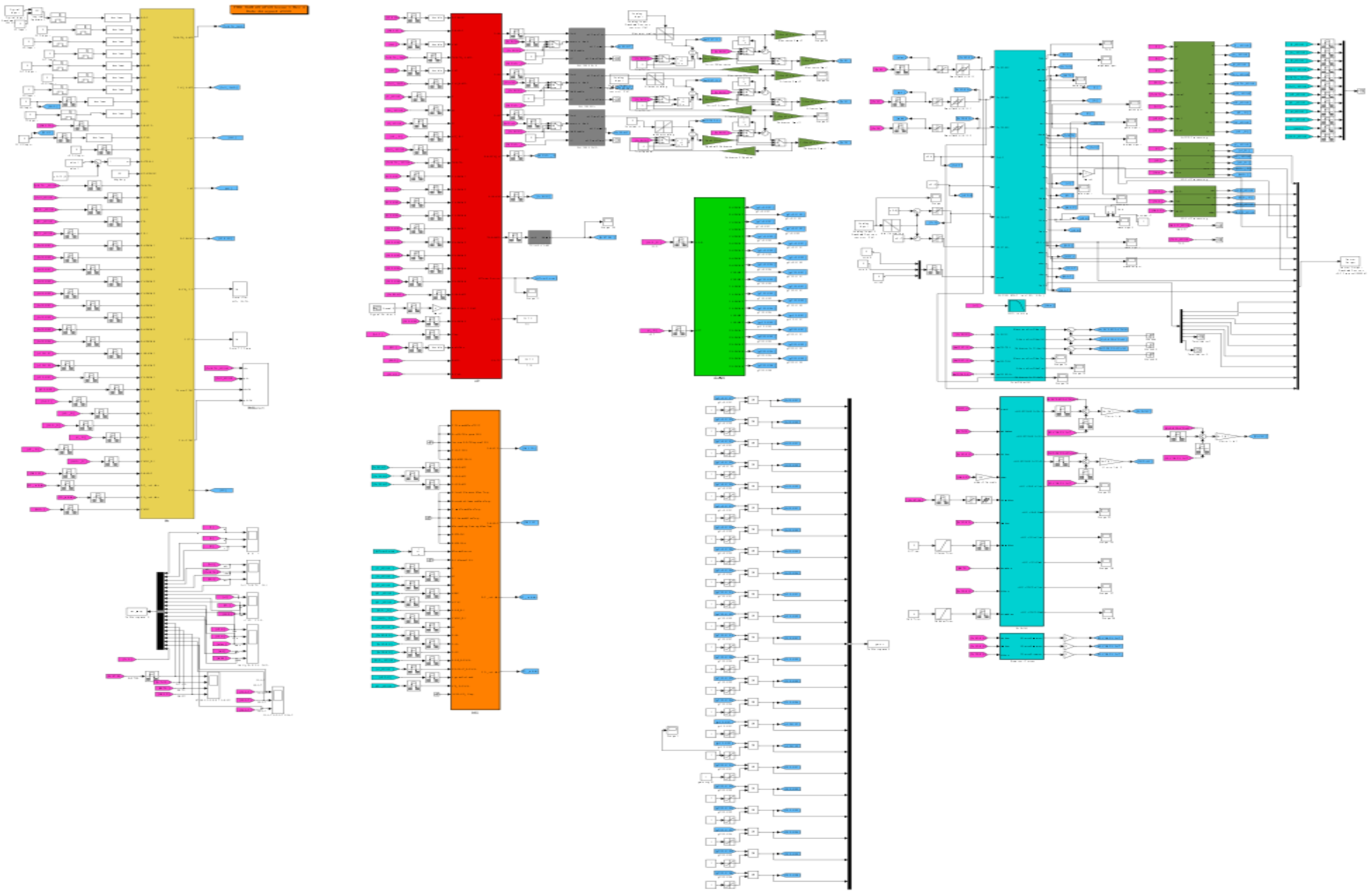


DFS Pilot inputs and code generation

- USB joystick is interfaced with the flight model.
- This is achieved using Analog Input block available in Real Time Windows Target toolbox.
- This block is configured as “Standard Devices joystick”
- Real Time Windows Target kernel is generated from the (closed loop aircraft) simulink model



DFS MATLAB SIMULINK MODEL

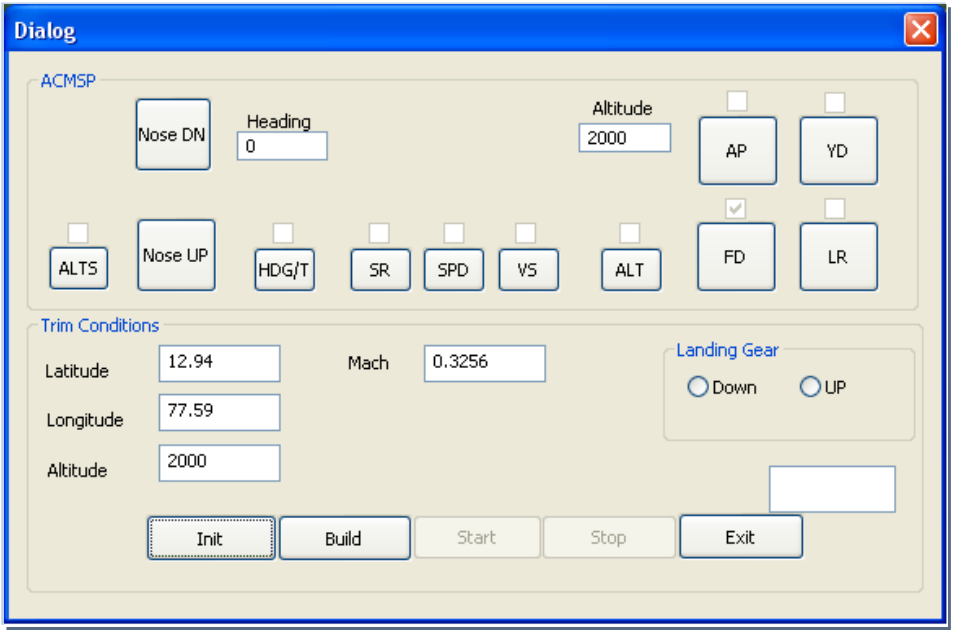


Why RTWT?

- Designers may find it difficult to use legacy codes for aircraft model and control algorithms in a real time simulation.
- RTWT provides features to run simulink and state flow models in real time on desktop or laptop PCs.
- The RTWT includes I/O device drivers to support an extensive selection of I/O boards (for e.g analog inputs/outputs from Analog devices, National Instruments data acquisition cards) and also UDP socket interfaces.
- This enables the designers to interface the system to sensors, actuators and other devices for experimentation, development and testing of the real time systems.

Graphical User Interface

- A Visual C++ based GUI is developed that enables code generation, loading the kernel and to control the simulation execution.
- GUI includes autopilot panel that interacts with the flight model in real time.
- MATLAB Engine API is used to interact with the model



Why Visual C ++ Graphical User Interface?

- MATLAB GUI with RTWT Crashes frequently
- Limited customization of instrument panels
- MATLAB provides APIs that can be used in C/C++ application
- Hence VC++ GUI is highly flexible for the present application

Instrumentation

- The Primary flight display is developed using VAPS XT and interfaced to the GUI.
- VAPS XT (Virtual Applications Prototyping System Extended) is a software tool for the rapid development of dynamic, interactive, real-time graphical Human Machine Interface(HMIs)
- OpenGL Viewport is created in MFC dialog and the VAPS object is loaded in to that viewport.

Visualization

- Out of the window visualization is required for the pilot to get the motion cues.
- Various COTS tools are available for visualization.
- To obtain a cost effective solution open source tools are used.
- For the present application, one could use flightgear.
- If customization is required one can resort to tools like OpenSceneGraph(OSG).
- In this work OSG based visualization software is developed.

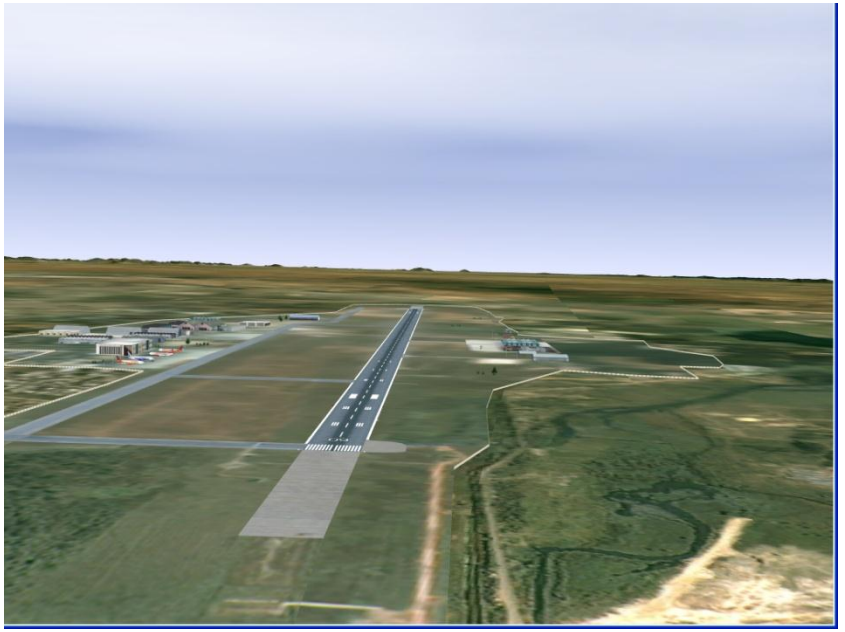
Generation of Visual Images

- Generate the terrain database from height information and geo referenced ground textures.
- Modeling airport including runways and buildings
- Rendering - translating the virtual world for viewing from a particular view point - that of the pilot’s eye-point and projecting 3D world on a 2D monitor screen by appropriate coordinate transformation.

Tools used for Visual Images

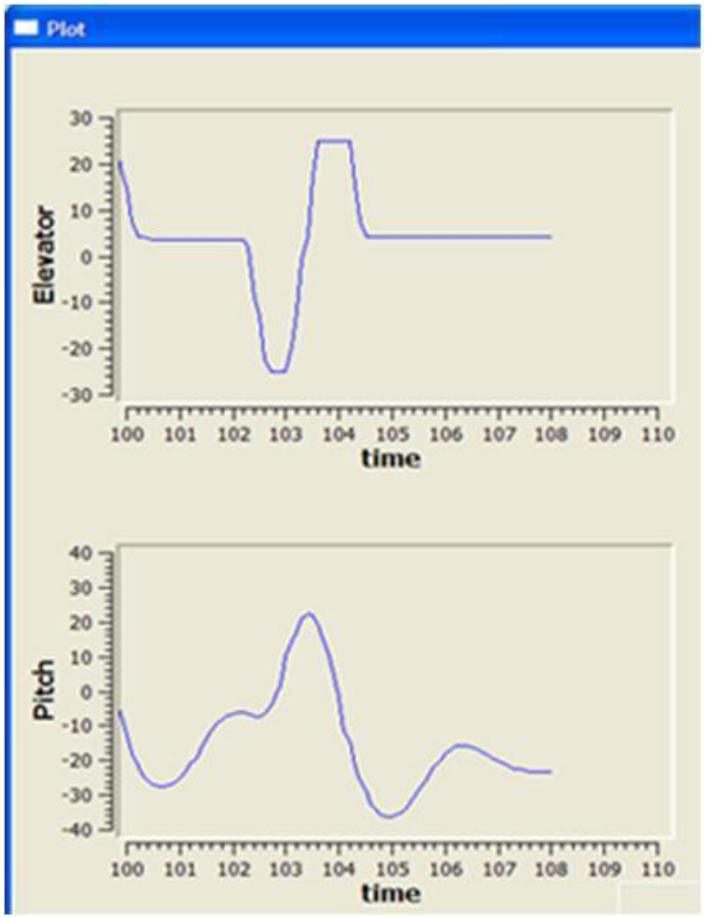
The following tools are used for this application:

- GDAL (geo referencing the textures)
- VirtualPlanetBuilder (creating the terrain database)
- Presagis Creator (Airport 3D modeling)
- Delta3D (real time rendering)



Real time plotting and data logging

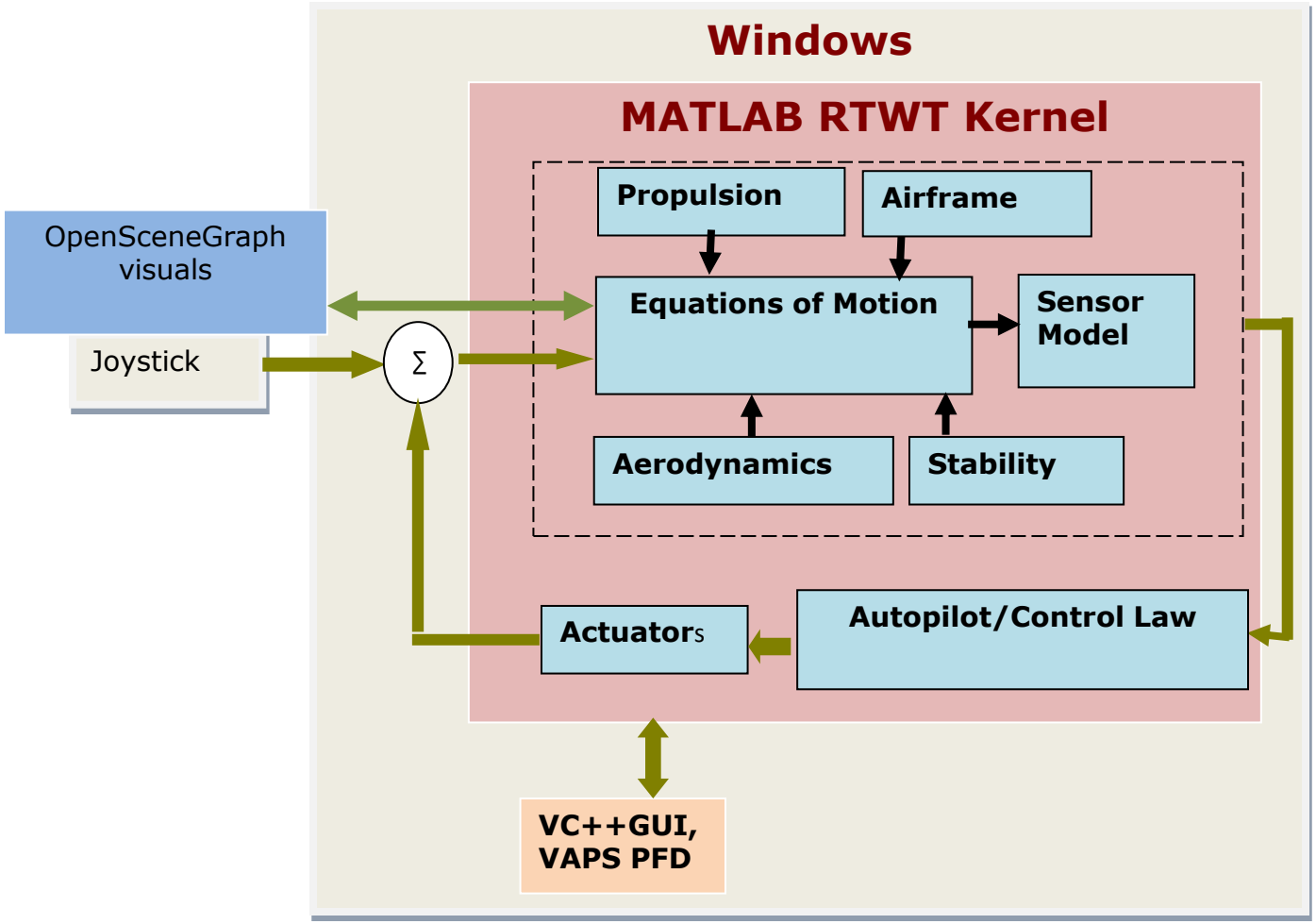
- Real time plotting and data logging is required for analysis.
- Real time plotting is achieved using QT.
- For data logging file write is performed in QT application.



Integrating the components to MATLAB environment

- Intercommunication between the above components are achieved by-
- RTWT UDP Block is added to the simulink model to communicate with the OTW visuals in real time.
  - (Adding multiple UDP blocks slows down the communication)
- GUI will interact with the model using constant blocks and scope objects.
  - (MATLAB GUI can use event listeners for getting data out from model)

Block diagram of DFS



Snapshot of DFS

Conclusions

- Cost effective desktop flight simulator is built around the MATLAB environment.
- Work exhibits capability of integrating several open source tools to MATLAB.
- Design evaluation and real time flying can all be performed in a single desktop PC.

Thank You