Scalable Visual Simulation Platform (SVSP)

Deployment of Microsoft Visual Simulation Platform™(ESP) in the area of professional flight training and system simulators

Concept, design and approval.
Valery Burkin. 2009.

Purpose of SVSP

- Provide cost efficient development and training platform
  - Represent basis technology for modular simulation software design and development for both the native C/C++ and COM/.NET run-times.
  - Provide extension of ESP™ to enable professionally designed modules with high-end graphics capabilities.
  - Minimize development costs through the extension and usage of available code.

- Usage of modern SW technology available on PC
  - Modern software development tools like MS Visual Studio or GCC provide highly optimized code and present excellent code debugging capability.
  - The rendering and shading capabilities of high-end 3D-graphics hardware are considered to be capable of covering all aspects of modern professional flight simulation.

Main features of ESP™ platform

- WGS-84 high precession reference earth model.
- Multi-window graphics interface.
- Barotropic atmospheric model. Top ceiling at 440000 feet.
- Accurate flight model only up to ~22-25 degrees angle of attack.
- Beyond ~25 degrees flight model needs major correction.
- Deep wing stalls are generally not covered. Only slight departures are modeled.
- C(L) and C(D) are configurable Look-Up Tables (LUT).
- Allows for multiple LUT entries at different altitudes and IAP ratios.
- Extensive read-write simulation data interface. SimConnect API.
- Control and instrument Gauge module development is C and XML based.
- Includes redistributable C++ and .NET run-times.
- Based on Direct3D9/10.
- Allows for the design of exceptionally quality aircraft models. Very fast model export and rendering capabilities.
- Good documented SDK.

Basic features of SVSP

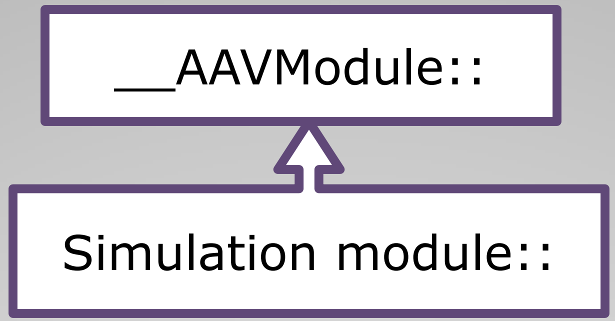
- Based on Microsoft ESP™ and its extensive SDK.
- Modular software design (Scalable platform)

  Each simulated object e.g. Primary Flight Control, Head Up Display, Touch-Screen HDD, EICAS display e.t.c. is represented as a separate software module, which uses its own instance of a data communication interface to ESP and separate data processing thread (module task). Run-time link capability. Every module implements base __AAVModule interface.
- Binary module compatibility on demand (COM/.NET concept)

  There is no general programming language restriction. Every module can be programmed using its own programming language e.g. C, C++, C#, VB, XML, MATLAB/SIMULINK. This requires every module to be binary compatible with system Run-Time (C++/.NET).
- Provides own 3D-Graphics layer (D3D9/10 or OpenGL based)

  ESP itself restricts every linked module in terms of its own graphics demands. There is generally no way to utilize D3D9/10 methods to render in the main frame buffer synchronously or asynchronously.
  SVSP's graphics layer overcomes that restriction to provide professionally designed HUD/HMD modules with on-screen rendering capabilities with no regard what 3D standard is used.

SVSP Module

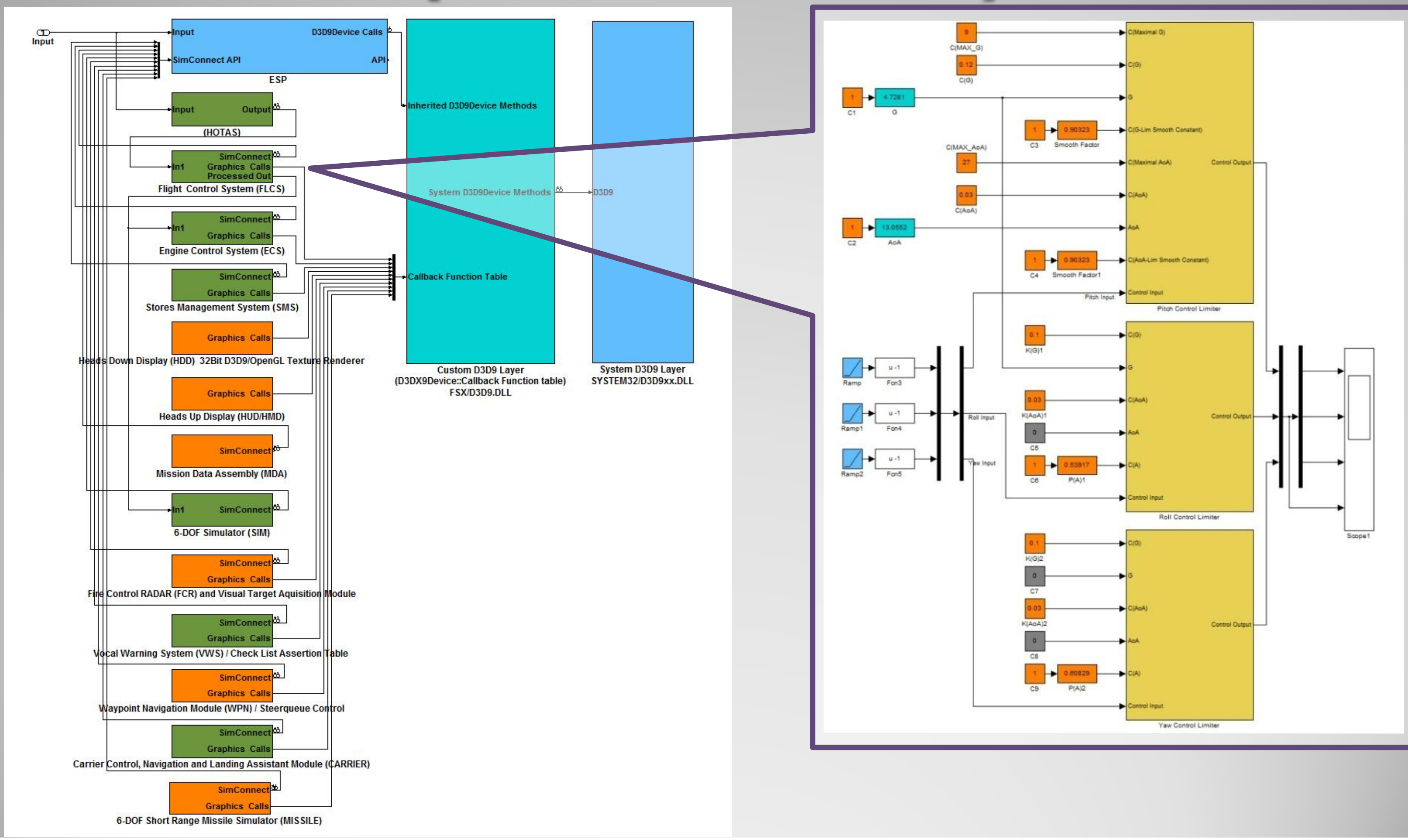


- Provides basis for modular design, development and reusability.
- Single class, single level inheritance of a __AAVModule:: base class.

__AAVModule class provides:

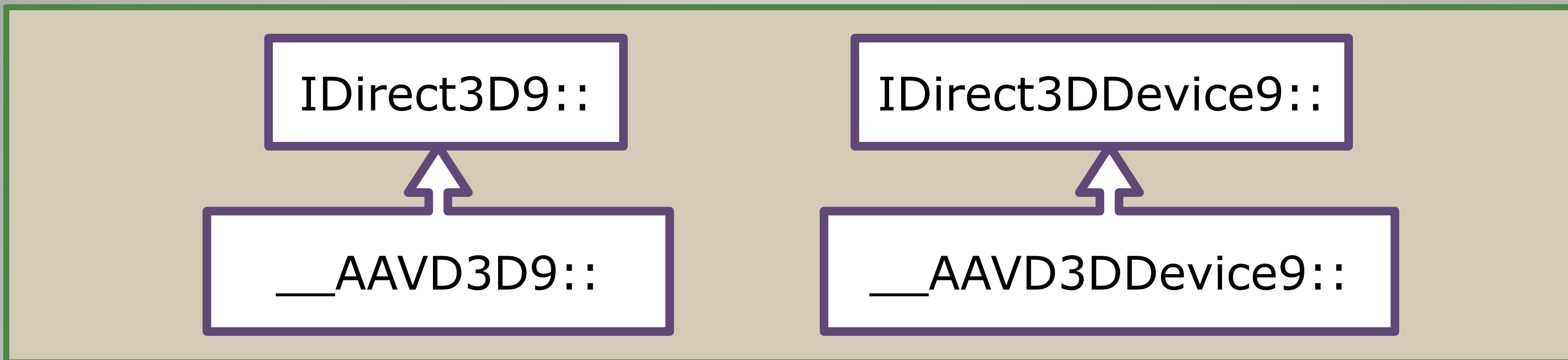
1. Basis 3D on-screen and 2D gauge texture rendering.
2. Basis math for vector and tensor transformation methods. Global-To-Local, Local-To-Body. Cartesian and spherical coordinates.
3. Basis touch-screen and gauge event handling methods.
4. Basis sim-data communication handling. Read,Write,Event data dispatch.
5. Basis module task(thread) handling. Start,Stop,Pause,Resume events.
6. Compliance to Advanced Aerial Vehicle (AAV) standard.

SVSP Module context diagram (SIMULINK Model)



SVSP 3D layer (__AAVD3D9::)

- __AAVD3D9 library is an extension (public class inheritance) of a system D3D9_vX.X dynamic link library (DLL).

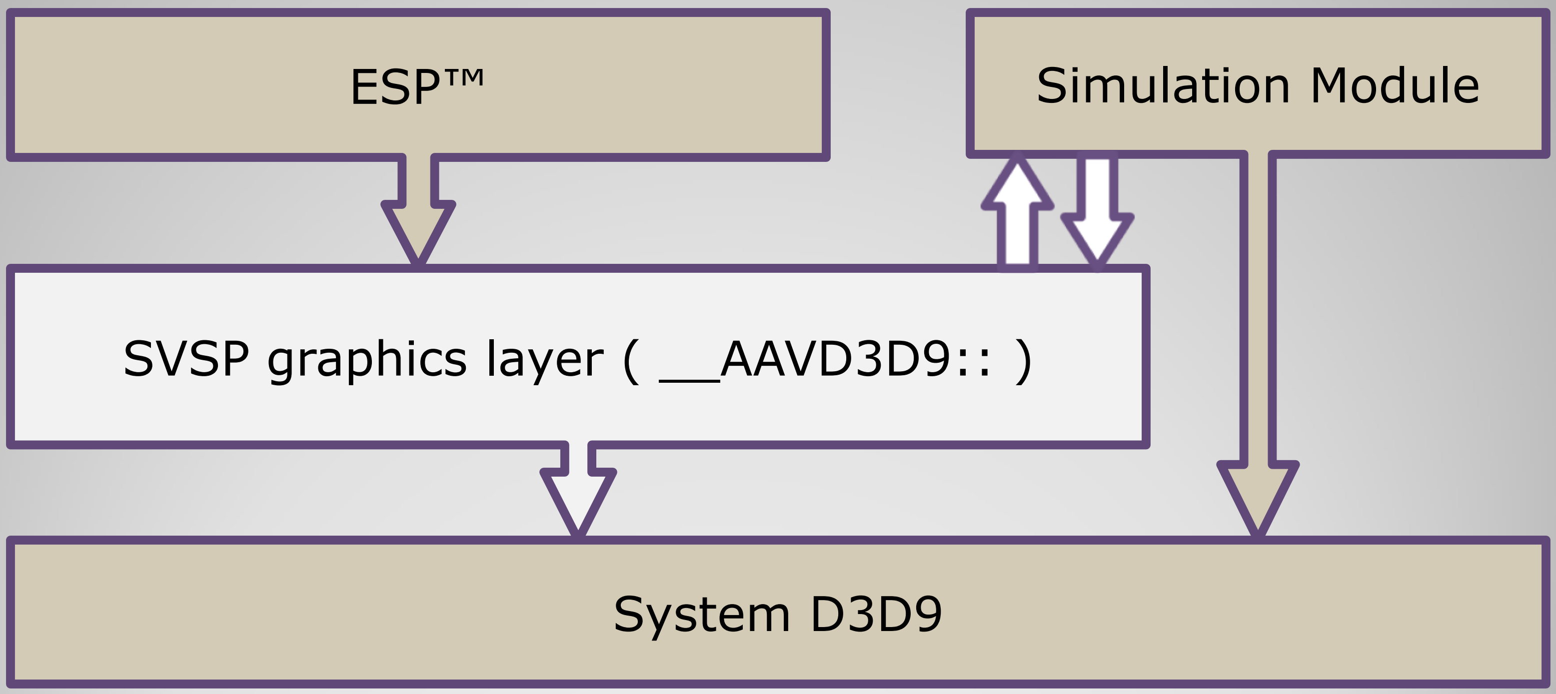


- Both __AAVD3D9:: and __AAVD3DDevice9:: classes provide redirected function calls to all inherited methods. Call redirection is performed via corresponding callback functions available upon subscription to a certain graphics device event.

- AAVD3D9_RegisterCallback(AAVD3D9_SOME_CALLBACK, &Module::Callback, this);
- AAVD3D9_UnregisterCallback(AAVD3D9_SOME_CALLBACK);

SVSP graphics layer context

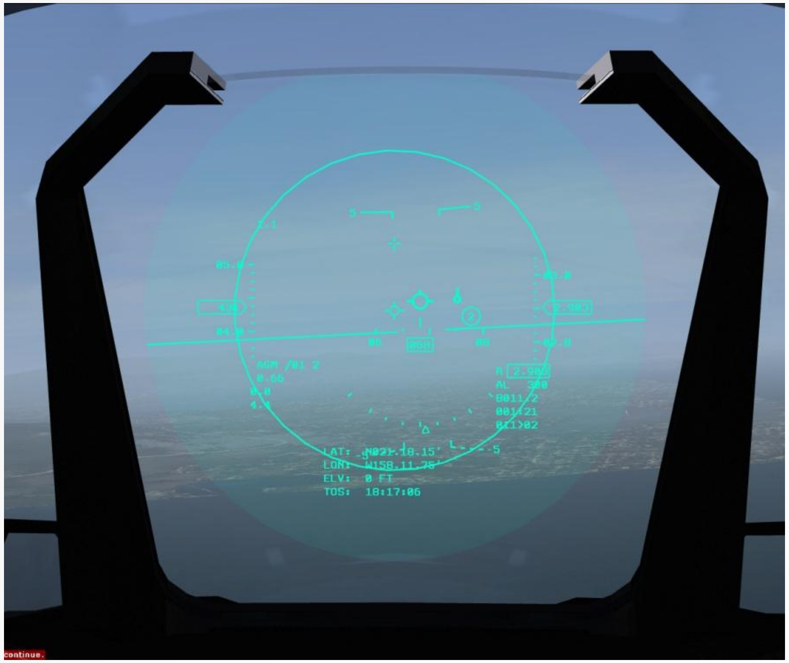
__AAVD3D9 graphics layer intercepts all graphics device calls to provide Simulation Modules with all necessary 3D graphics capabilities.



Application of SVSP

Head-Up/Down display modules are the basic application of SVSP technology and represent conceptual approval of the platform.

Collimated Head-Up Display (HUD/HMD)

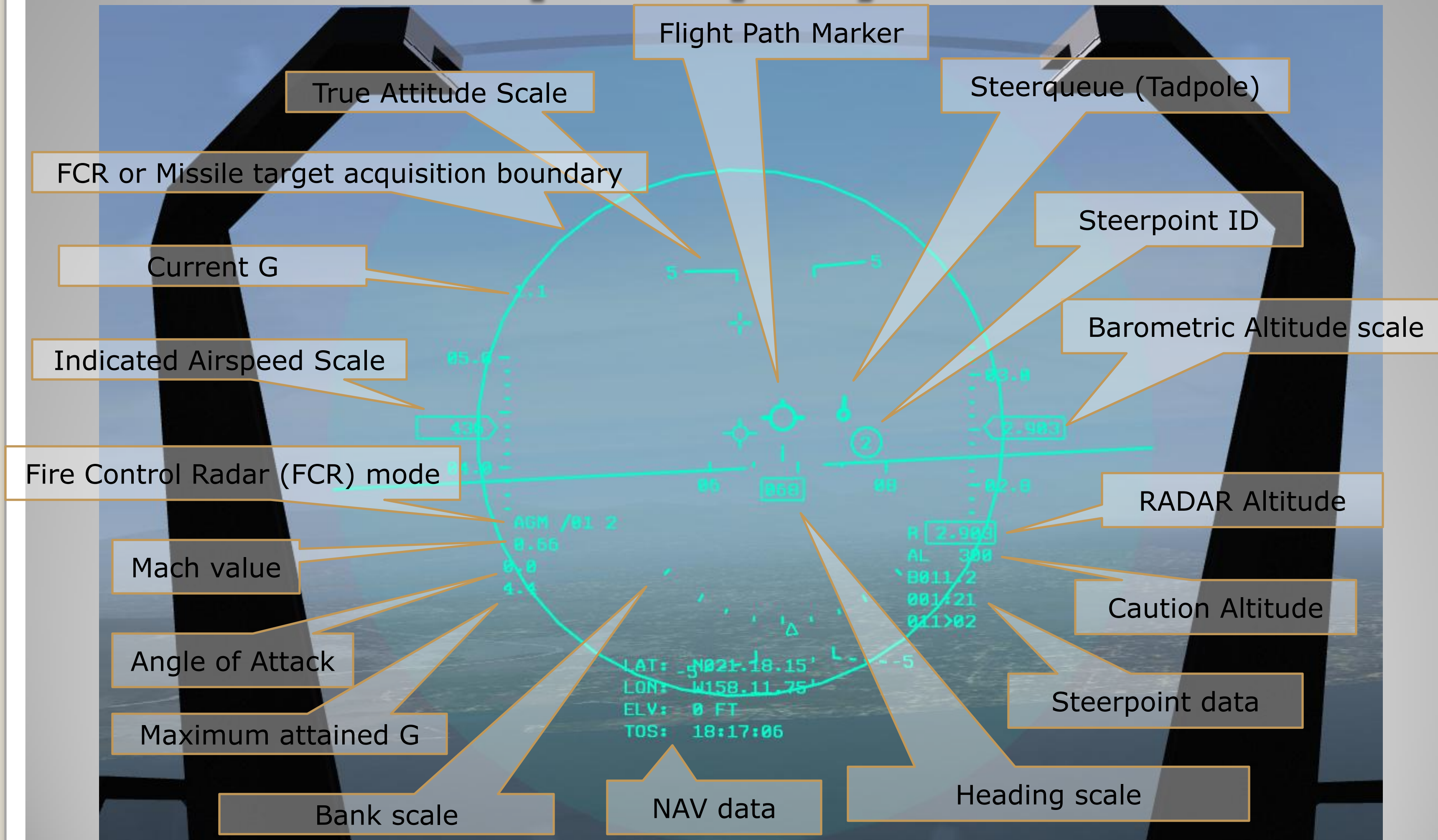


Touch Screen Head-Down Display (HDD/DEC)

Head-Up Display features

- Direct rendering in the ESP’s Framebuffer. No texture rendering.
- Collimated Projection (Two Pass Stencil rendering).
- Conformal Flight-Path-Marker (Velocity Vector) handling. Points in the true flight direction.
- Linked to true HUD position in the aircraft model.
- Helmet-Mounted display (HMD) mode.

Head-Up Display Elements



SVSP AAVTextureRenderer class

- Provides run-time texture rendering functionality

  Touch-screen glass cockpit HDD panel is a 2D **32Bit RGBA texture**. There is absolutely no restriction on what graphics standard is used to render it. Industry standard **OpenGL/ES/SC** driven panels can be rendered in offset buffers via separate GPU asynchronously and applied as 32Bit texture buffers in the ESP's cockpits in a ring-buffer manner.

Internal 6-DOF VTOL simulator

Basic VTOL simulation engine.

- 6-Degree Of Freedom computational model for linear and angular velocity in aircraft body reference system.

Basic Flight Control System (FLCS)

Provides flight stability at high G and high AoA. Basic care-free handling functions, angular stability during hover mode. Terrain following. Conversion to level flight.

- Flight condition state machine :
  (SLEW, VTOL, GROUND, AIRBORNE_GEAR_DOWN, AIRBORNE_GEAR_UP)
- (AIRBORNE) Flight surface deflection controller (G and AoA limiter).
- (AIRBORNE) Pitch and flight path hold. Automatic pitch trim control.
- (AIRBORNE) Automatic Flaps control.
- (AIRBORNE) Automatic Airbrake control.
- (AIRBORNE) Vectored Thrust control.
- (AIRBORNE) Automatic steer-point navigation.
- (VTOL) Augmented thrust control, linked to Engine Control System.
- (VTOL) Reference Altitude hold. PID Controller.
- (VTOL) Reference Heading hold. PID Controller.
- (VTOL) Torque control.

Flight Surface Deflection Controller

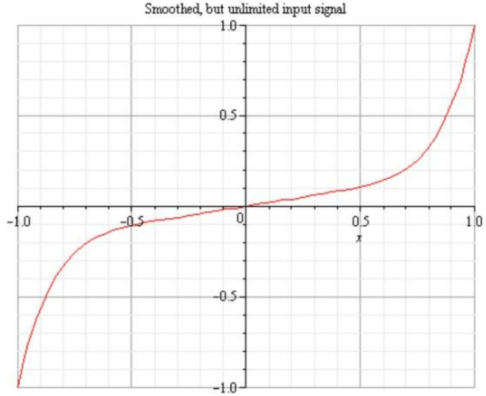
- Flight surface deflection controller and flight stick input processing module. G and AoA limiter. (MATLAB/SIMULINK) model.

Applying the polynomial function assures the signal is smoothed and no sharp gains are present around 0:

$$\boldsymbol{A}_{\boldsymbol{x}} \cdot x^7 + \left(1 - \boldsymbol{A}_{\boldsymbol{x}}\right) \cdot x$$

$$plot\left(0.8 \cdot x^7 + (1 - 0.8) \cdot x, x = -1..1\right)$$

Smoothed, but unlimited input signal



Applying the gain function of current G and AoA parameters would strech down the signal so that the limiting factor at high speed condition would be G and that of the low speed condition would be AoA :

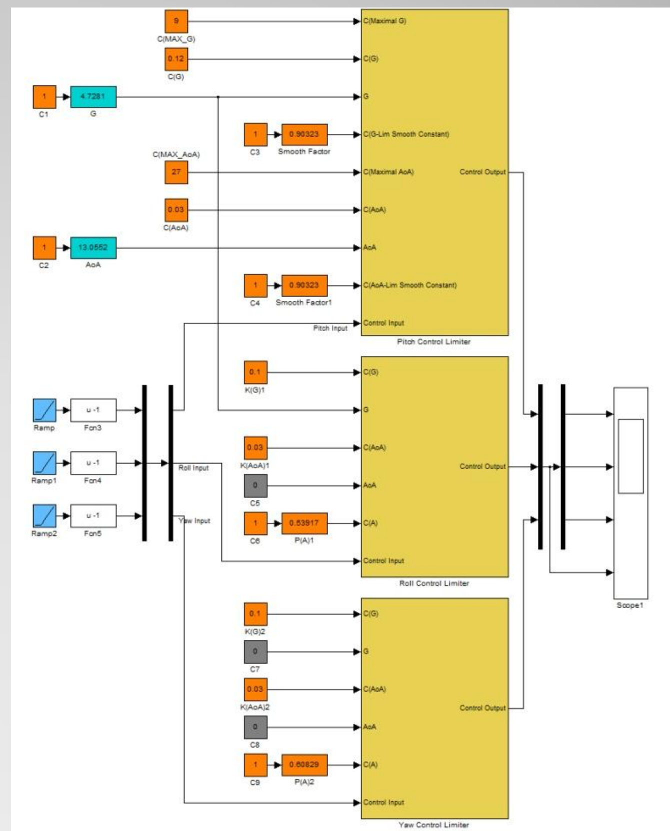
$$f(G, \alpha) = \frac{1}{\left(1 + \left(\boldsymbol{C}_{\boldsymbol{G}} \cdot G\right)^2\right)} \cdot \frac{1}{\left(1 + \left(\boldsymbol{C}_{\alpha} \cdot \alpha\right)^2\right)}$$

where $\boldsymbol{C}_{\boldsymbol{G}}$ and $\boldsymbol{C}_{\alpha}$ are constant factors for G and AoA respectively.

The complete input signal controller function is defined as the following :

$$Y(G, \alpha, x_i) = \frac{1}{\left(1 + \left(\boldsymbol{C}_{\boldsymbol{G}} \cdot G\right)^2\right)} \cdot \frac{1}{\left(1 + \left(\boldsymbol{C}_{\alpha} \cdot \alpha\right)^2\right)} \cdot \left(\boldsymbol{A}_{\boldsymbol{i}} \cdot x_i^7 + \left(1 - \boldsymbol{A}_{\boldsymbol{i}}\right) \cdot x_i\right);$$

All constants are controller specific and normally defined as vectors for pitch, yaw and roll respectively, $\boldsymbol{A}_{\boldsymbol{i}} = \left(\boldsymbol{A}_{pitch}, \boldsymbol{A}_{yaw}, \boldsymbol{A}_{roll}\right)$.



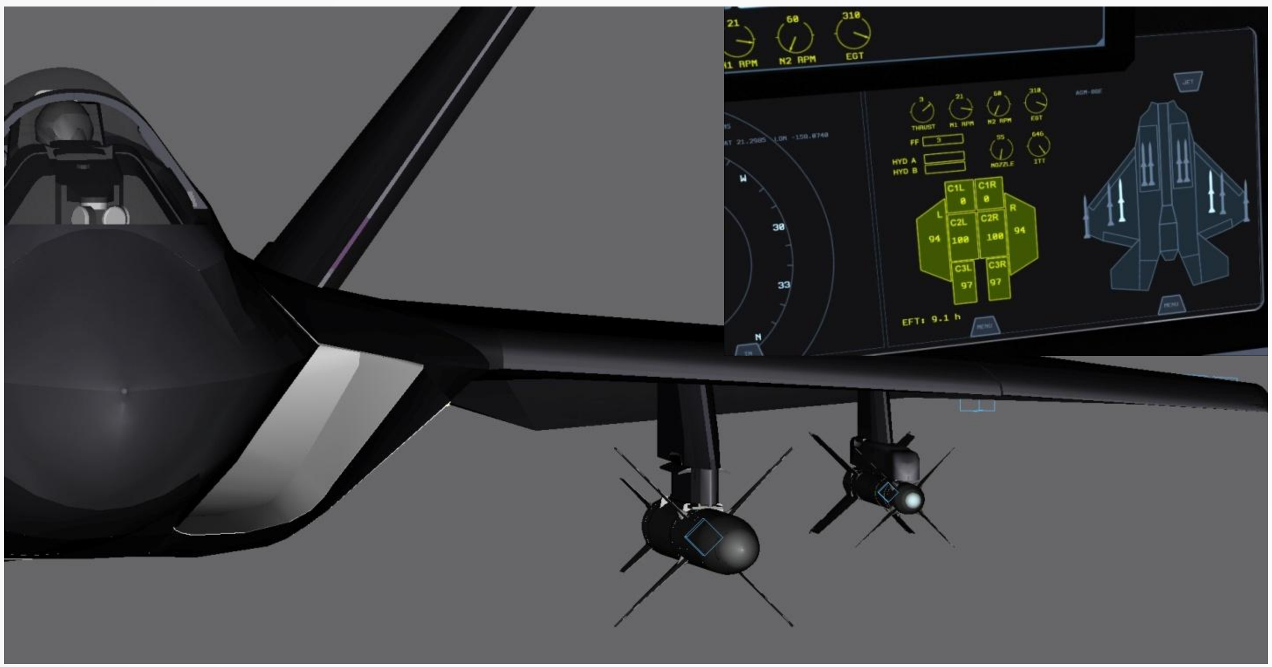
Vocal Warning System (VWS)

- Flight parameter and check list assert table. Visual warning indication stack.
- FLCS state dependent parameter check :
  1. High, Critical and over limited AoA.
  2. Minimum and Maximum airspeed check.
  3. Engine RPM, ITT and EGT check.
  4. Gear down descent rate.
  5. Terrain proximity (Radar altitude change rate).
  6. Over G check.
  7. Fuel quantity check. Warning indication on 1500, 800 and 500 kg.
  8. Bingo fuel check (across all route).
  9. Estimated flight time (fuel flow) check.
  10. Radar Warning Receiver check.

Stores Management System(SMS)

- Manages all external payload and droppable fuel tanks.
- Manages state of a payload item.
- Manages weapon group selection according to FCR mode.

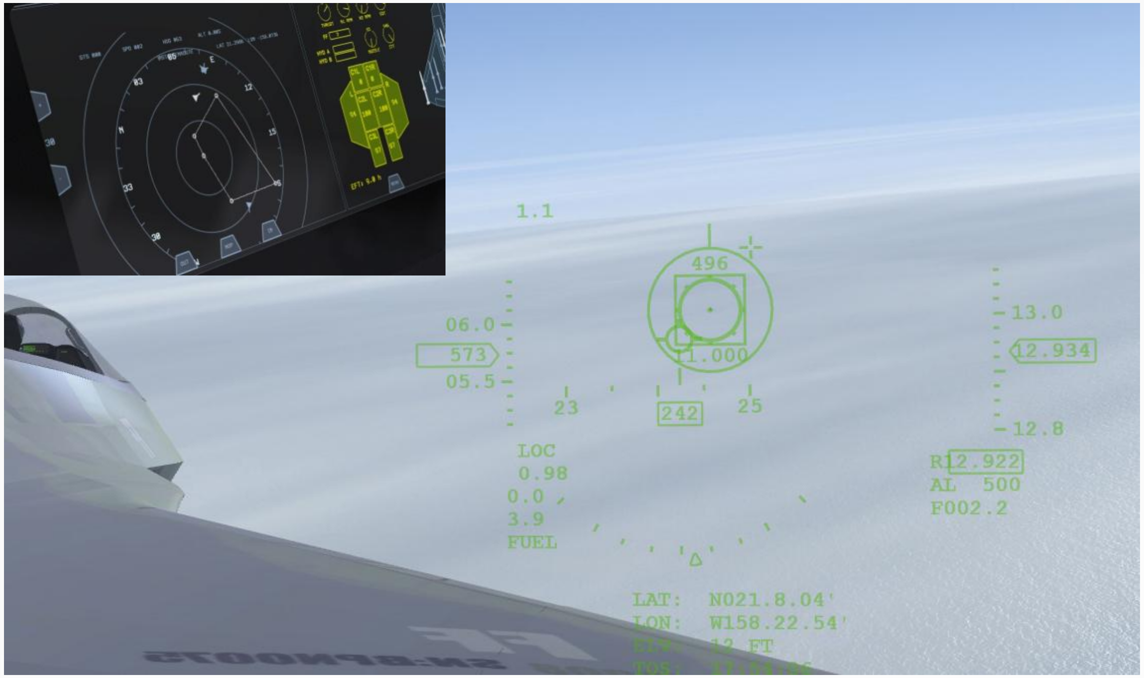
SMS (External model) and HDD console



Fire Control Radar (FCR)

- 16K air/ground/naval traffic monitoring table.
- Single target track (STT), Track while scan (TWS) modes. Airborne only.
- Visual target acquisition through HMD mode.

HMD A-A mode and FCR gauge console



Where is ESP™ being used

BOEING®

The Skunk Works

LOCKHEED MARTIN

Flight-Dynamix, LLC

NORTHROP GRUMMAN