AeroFuncs

Custom Derived Function

January 2014

SYMVIONICS Document SSD-IADS-155, Rev 8.0

SYMVIONICS, Inc.
Telemetry Systems

Created June 4, 2012

Table of Contents

Table of Figures

1. Introduction

This document describes the IADS AeroFuncs Custom Derived Function developed for flight test engineers to monitor aircraft flight conditions and attitude parameters from measured values.

1.1. Overview

The IADS AeroFuncs custom derived function is provided as a Dynamic Link Library (DLL) which contains the following internal functions:

ProgId	*Description*
AeroFuncs.Airspeed	Returns airspeed
AeroFuncs.AltitudeError	Returns altitude error
AeroFuncs.AmbDensityRatio	Returns ambient density ratio
AeroFuncs.AmbPressureRatio	Returns ambient pressure ratio
AeroFuncs.DifferentialPressure	Returns calibrated differential pressure
AeroFuncs.EquivalentAirspeed	Returns equivalent airspeed
AeroFuncs. IncompressibleDynamicPressure	Returns incompressible dynamic pressure
AeroFuncs. IncompressibleDynamicPressure_PSF	Returns incompressible dynamic pressure PSF
AeroFuncs.IndPressureRatio	Returns indicated pressure ratio
AeroFuncs.Mach	Returns mach
AeroFuncs.MachError	Returns mach number error
AeroFuncs.PressureAltitude	Returns calibrated pressure altitude
AeroFuncs.StagPressure	Returns stagnation pressure
AeroFuncs.StatAirTempRatio	Returns static air temperature ratio
AeroFuncs.StaticAirTemp	Returns static air temperature
AeroFuncs.StaticPressure	Returns static pressure (static and dynamic pressure are in inches mercury)
AeroFuncs.TotalAirTemp	Returns total air temperature
AeroFuncs.TotalPressure	Returns calibrated total pressure
AeroFuncs.TrueAirspeed	Returns true airspeed
AeroFuncs.VelocityError	Returns velocity error

Figure 1-1 AeroFuncs.dll Installed Functions

2. Installation

The IADS AeroFuncs.dll is included in IADS Version 8.0 (or greater) and registers automatically upon installation of the IADS application. The following procedures are provided for other circumstances.

2.1. To register the AeroFuncs.dll

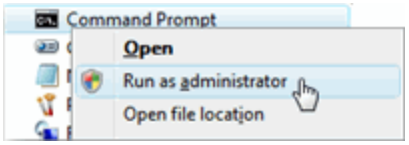
1) Copy the *AeroFuncs.dll* file to a known directory. If you are unsure where to place it, consider the C:\Program Files\IADS directory.
2) In Windows Explorer, navigate to the directory where you copied the file and right-click on the *AeroFuncs.dll* file and choose **Open With …**
3) Click the **Browse** button.
4) Browse to the **C:\Windows\System32** directory.
5) Select the **regsvr32.exe** file and then click the **Open** button.
6) Click **OK**. The function is now registered and available for use within IADS.

2.2. To turn off User Account Control (Windows Vista/7)

1) Open the *Control Panel* (Classic View).
2) Click on the **User Account** icon.
3) Click on **Turn User Account Control on or off**.
4) Uncheck **Use UAC to help protect your computer**.
5) Click **OK**.
6) Click on **Restart Now** to apply the changes.
7) Register the *AeroFuncs.dll*.
8) Repeat steps 1-6 in this set of procedures to turn back on *User Account Control*.

2.3. To register a DLL as an Administrator (Windows Vista/7)

1) Right-click on the Command Prompt icon > **Run as Administrator**.



2) Enter your administrative credentials (User name, Password).
3) In the Command Prompt, navigate to the location of the dll.
4) Type: **regsvr32 AeroFuncs.dll**
5) Press **Enter**.

3. AeroFuncs Installed Functions

3.1. Velocity Functions

3.1.1. Airspeed

Description: Calculates speed in KNOTS given dynamic pressure.

Syntax: AeroFuncs.Airspeed(Dynamic Pressure)

$$Subsonic:\ V = a_0\sqrt{5\left[\left(\frac{dynPressure}{P_0}\right)^{\frac{2}{7}} - 1\right]}$$

$a_0 = Speed\ of\ sound\ at\ sea\ level$

$P_0 = Static\ air\ pressure\ at\ sea\ level$

$Supersonic: Same\ formula\ but\ determination\ of\ dynPressure\ is\ iterative$

3.1.2. Equivalent Airspeed

Description: Calculates equivalent airspeed in KNOTS given static pressure and MACH number.

Syntax: AeroFuncs.EquivalentAirspeed(Static Pressure, Mach)

$$EAS = a_0 M\sqrt{\frac{statPressure}{P_0}}$$

3.1.3. Mach

Description: Calculates MACH number given dynamic pressure and static pressure.

Syntax: AeroFuncs.Mach(Dynamic Pressure, Static Pressure)

$$Subsonic:\ M = \sqrt{5\left[\left(\frac{dynPressure}{statPressure} + 1\right)^{\frac{2}{7}} - 1\right]}$$

$$Supersonic:\ M = 0.88128485\sqrt{\left[\left(\frac{dynPressure}{statPressure} + 1\right)\left(1 - \frac{1}{7\,M_{old}^2}\right)^{2.5}\right]}$$

$M_{old} = Previous\ Mach\ Value$

3.1.4. Mach Error

Description: Calculates MACH number error given indicated MACH and corrected MACH

Syntax: AeroFuncs.MachError(Indicated Mach, Corrected Mach)

$Error = indicatedMach - correctedMach$

3.1.5. True Airspeed

Description: Calculates true airspeed in knots given equivalent airspeed and ambient density ratio.

Syntax: AeroFuncs.TrueAirpseed(Equivalent Airspeed, Ambient Density Ratio)

$$TAS = \frac{EAS}{Ambient\ Density\ Ratio}$$

3.1.6. Velocity Error

Description: Calculates velocity error in knots given indicated airspeed and corrected airspeed.

Syntax: AeroFuncs.VelocityError(Indicated Airspeed, Corrected Airspeed)

$$Error = indicatedAirSpeed - correctedAirSpeed$$

3.2. Pressure Functions (all pressures are in Hg)

3.2.1. Total Pressure

Description: Calculates total pressure given static pressure, dynamic pressure and total pressure source position error.

Syntax: AeroFuncs.TotalPressure(Static Pressure, Dynamic Pressure, Position Error)

$$totalPressure = \frac{statPressure + dynPressure}{1 - positionError}$$

3.2.2. Static Pressure

Description: Calculates calibrated static pressure given static pressure, dynamic pressure, position error correction and angle of attack (AOA) correction.

Syntax: AeroFuncs.StaticPressure(Static Pressure, Dynamic Pressure, Position Error, AOA Correction)

$$staticPressure = \frac{statPressure - (positionError)(dynPressure)}{1 - AOAcorrection}$$

3.2.3. Differential Pressure

Description: Calculates calibrated differential (dynamic) pressure given total pressure and calibrated static pressure.

Syntax: AeroFuncs.DifferentialPressure(Total Pressure, Static Pressure)

$$differentialPressure = totalPressure - staticPressure$$

3.2.4. Incompressible Dynamic Pressure

Description: Calculates incompressible dynamic pressure given static pressure and MACH number.

Syntax: AeroFuncs.IncompressibleDynamicPressure(Static Pressure, Mach)

$$incomprehensibleDynamicPressure = (0.7)(statPressure)(M^2)$$

$$Derived\ from: 9 = \left(\frac{r}{2}\right) pM^2 \quad r = 1.4\ for\ air$$

3.2.5. Incompressible Dynamic Pressure (PSF)

Description: Calculates incompressible dynamic pressure in pounds per square foot given static pressure and MACH number.

Syntax: AeroFuncs.IncompressibleDynamicPressure_PSF(Static Pressure, Mach)

$$incomprehensibleDynamicPressure\_PSF = (0.7)(statPressure)(M^2)(70.7263)$$

3.2.6. Stagnation Pressure

Description: Calculates stagnation pressure (free stream total pressure) given calibrated static pressure and MACH.

Syntax: AeroFuncs.StagPressure(Static Pressure, Mach)

$$stagnationPressure = (statPressure)\left(1 + \frac{M^2}{5}\right)^{\frac{7}{2}}$$

3.2.7. Indicated Pressure Ratio

Description: Calculates indicated pressure ration given indicated static pressure.

Syntax: AeroFuncs.IndPressureRatio(Static Pressure)

$$indicatedPressureRatio = \frac{statPressure\ (indicated)}{P_0}$$

3.2.8. Ambient Pressure Ratio

Description: Calculates ambient pressure ratio given calibrated static pressure

Syntax: AeroFuncs.AmbPressureRatio(Static Pressure)

$$ambientPressureRatio = \frac{statPressure\ (calibrated)}{P_0}$$

3.3. Altitude Functions

3.3.1. Pressure Altitude

Description: Calculates altitude in FEET given static pressure.

Syntax: AeroFuncs.PressureAltitude(Static Pressure)

$pressureAltitude$:

$$statPressure \geq 6.68321, H = (145442)\left[1 - \left(\frac{statPressure}{P_0}\right)^{0.190262}\right] Up\ to\ 11000m/36089.4ft$$

$$statPressure < 6.68321/\geq 1.61672, H = 36089.4 - 20805.7\left[1n - \left(\frac{4.47708 * statPressure}{P_0}\right)\right] Up\ to\ 20000m/6261.8\ ft$$

$$statPressure < 1.61672, H = 65616.8\ ft$$

3.3.2. Altitude Error

Description: Calculates altitude error given indicated pressure altitude and corrected pressure altitude (output is the same units as the two inputs).

Syntax: AeroFuncs.AltitudeError(Indicated Pressure Altitude, Corrected Pressure Altitude)

$$altitudeError = indicatedPressureAltitude - correctedPressureAltitude$$

3.4. Air Functions

3.4.1. Static Air Temperature

Description: Calculates static air temperature in degrees C given static air temperature ratio.

Syntax: AeroFuncs.StaticAirTemp(Static Air Temperature Ratio)

$$staticAirTemp = (T_{sl})(measuredTotalTemp) - 273.15$$

$$T_{sl} = Standard\ sea\ level\ temperature\ (288.15^{\circ}k)$$

3.4.2. Static Air Temperature Ratio

Description: Calculates static air temperature ratio given measured total temperature, MACH number and K factor (K factor is set at 0.99 if the argument is left blank).

Syntax: AeroFuncs.StaticAirTempRatio(Total Temperature, Mach, K factor)

$$staticAirTempRatio = \frac{(totalTemp + 273.15)}{[1 + 0.2(Kfactor)(M^2)](T_{sl})}$$

3.4.3. Total Air Temperature

Description: Calculates total air temperature in degrees C given static air temperature ratio and MACH number.

Syntax: AeroFuncs.TotalAirTemp(Static Air Temperature Ratio, Mach)

$$totalAirTemp = [(1 + 0.2M^2)(T_{sl})(measuredTotalTemp)] - 273.15$$

3.4.4. Ambient Density Ratio

Description: Calculates ambient air density ratio given ambient pressure ratio and ambient temperature ratio.

Syntax: AeroFuncs.AmbDensityRatio(Ambient Pressure Ratio, Ambient Temperature Ratio)

$$ambientDensityRatio = \frac{ambientPressureRatio}{ambientTempRatio}$$

4. Instructions for Use

This section will explain how to create the derived functions in IADS. Figure 4-1 displays the AeroFuncs functions in the Parameter Defaults Table as derived parameters.

	ParameterDe...	Parameter	DataSourceT...	DataSourceArguement
145	AeroFuncs	Mach	Derived	AeroFuncs.Mach(Dynamic Pressure, Static Pressure)
146	AeroFuncs	StaticPressure	Derived	AeroFuncs.StaticPressure(Static Pressure, Dynamic Pressure, Position Error, AOA Correction)
147	AeroFuncs	Airspeed	Derived	AeroFuncs.Airspeed(Dynamic Pressure)
148	AeroFuncs	AltitudeError	Derived	AeroFuncs.AltitudeError(Indicated Pressure Altitude, Corrected Pressure Altitude)
149	AeroFuncs	AmbDensityRatio	Derived	AeroFuncs.AmbDensityRatio(Ambient Pressure Ratio, Ambient Temperature Ratio)

Figure 4-1 Derived Parameter Setup in the Parameter Defaults Table

To create an AeroFuncs derived parameter:

1) In IADS, on the Dashboard click the **Configuration** button.
2) Open the **Data** folder > click **Parameter Defaults**.
3) Copy and paste an existing row of data for a parameter that is similar to the one you are creating.
4) In the *Parameter* column enter a unique parameter name.
5) In the *Data Source Type* column, select **Derived**.
6) Enter the desired *AeroFuncs* function in the *Data Source Argument* column, for example: AeroFuncs.Airspeed(Dynamic Pressure parameter)
7) Click the *Save* button.