



---

***IADS UDP***  
***Custom Derived Function***

---

February 2014 Version 8.1.1  
SYMVIONICS Document SSD-IADS-050  
© 1996-2018 SYMVIONICS, Inc.  
All rights reserved.



## Table of Contents

|   |          |
|---|----------|
| <b>1. Introduction</b> .....  | <b>3</b> |
| 1.1. <i>Overview</i> .....  | 3        |
| <b>2. Installation</b> .....  | <b>4</b> |
| <b>3. Instructions for Use</b> .....  | <b>5</b> |
| 3.1. <i>Creating the Blob Parameter</i> .....                               | 5        |
| 3.1.1. <i>IadsUDP.Read Function Input Arguments</i> .....                   | 6        |
| 3.1.2. <i>Verifying the Blob Parameter in the Blob Viewer Display</i> ..... | 6        |
| 3.2. <i>Creating the Decom Parameters</i> .....                             | 7        |
| 3.2.1. <i>Decom Function Input Arguments</i> .....                          | 7        |
| <b>4. Limitations</b> .....   | <b>8</b> |

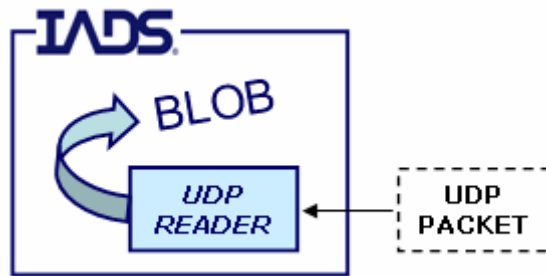
## Table of Figures

|   |   |
|---|---|
| Figure 1-1 The Blob and Decom Parameters in IADS .....                  | 3 |
| Figure 2-1 The IadsUDP.dll Installed Functions .....                    | 4 |
| Figure 3-1 Derived Parameter Setup in the IADS Configuration Tool ..... | 5 |
| Figure 3-2 IadsUDP.Read Function Input Arguments .....                  | 6 |
| Figure 3-3 The Blob Viewer (Bus Message) Display.....                   | 6 |
| Figure 3-4 Decom Function Input Arguments.....                          | 7 |

# 1. Introduction

This document describes the IADS UDP custom derived function, developed as a general purpose UDP reader.

## 1.1. Overview



The IADS UDP custom derived function is provided as a Dynamic Link Library (DLL). The IadsUDP.Read function uses the UDP protocol to read the contents of a UDP packet of any size and content; this information is then passed to the IADS Client as a blob parameter. Along with the Decom function in IADS, these arguments are used in derived parameters that you create in the IADS Configuration Tool to retrieve the individual parameters for display usage.

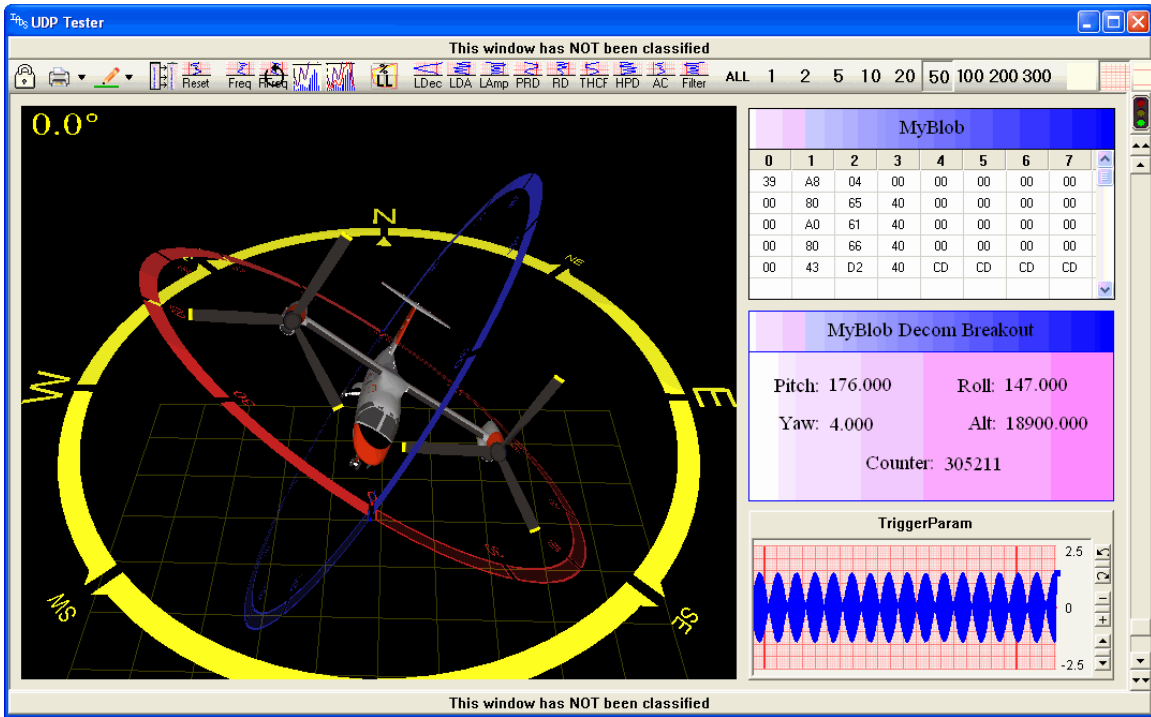


Figure 1-1 The Blob and Decom Parameters in IADS

## 2. Installation

Registration of the IadsUDP.dll needs must be executed on the machine that will receive the UDP packets; typically, this will be the IADS Server or the RT Station machine.

### To register the IadsUDP.dll:

- 1) Copy the IadsUDP.dll file to a known directory. If you're unsure where to place the it, you might consider the C:\Program Files\IADS directory.
- 2) In Windows Explorer, navigate to the directory where you copied the file.
- 3) Right-click on the IadsUDP.dll file and choose **Open With...**
- 4) Click the **Browse** button.
- 5) Browse to the **C:\Windows\System32** directory.
- 6) Select the **regsvr32.exe** file and then click the **Open** button.

*Note: If you select **Always use the selected program to open this kind of file** checkbox in the Open With dialog, from this point forward just double click on any other custom COM dll and it will register.*

- 7) Click **OK**. The function is now registered and available for use within IADS.

| <i>ProgId</i>       | <i>Description</i>  |
|---------------------|---|
| <i>IadsUDP.Read</i> | Identifies the derived function that is used to create the blob parameter in IADS |

Figure 2-1 The IadsUDP.dll Installed Functions

### 3. Instructions for Use

This section will detail how to setup and verify the blob and Decom parameters in IADS. After the IadsUDP.dll is registered, the next steps are:

- 1) Create the blob parameter
- 2) Verify the blob parameter in the Blob Viewer display
- 3) Create the Decom parameters

Figure 3-1 below, shows the blob parameter “MyBlob” created using the IadsUDP.read function, and the parameters that were pulled from “MyBlob” (Packet Count, Pitch, Roll, Yaw, Alt) using the Decom function; in the Parameter Defaults Table of the IADS Configuration Tool.

|    | Parameter         | ParamType | ShortName    | DataSourceType | DataSourceArgument   | UpdateRate |
|----|-------------------|-----------|--------------|----------------|--|------------|
| 1  | Target1_lat       | float     |              | Tpp            |  | 10.000000  |
| 2  | Target1_lon       | float     |              | Tpp            |  | 10.000000  |
| 3  | MyBlob            | blob      |              | Iap            | FiveHundredFiveHz, SetDataOutputSize(40), IadsUDP.Read(40, 1001) | 1024.0     |
| 4  | PacketCount       | int       |              | Derived        | Decom(MyBlob, 0, 4, 0, 31, 0, 1, 0)                              | 10.0       |
| 5  | Pitch             | double    |              | Derived        | Decom(MyBlob, 4, 8, 0, 63, 1, 1, 0)                              | 20.0       |
| 6  | Roll              | double    |              | Derived        | Decom(MyBlob, 12, 8, 0, 63, 1, 1, 0)                             | 20.0       |
| 7  | Yaw               | double    |              | Derived        | Decom(MyBlob, 20, 8, 0, 63, 1, 1, 0)                             | 20.0       |
| 8  | Alt               | double    |              | Derived        | Decom(MyBlob, 28, 8, 0, 63, 1, 1, 0)                             | 20.0       |
| 9  | FiveHundredFiveHz | float     | TriggerParam | Tpp            |  | 10.000000  |
| 10 | Time              | double    |              | Tpp            |  | 100.000000 |

Figure 3-1 Derived Parameter Setup in the IADS Configuration Tool

#### 3.1. Creating the Blob Parameter

The blob parameter is created using the IadsUDP.Read function. It is recommended that you process this parameter as an “IAP” type in real time. This will alleviate any random access issues and has other benefits such as producing an output data file.

##### To create the blob parameter:

- 1) In IADS, on the Dashboard click the **Configuration** button.
- 2) Open the **Data** folder, then 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, for example, **MyBlob**.
- 5) In the *ParamType* column, click the drop down arrow to select **blob**.
- 6) In the *Data Source Type* column, select **IAP**.
- 7) In the *Data Source Argument* column, enter:  
***TppParameter, SetDataOutputSize(packet size), IadsUDP.Read(arg1, arg2)***  
 For example: FiveHundredFiveHz, SetDataOutputSize(40), IadsUDP.Read(40, 1001)
- 8) Click a save option.

### 3.1.1. IadsUDP.Read Function Input Arguments

This table details the input arguments which are required in order run the function properly. If not entered correctly, IADS will flag a syntax error.

Required entries in the data source argument field before the IadsUDP.read function are:

*TppParameter* - The Tpp parameter that is used as the trigger parameter for the blob

*SetDataOutputSize(value)* - Sets the size of the blob


| <i>Input Argument</i> | <i>Description</i>                            |
|-----------------------|---|
| <i>Arg1</i>           | Packet size in bytes                          |
| <i>Arg2</i>           | PortId of the machine the data is received on |

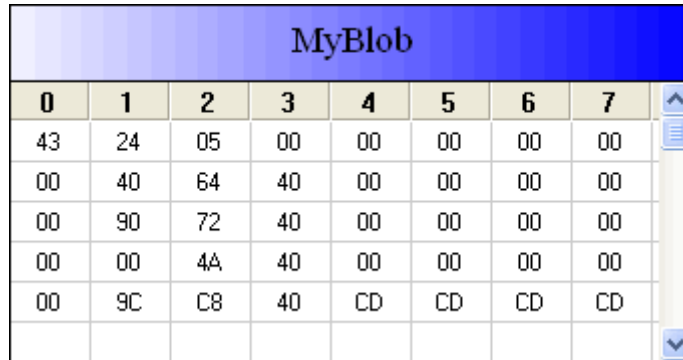
Figure 3-2 IadsUDP.Read Function Input Arguments

### 3.1.2. Verifying the Blob Parameter in the Blob Viewer Display

The Blob Viewer displays an IADS blob parameter in a byte-wise manner. The start and stop bytes are referenced from left to right and top to bottom.

**To view a blob parameter in the Blob Viewer display:**

- 1) On the IADS Dashboard, click the **Display Builder** button.
- 2) Click the **ActiveX Controls** tab.
- 3) If the Blob Viewer does not exist add the *IadsBusMessageDisplays.BlobViewer* control to the ActiveX Controls tab.
- 4) Drag and drop the  control onto the Analysis Window.
- 5) On the IADS Dashboard, click the **Parameter Tool** button.
- 6) Drag the “MyBlob” parameter onto the Blob Viewer display and click **Blob\_Input**.



| MyBlob |    |    |    |    |    |    |    |
|--------|----|----|----|----|----|----|----|
| 0      | 1  | 2  | 3  | 4  | 5  | 6  | 7  |
| 43     | 24 | 05 | 00 | 00 | 00 | 00 | 00 |
| 00     | 40 | 64 | 40 | 00 | 00 | 00 | 00 |
| 00     | 90 | 72 | 40 | 00 | 00 | 00 | 00 |
| 00     | 00 | 4A | 40 | 00 | 00 | 00 | 00 |
| 00     | 9C | C8 | 40 | CD | CD | CD | CD |

Figure 3-3 Blob Viewer (Bus Message) Display

### 3.2. Creating the Decom Parameters

Individual parameters are created from the blob parameter as derived parameters using the Decom function.

#### To create a Decom parameter:

- 1) In IADS, on the Dashboard click the **Configuration** button.
- 2) Open the **Data** folder, then 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, for example, **PacketCount**.
- 5) In the *Data Source Type* column, select **Derived**.
- 6) In the *Data Source Argument* column, enter:  
**Decom(*arg1, arg2, arg3, arg4, arg5, arg6, arg7, Arg8*)**  
For example: Decom(MyBlob, 0, 4, 0, 31, 0, 1, 0)
- 7) Click a save option.
- 8) Repeat as necessary for additional parameters.

#### 3.2.1. Decom Function Input Arguments

This table details the input arguments for the Decom function in IADS which are required in order to properly run this function; if not entered correctly, IADS will flag a syntax error. This is a standard IADS function and not specific to the IadsUDP.read function, but detailed here for proper usage.

| <i>Input Argument</i> | <i>Description</i>  |
|-----------------------|---|
| <i>Arg1</i>           | The input data parameter  |
| <i>Arg2</i>           | Byte offset   |
| <i>Arg3</i>           | Number of bytes   |
| <i>Arg4</i>           | Start bit   |
| <i>Arg5</i>           | Stop bit  |
| <i>Arg6</i>           | Data Type to return: Int = 0, IEEEFloat = 1, 1750Float = 2, CharString = 3, Array = 4 |
| <i>Arg7</i>           | Signed: False = 0, True = 1   |
| <i>Arg8</i>           | Reverse bytes: False = 0, True = 1  |

Figure 3-4 Decom Function Input Arguments

#### **4. Limitations**

- The IadsUDP.Read function is not guaranteed to receive every packet. There is no notification to the user that a packet was not received.
- Large UDP packets could get fragmented on the network.
- After editing your configuration file and saving, it may be necessary to log off and log back into IADS to display the data for the Decom parameters.