

## Deos ARINC-664/AFDX Driver Library

Deos includes an industry standard lightweight TCP/IP stack (LwIP) with a DAL-A sockets library so it can provide data transport during in flight or on ground as part of its standard package. While it may have high data integrity (e.g., through CRC or other such mechanisms), TCP/IP over Ethernet is a non-deterministic protocol. As such, it is not suitable for some avionics applications. In contrast, ARINC-664/AFDX is a redundant and deterministic data network designed for aircraft data network applications and based on switched Ethernet technologies. These features enable ARINC-664/AFDX to be used in aircraft control and other applications where both integrity and availability are essential. Since ARINC-664/AFDX is often employed as a data bus backbone for an aircraft, it's also a common interface for modern aircraft monitoring and recording systems. AFDX is an ARINC 664 data network, patented by Airbus, that is a specific implementation of ARINC 664 Part 7.

### **DO-178 DAL-A Reconfigurable, Reusable, and Adaptable Library**

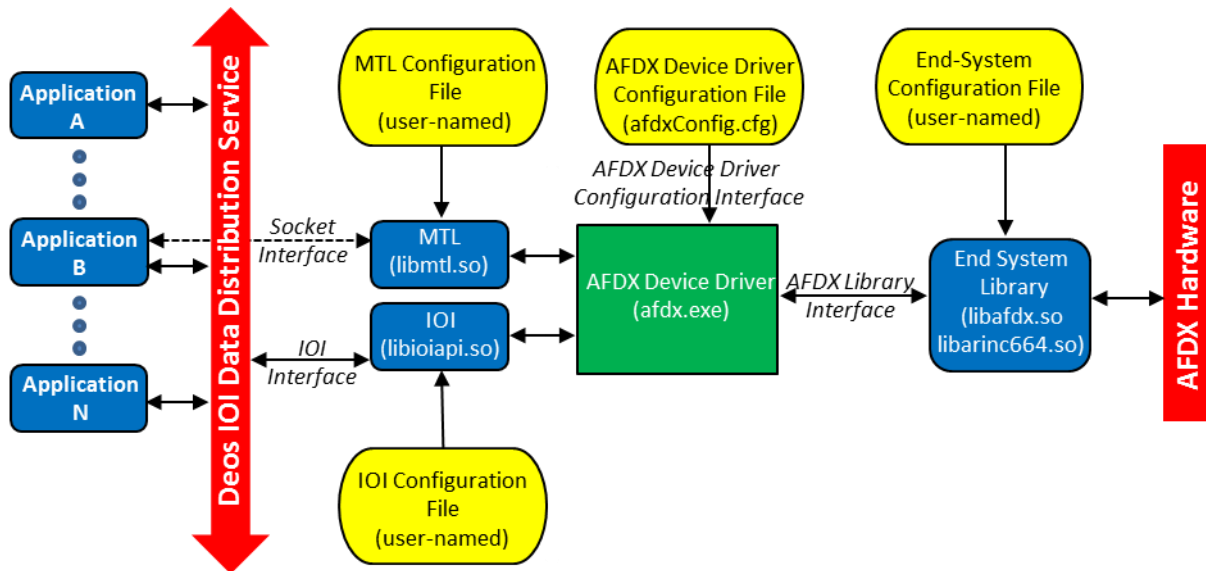
The Deos ARINC-664/AFDX library was built from the ground up as a DO-178 high Design Assurance Level (DAL A) module. Designed with the goal of reuse of certification artifacts, the ARINC-664/AFDX is a user space application that's adaptable through an XML configuration file to isolate change impact on the application during reuse<sup>1</sup>. Additionally, DDC-I provides a DO-330 qualified XML configuration checking tool.

### **Data Decoupling and Reconfiguration**

The ARINC-664/AFDX driver library allows reconfiguration and optionally leverages the Deos I/O Infrastructure (IOI) data distribution service for data decoupling. IOI provides capabilities for the distribution of data throughout an avionics system. It implements in inter-partition communications data interface between avionics applications including ARINC-653 partitions using the ARINC-653 APEX API sampling/queueing ports. Together these features allow developers to readily adapt to changes in communication structures – without impacting the verification evidence of the module. For example, a data type, or location of data within an ARINC664-AFDX packet, and other factors can change without causing the Driver Library or end application(s) to be modified (and thereby also allowing no change to the verification evidence). Instead, changes of this type may only require XML changes, and perhaps the addition of a string of conversion code. These Deos facilities minimize change impact for reuse/ reverification.

### Key Features Overview:

- Deos Libraries to DAL-A
- Deos or Deos SafeMC (multicore) support on ARM, PowerPC, & x86 based systems
- Provides Application Data Decoupling and Reconfiguration
  - Optional connection<sup>2</sup> to Deos IOI data distribution service
  - ARINC-664/AFDX packet data may be changed without recompiling end applications or the ARINC-664/AFDX driver library
  - XML configurable, with qualifiable tooling
- Portable with binary & certification reuse:
  - User-space application
  - Reusable binary library with XML-based configurability
- ARINC-664/AFDX device options
  - Rockwell Collins V5 and V6 ASICs
- Supports mixed DAL applications



The diagram above shows the multiple components provided by DDC-I to enable AFDX/664 operations.

## Key Components Include:

### End System Library

- This software interfaces to the End-System hardware.
  - Currently both Rockwell Collins Versions 5 (AFDX) and 6 (ARINC 664) ASICs are supported
- Using the End-System Configuration file, the End System Library provides API functions to initialize, configure, and operate the hardware.
- The End System Library supports Com Ports (sampling and queuing), SAP Ports, ICMP and SNMP.

### AFDX/664 Device Driver

- Manages transactions data between IOI or MTL interfaces and the AFDX/664 Library.
- In the outgoing case, the AFDX/664 Driver consumes data from IOI or Mailbox Transport Library (MTL), creates AFDX/664 data packet(s) for transmission, then calls the appropriate AFDX/664 Library routines to send data to the AFDX/664 bus.
- In the incoming case, the AFDX/664 Driver calls the appropriate AFDX/664 Library routines to receive data from the AFDX/664 bus, breaks down the AFDX/664 data packet(s) to the user-specified resolution, then produces the data to the IOI or MTL for use by User Processes.
- The AFDX/664 Driver Configuration File describes ports, rates, and data used in a given integration:
  - For example, port 500 is defined as a packet with 250 bytes with a periodic rate of 50ms; and assume that User Process 1 only needs one DWORD from the entire packet altitude, which is at byte offset 32 in the packet. The AFDX/664 Driver Configuration File can specify to the AFDX/664 Driver to read port 500 every 50ms, get the DWORD at byte 32 (altitude) and produce it to IOI. Once produced to the IOI shared memory buffering, the data (altitude) can be made available to applications that are configured to subscribe to it.
  - The AFDX/664 Driver Configuration File provides the reverse capability also. It can tell the AFDX/664 Driver to collect up various items from IOI, place them in a single AFDX/664 data packet, then transmit the packet on the configured port.
  - The resolution of the data packets is defined in the configuration file. There is nothing to preclude reading the entire data packet from a port and producing it to IOI or MTL. User Process would then consume the packet from IOI or MTL.