Neil A. Armstrong Test Facility
Common DAC System
System Architecture Specification

revision 37113 (*2023-11-02*)

Comments and Suggestions - Click HERE

1. Introduction and Scope

About the Neil A. Armstrong Test Facility

The NASA Glenn Research Center's Neil A. Armstrong Test Facility (GRC-ATF) provides unique world-class test facilities which are available for use by NASA, other Governmental agencies, and the private sector. GRC-ATF provides full service facility and test system preparation for complex and innovative research and flight programs. GRC-ATF's facilities can simulate the conditions of the upper atmosphere, deep space, planetary, lunar environments, and cryogenics. GRC-ATF staff assures safe, effective, responsive and reliable performance of research, development, and qualification testing at GRC-ATF to accomplish the mission of NASA, other Governmental agencies, and the private sector.

2. GRC-ATF Common DAC System Architecture Overview

The GRC-ATF DAC system architecture is designed to ensure that a true "n-tier" (also called "multi-tier") software architecture is achieved. "n-tier" software architectures are software architectures which enable the data processing, data management, and data presentation functions of the system to be physically and logically separated and thus distributed as needed in the system. This means that the functions of the system can be distributed across several machines or clusters of machines, ensuring that each system function can be provided without the hardware resources being shared and, as such, enables these functions to achieve maximal functional capacity and scalability as a collection of interoperable and independently manageable modules. The “N” in the name "n-tier" architecture refers to any positive integer. More through descriptions n-tier software architectures and their benefits and values can be found in all modern software system design texts.

DAC systems are more than just Information Technology (IT) systems. DAC systems are IT systems that must additionally be capable of integrating a wide-variety of Industrial Control System (ICS) Operational Technologies (OT) and scientific and industrial Measurement and Test Equipment (MTE). To ensure that GRC-ATF DAC systems can integrate all the ICS OT and MTE into an N-Tier vendor independent DAC software application, GRC-ATF has defined an *combined* hardware and software system architecture. The DAC system architecture that NASA GRC-ATF is defining is shown in Figure 1. This diagram is designed to serve as a common framework for which all specific facility DAC systems can be designed to conform to.

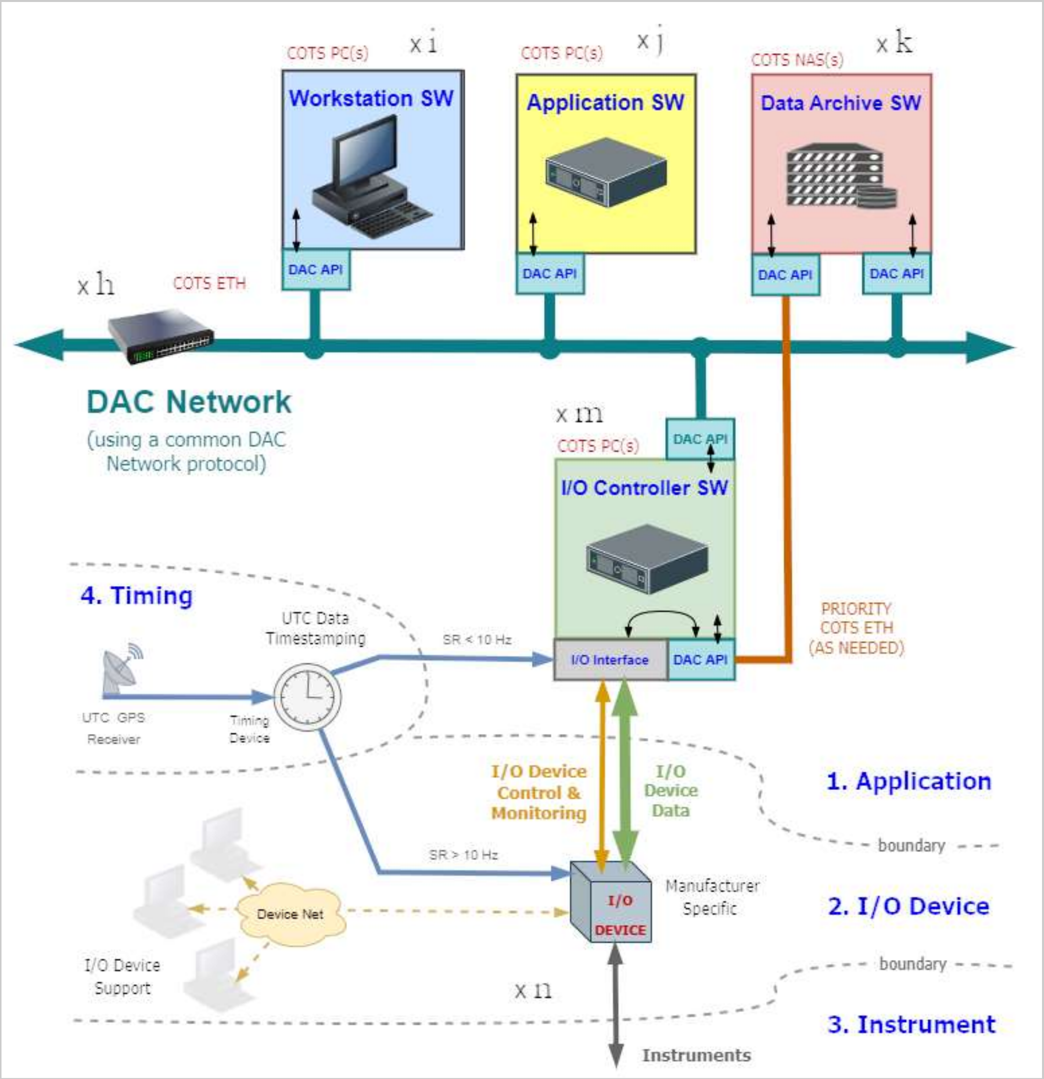


Figure 2.1 - ATF DAC System Architecture

The *hardware* architecture of a GRC-ATF DAC System comprised of four layers:

1. The Application Layer
2. The I/O Device Layer
3. The Instrument Layer
4. The Timing Layer

A general description of each DAC system hardware layer is provided in Table 2 below along with other key terms used in Figure 1. In addition to the general description, each DAC system hardware layer has a set of minimum hardware and software requirements is provided in the subsequent chapters that must be met by each hardware component that is considered for use in a given hardware layer. In cases where a single component provides functionality that spans multiple hardware layers, that component must comply with all of the requirements for each layer that it functions in.

The acronyms used in Figure 1 are defined in Table 2, descriptions of the architecture elements are provided in Table 3 and the use of multiplier indices for general scalability is clarified in Table 4 below.

Table 2.1 - Acronyms used in Figure 1

API	Application Programming Interface
COTS	Commercial Off-The-Shelf
DAC	Data Acquisition and Control
ETH	Ethernet
GPS	Global Positioning System
I/O	Input/Output
NAS	Network Addressable Storage
PC	Personal Computer
SR	Sample Rate
SW	Software
UTC	Coordinated Universal Time

Table 2.2 - Terms used in Figure 1

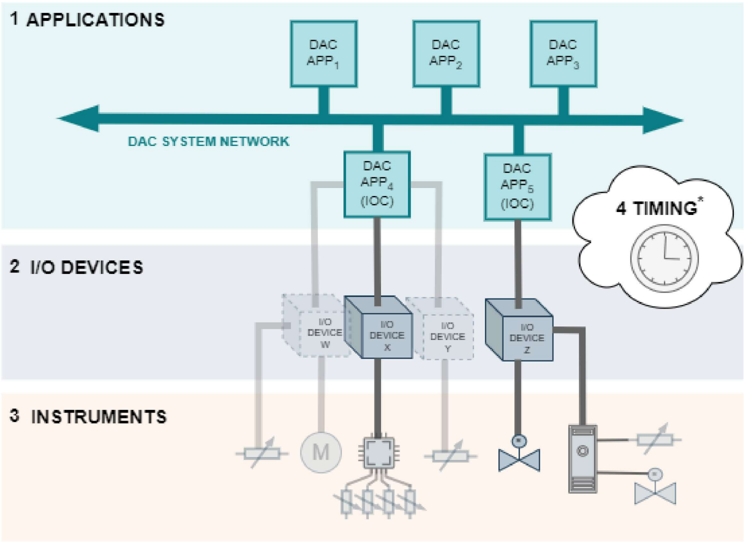
Application SW	The term "Application SW" refers to any software in the application layer that provides the overall application functionality required by the DAC systems functional requirements. *Notes: 1) N-Tier software architecture refers to this as the "data processing" tier. 2) Application SW only communicates with other devices on the Application Device Layer and only does so using the common DAC API and the common DAC protocol (i.e. pvAccess, or OPCUA).*
COTS PC	The term "COTS PC" is used generically to refer to any readily available industry standard computer platform in any readily available industry standard form factor as can be found in all major IT catalogs.

COTS NAS	The term "COTS NAS" is used generically to refer to any readily available industry standard NAS device in any readily available industry standard form factor as can be found in all major IT catalogs.
COTS ETH	same as COTS TCP/IP
COTS TCP/IP	The term "COTS TCP/IP" is used generically to refer to any readily available industry standard TCP/IP networking devices in any readily available industry standard form factor as can be found in all major IT catalogs.
DAC API	The term "DAC API" refers to a common API used
DAC Network	The DAC Network is a standard Ethernet-based communication network that allows the devices in the Application Layer devices to communicate with each other. The software elements installed on the Application Layer devices are required to communicate with each other using a common DAC Network Protocol through a common DAC Network (i.e. pvAccess, or OPCUA)
Data Archive SW	Data Archive Software is software installed on a COTS PC in the application layer that provides the overall data storage functionality required by the DAC systems functional requirements. Note: 1) N-Tier software architecture refers to this as the "data storage" tier. 2) Data Archive SW only communicates with other devices on the Application Device Layer and only does so using the common DAC API and the common DAC protocol (i.e. pvAccess, or OPCUA).
Instruments	The term "Instruments" refers to any/all passive or active sensors and/or indicators in the field, smart or otherwise. Instruments can be manufacturer specific and in most cases rely on an I/O device to condition, excite, power and/or make use of in the most general sense. Examples of instruments are: 1) Thermocouples, 2) accelerometers, 3) microphones, 4) pressure sensors, 5) electric field probes, 6) indicator lamps, 7) beacons, 8) motors, 9) pumps, 10) actuators, 11) limit switches, 12) antennas, 13) ADC/digitizers, 14) Remote I/O devices, etc..
I/O Controller SW	The term "I/O Controller SW" refers to any software installed on a COTS PC in the application layer that provides an device specific software interface between al other the Application SW in the application layer and a specific I/O device in the I/O Device Layer. Note: 1) As software running on a device in Application Device Layer, the I/O Controller SW only communicates with the other devices in the application layer using the common DAC API and the common DAC protocol (i.e. pvAccess, or OPCUA). 2) A specific I/O Controller SW only communicates with a specific set of I/O Devices in the I/O device layer through an "I/O Interface".
I/O Device Control & Monitoring	The term "I/O Device Control & Monitoring" refers to any and all transmitted and received communication between an "I/O Interface" in the application layer and a specific specific I/O device in the I/O device Layer for the purpose of monitoring or controlling the state and status of the I/O Device.
I/O Device Data	The term "I/O Device Data" refers to any and all measurement data transmitted to an "I/O Interface" in the application layer from a specific specific I/O device in the I/O device Layer.
I/O Device	The term "I/O Device" refers to any/all devices that serves to A) provide whatever electronics are needed to facilitate the purpose of the instrument (power, excite, condition, etc..) as well as B) facilitate all communicate with the instrument and make all control, monitoring and data functions accessible to an I/O Controller in the Application Layer. Examples of I/O Devices are: 1) PLCs, 2) signal conditioners, 3) Oscilloscopes, 4) DMMs, 5) Function Generators, 6) electric field probe Interfaces, etc...
I/O Device Support	The term "I/O Device Support" refers generally to any aspect of an I/O Device's maintenance, configuration or diagnostics that must be performed by connecting directly to the Device itself. In other words; functions that are not provided by the application layer. Examples include: reading I/O Device error logs or flashing the firmware. In some cases configuring the device until the device configuration functions can be performed by the I/O Controller in the Application Layer. Examples of this type of I/O Device Support include: Configuration of the Precision Filters Inc. (PFI) Signal Conditioners using the PFI Graphical User Interface (GUI) with Factory Acceptance Test (FAT) capabilties.
I/O Device Net	The term "I/O Device Net" refers to any network devices that are required for I/O Device Support.
I/O Interface	The term "I/O Interface" refers to the low-level software drivers that is integrated with and used by the I/O Controller SW to control and monitor the essential features, data and performance of a specific I/O device in the I/O device Layer.
Timing Device	The term "Timing" Device" refers to any device in the Timing Layer that receives UTC time data from a GPS system and uses that information to provide the necessary signal (or signals) required for I/O Devices and I/O Controllers to set internal clocks and timestamp data when required.
UTC Data Timestamping	The term "UTC Data Timestamping" refers generally to the primary function of the Timing Layer to obtain the UTC time from a GPS-based receiver and provide whatever signal or signal(s) that are required by the I/O Devices and I/O Controllers to set their clocks and timestamp data as and when required.
UTC GPS Reciever	The term "UTC GPS Receiver" refers to any device in the Timing Layer that continuously receives the UTC time from GPS signals and continuously broadcasts the current UTC time to any Timing Devices in the Timing Layer.
Workstation SW	Workstation Software is software installed on a COTS PC in the application layer that provides the operator screens required by the DAC systems HMI (Human/Machine Interface) requirements. Note: 1) N-Tier software architecture refers to this as the "data presentation" tier. 2) Workstation SW only communicates with other devices on the Application Device Layer and only does so using the common DAC API and the common DAC protocol (i.e. pvAccess, or OPCUA).

Table 2.3 - Scalability indices used in Figure 1

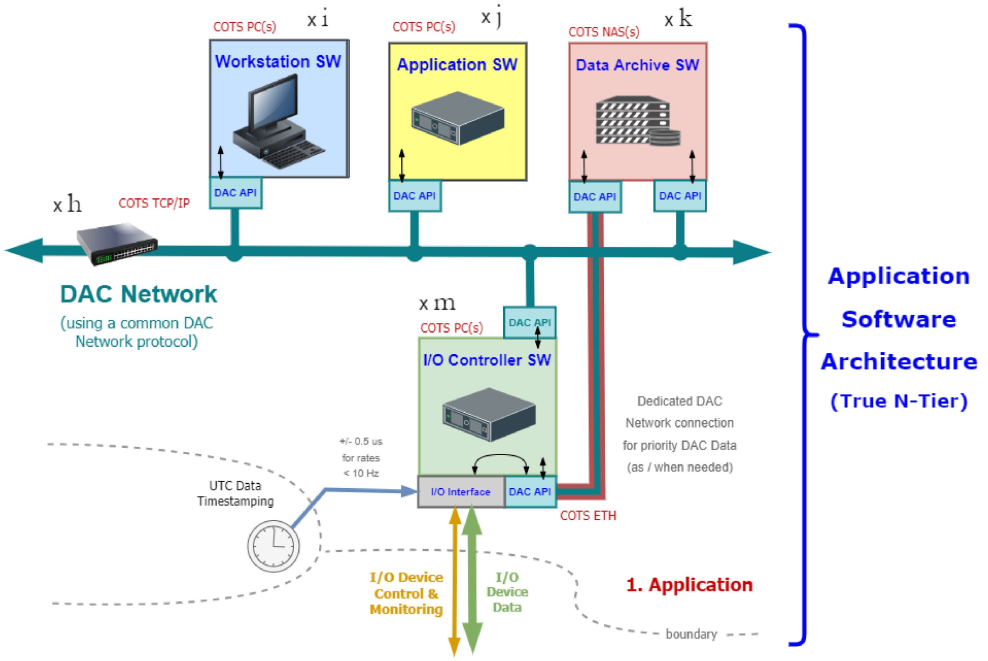
index	referent	description
i	Number of Workstation PCs	The index "i" implies the scalable number of workstations that a specific DAC system has. The term "xi" should be read, "The DAC system has "i" number of COTS PCs being used as operator workstations".
j	Number of Application PCs	The index "j" implies the scalable number of Application PCs that a specific DAC system has. The term "xj" should be read, "The DAC system has "j" number of COTS PCs being used as application servers".
k	Number of Data Archive PCs	The index "k" implies the scalable number of Data Archive PCs that a specific DAC system has. The term "xk" should be read, "The DAC system has "k" number of COTS PCs being used as Data Archive servers".
h	Number of Network Devices	The index "h" implies the scalable number of Network devices that a specific DAC system has. The term "xh" should be read, "The DAC system has "h" number of COTS Ethernet devices being used to provide the DAC Network".
m	Number of I/O Controller PCs	The index "m" implies the scalable number of I/O Controller PCs that a specific DAC system has. The term "xm" should be read, "The DAC system has "m" number of COTS PCs being used as I/O Controllers".
n	Number of I/O Devices	The index "n" implies the scalable number of I/O Devices PCs that a specific DAC system has. The term "xn" should be read, "The DAC system has "n" number of I/O Devices".

3. DAC System Architecture Requirements

REQID	Requirement Title	Requirement Text
3.1	Hardware Architecture	The GRC-ATF DAC System hardware architecture shall conform to the DAC System Hardware Architecture diagram defined in **Figure 3.1** such that each physical component (device) of the system can be shown to conform to the hardware and software requirements of the device layer it is located in.  1 APPLICATIONS DAC APP$_1$ DAC APP$_2$ DAC APP$_3$ DAC SYSTEM NETWORK DAC APP$_4$ (IOC) DAC APP$_5$ (IOC) 4 TIMING* 2 I/O DEVICES I/O DEVICE W I/O DEVICE X I/O DEVICE Y I/O DEVICE Z 3 INSTRUMENTS M **Figure 1 - ATF DAC Hardware Architecture** *Rationale - The architecture provides a framework that allows each layer to define hardware and software requirements that are specific to each layer. Managing the requirements of each layer is essential to being able to achieve vendor independent interoperability of the system components across all DAC systems at NASA GRC-ATF.*
3.1.1	Application Device Layer Requirements	All components of the GRC-ATF DAC System that are located in the "Application Device Layer" shall: 1. Be identified in the ATF DAC System Design as an "Application Device" 2. Conform to all requirements within **section 4** of this document which defines the minimum Application Device Layer hardware and software requirements for components to qualify for use in that layer of the overall system architecture. *Rationale - NASA GRC-ATF seeks to define and manage the functionality and supportability of the Application Layer independently of the rest of the system.*

3.1.2	I/O Device Layer Requirements	All components of the GRC-ATF DAC System that are located in the "I/O Device Layer" shall: 1. Be identified in the ATF DAC System Design as an "I/O Device" 2. Conform to all requirements within **section 5** of this document which defines the minimum I/O Device Layer hardware and software requirements for components to qualify for use in that layer of the overall system architecture. *Rationale - NASA GRC-ATF seeks to define and manage the functionality and supportability of the I/O Layer independently of the rest of the system.*
3.1.3	Instrument Device Layer Requirements	All components of the GRC-ATF DAC System that are located in the "Instrument Device Layer" shall: 1. Be identified in the ATF DAC System Design as an "Instrument Device" 2. Conform to all requirements within **section 6** of this document which defines the minimum Instrument Device Layer hardware and software requirements for components to qualify for use in that layer of the overall system architecture. *Rationale - NASA GRC-ATF seeks to define and manage the functionality and supportability of the Instrument Layer independently of the rest of the system.*
3.1.4	Timing Device Layer Requirements	All components of the GRC-ATF DAC System that are located in the "Timing Device Layer" shall: 1. Be identified in the ATF DAC System Design as an "Timing Device" 2. Conform to all requirements within **section 7** of this document which defines the minimum Timing Device Layer hardware and software requirements for components to qualify for use in that layer of the overall system architecture. *Rationale - NASA GRC-ATF seeks to define and manage the functionality and supportability of the Timing Layer independently of the rest of the system.*

4. Application Device Layer Requirements

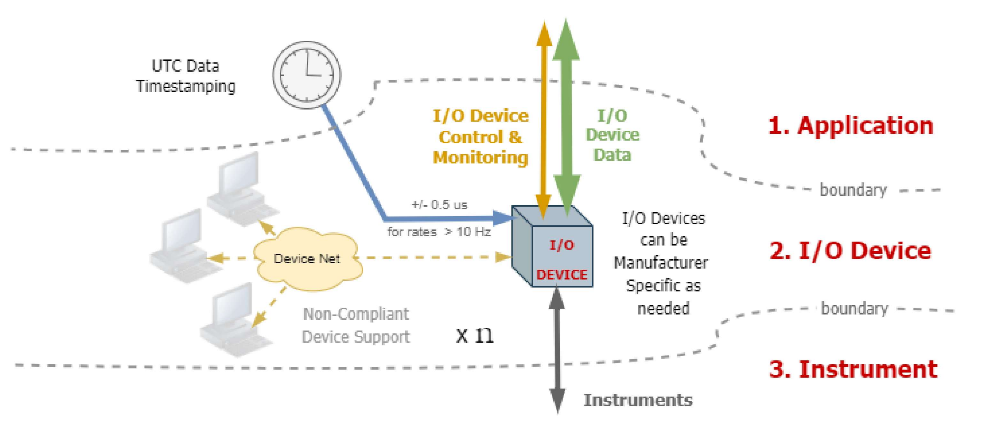
REQID	Requirement Title	Requirement Text
4.1	Application Layer Architecture	The **GRC-ATF DAC System Application Layer** shall conform to the DAC System Application Layer Architecture diagram defined in **Figure 2** such that each physical component (device) of the Application Layer can be shown to conform to the hardware and software requirements defined in this section.  COTS PC(s) x i Workstation SW COTS PC(s) x j Application SW COTS NAS(s) x k Data Archive SW DAC API DAC API DAC API DAC API x h COTS TCP/IP DAC Network (using a common DAC Network protocol) x m COTS PC(s) DAC API I/O Controller SW Dedicated DAC Network connection for priority DAC Data (as / when needed) +/- 0.5 us for rates < 10 Hz UTC Data Timestamping I/O Interface DAC API COTS ETH I/O Device Control & Monitoring I/O Device Data 1. Application boundary Application Software Architecture (True N-Tier) **Figure 2 - ATF DAC Application Layer Architecture** *Rationale -*

REQID	Requirement Title	Requirement Text
4.1.1	**Application Layer Hardware Requirements**	
4.1.1.1	Open-Standards-based COTS Hardware	All Application Device Layer devices shall be based on vendor independent open-standards-based COTS hardware that meets NASA's IT hardware requirements *Rationale - This is required to ensure vendor independence between the hardware and the software in the application layer. In other words, all application layer devices shall be based on commonly available hardware found in all major IT catalogs (i.e. cabinets, racks, network switches, network cabling, power supplies, computer cases, motherboards, memory, disks, CPUs, network cards, graphics cards, monitors, keyboards, pointing devices, etc..) Avoid all hardware that is vendor specific.*
4.1.1.2	Open-Standards	Be based entirely on Open-Standards when possible
4.1.2.8	Availability	Be 100% Vendor Independent

REQID	Requirement Title	Requirement Text
4.1.2	**Application Layer Software Requirements**	
4.1.2.1	N-Tier	(SW) - Be based on a true N-tier Client-Server architecture
4.1.2.2	Open-Standards	Be based entirely on Open-Standards when possible
4.1.2.3	Modular	Be Modular
4.1.2.4	Distributable	Be highly distributable across multiple PCs as needed
4.1.2.5	Collapsible	Be able to be consolidated onto a single PC as appropriate
4.1.2.6	Interoperability 1	Be interoperable with multiple CPUs and Operating Systems
4.1.2.7	Interoperability 2	Be Interoperable with a variety of manufacturer specific I/O device hardware via a common internationally used open-standards-based TCP/IP API Protocol
4.1.2.8	Availability	Be 100% Vendor Independent
4.1.2.9	Maintainability	Use only GPL (or better, e.g. BSD) licensed fully open-source (or open-sourcable) software (including OSs)
4.1.2.10	Life Cycle	Be in active use by a variety of highly reputable international end-users.
4.1.2.11	Sustainability	Have a large, active, open, international developer community
4.1.2.12	Supportability	Have multiple options for for 3rd party support from an international list of experienced vendors and developers.
4.1.2.13	Interoperability	Utilize a common DAC API protocol that: • Supports scalar and N-dimensional data • Transports data with all metadata which includes (at a minimum) EUs, timestamp, and alarm status • Enables correlation of data across the entire application within ± 10 ns wrt UTC • Is an internationally used open-standard
4.1.2.14	Data Rates	Can support data recording rates of arbitrarily low speeds to up to 2 MSPS with ± 0.5 µs accuracy or better
4.1.2.15	Data Channel SW Capability	Be capable of supporting up to 1 million channels and process variables in a single DAC system (including both physical and calculated channels)
4.1.2.16	System Merging	Be capable of joining / combining multiple DAC systems seamlessly up to 1 million total channels / process variables in the composite system
4.1.2.17	System Partitioning	Be capable of arbitrarily partitioning segments of the DAC system to enable segment specific rules for viewing and controlling channels across the composite DAC system(s).
4.1.2.18	Recording Time	Be capable of continuous, gapless, data recording for at least 100 days (data storage limits not withstanding)
4.1.2.19	Storage Capacity	Have a total data storage capacity that is seamlessly scalable up to at least 100 TB and is system configurable to RAID 1, 2, 5, and 1+0.

4.1.2.20	Commonality	Utilize a common I/O Controller SW that: • Enables the integration of Manufacturer specific I/O device command and monitoring functionality • Enables the integration of Manufacturer specific I/O device data collection and data processing • Has a large, existing, open-source collection of I/O devices supported to pull from and contribute to
4.1.2.21	Code Re-Use	Has a large open collection of existing common applications and services to select from.
4.1.2.22	I/O Device Support	Has a large and internationally supported open collection of manufacturer specific devices that are supported.

5. I/O Device Layer Requirements

REQID	Requirement Title	Requirement Text
5.1	I/O Device Layer Architecture	The **GRC-ATF DAC System I/O Device Layer** shall conform to the DAC System I/O Device Layer Architecture diagram defined in **Figure 3.1** such that each physical component (device) of the I/O Device Layer can be shown to conform to the hardware and software requirements defined in this section. **Figure 3.1 - The GRC-ATF DAC System I/O Device Layer**  UTC Data Timestamping I/O Device Control & Monitoring I/O Device Data 1. Application boundary +/- 0.5 us for rates > 10 Hz I/O DEVICE I/O Devices can be Manufacturer Specific as needed Device Net 2. I/O Device Non-Compliant Device Support x n boundary 3. Instrument Instruments Rationale - The architecture needs a dedicated layer for managing the requirement on devices that provide an interface between the instruments and the applications.
REQID	**Requirement Title**	**Requirement Text**
5.1.1	Analog Front-End Performance	Meet or exceed the technical and performance requirements of the specific ATF DAC system's data collection goals (i.e. Sample Rate, SNR, Range, Sensitivity, Measurement / Conditioning Type, etc..) *Rationale - Not all DAC systems will require the same analog front-end performance. The analog front-end requirements for a specific DAC system need to be defined uniquely for that specific system and ADDED to these common architecture requirements.*
5.1.2	Interoperability via an Open-API	Provide all I/O control, configuration, monitoring, data collection, and data transfer functions via well-documented and clearly defined interfaces for integration into the ATF DAC system Application Layer. *Rationale – An Open Application Programming Interface (Open-API) is essential for hardware/software vendor independence. The project is not interested in any features or capabilities of commercial systems that that require the vendors application software to achieve.*
5.1.3	Supportability as Open-Source	Provide source code for all application layer interfaces required to integrate the manufacture's device into the ATF system Application Layer and allow the interface code to be freely published as open source by NASA. *Rationale – Preferably an Open Software Developers Kit (Open SDK) is essential for enabling third-party development of the software (driver) needed to integrate the device into the EPICS application software layer.*
5.1.4	Data Latency (for timeseries scalar data channels with sample rates ≤ 10 Hz)	ATF DAC System I/O Devices that produce data at data rates ≤ 10 Hz shall deliver that data to the Application Layer I/O Controller within 100 ms of time of the time of conversion or production by the device. *Rationale - Low-Speed Data Channels must deliver their data to the application layer within a minimum time*

5.1.5	On-board Timestamping (for timeseries scalar data channels with sample rates > 10 Hz)	ATF DAC System I/O Devices producing data channels at data rates > 10 Hz I/O devices shall timestamp each datum of data created by the device to ± 0.5 us (or better) from the time of creation with respect to an externally provided UTC time reference that has an accuracy of ± 0.1 us (or better). *Rationale – "On-board timestamping" is essential for eliminating the need for timestamping to be performed in the application layer. This enables truly arbitrary scaling of system channel counts and enables the application layer to process all data (time-series or otherwsie) natively as event-based scatter plot data. In other words, the application does not need to calculate the timestamp of any individual data point from the context of a recording header or some official recording start time.*
5.1.6	On-board Calculated Channels for timeseries scalar data channels	For I/O Devices producing timeseries scalar data (typically direct voltage channels) I/O device shall be capable of being programmed (typically using an on-board FPGA) to generate calculated channels in parallel and in real time with locally generated scalar data. At a minimum, the following types of calculated channels shall be possible by I/O Devices producing timeseries scalar data: • Rolling (box-car) time-average with configurable averaging parameters • FFTs and PSDs with configurable FFT and PSD parameters • FFT and PSD peak tracking with configurable FFT and PSD peak tracking parameters • FIR and IIR filters with configurable filter parameters *Rationale – An "On-board FPGA" is essential for enabling the on-board calculation of real-time FFTs, box-car averages, and many other time and frequency based filters and functions as calculated channels that can be received by the application layer in parallel with the raw data in real time.*

6. Instrument Device Layer Requirements

Once the Application and I/O Device Layer requirements have been met, there are only 3 "common" requirements defining qualifying Instrument Devices as listed in the Instrument Device Layer Requirements Table below.

REQID	Requirement Title	Requirement Text
6.1	Measurement Performance	ATF DAC Instrument Devices shall meet or exceed the technical and performance requirements of the specific ATF DAC system's data collection goals (i.e. Sample Rate, SNR, Range, Sensitivity, Measurement / Conditioning Type, etc..)
6.2	Architecture Fidelity	Instruments may only interface with qualifying I/O Devices that comply with the requirements of section 4 above. Instruments shall NOT interact directly with any aspect of the application layer. All instrument interfacing must be done through a qualifying I/O Device as defined in section 5 above. *Rationale - The Application layer needs to be agnostic with respect to specific instruments. Whatever an instrument is or does, must be addressed by a qualifying I/O device. Instruments are not allowed to "bypass" the I/O Device layer and interface to the application layer directly.*

7. Timing Device Layer Requirements

The Timing Layer:

REQID	Requirement Title	Requirement Text
7.1	Receives UTC via GPS	The Timing Device Layer shall be capable of receiving an external GPS-based UTC time code signal as the basis of its global time reference. *Rationale - DAC systems are often provided with an existing GPS-based UTC time code signal from a remote GPS receiver system. Note - The UTC accuracy is not specified in this document. Each test site shall be responsible for defining the accuracy of the UTC signal it provides its DAC systems.*
7.2	Timing Stability	The Timing Device Layer shall be capable of establishing an internal timestamping reference for the DAC system to ensure that between any two I/O devices there is an accuracy of at least 100 ns or better. Rationale - While the timing offset to the outside world can vary greatly, the I/O Devices of the DAC system components need to be synchronous with each other to a much higher degree of precision to produce internally consistent data and events.

7.2	External Signal Loss Tolerance	In the event of an outage of the UTC time code signal, the Timing Device Layer shall be capable of providing uninterrupted timecode services to all devices in all formats used. *Rationale - The availability of the external GPS-based UTC time code signal should not be a risk to the continuous operation of a DAC system.*
7.3	NTP Timing Protocol	The Timing Device Layer shall be capable of providing Network Time Protocol (NTP) time service to any devices in the architecture that require it. *Rationale - DAC systems often include CPU-based network devices that require an NTP time service.*
7.4	IRIG Timing Protocol	The Timing Device Layer shall be capable of providing Inter-range instrumentation group (IRIG) timecode signals to any devices in the architecture that require it. *Rationale - DAC systems often include devices that require IRIG timecode signals for timing accuracies of up to 10 ms (IRIG-B) or even 1 ms (IRIG-A).*
7.5	PTP Timing Protocol	The Timing Device Layer shall be capable of providing Precision Time Protocol (PTP) per IEEE 1588 timecode signals to any devices in the architecture that require it. *Rationale - DAC systems often include devices that require PTP timecode signals for timing accuracies of up to 1 µs microsecond.*
7.6	MRF Timing Protocol	The Timing Device Layer shall be capable of providing (MRF) event-based timing signals to any devices in the architecture that require it. *Note - The mRF event-based tiing system specification is available at:* http://mrf.fi/fw/DCManual-191127.pdf *Rationale - DAC systems often include devices that require MRF event-driven timing signals for timing accuracies of > 1 µs microsecond and/or including event codes as context.*
7.7	Architecture Fidelity	All Timing Devices in the timing layer shall function collectively such that they provide no additional system functionality aside from providing the timing related services. *Rationale - The Application layer needs to be agnostic with respect to specific instruments. Whatever an instrument is or does, must be addressed by a qualifying I/O device. Instruments are not allowed to "bypass" the I/O Device layer and interface to the application layer directly.*

Once these timing requirements have been achieved, the overall DAC system can then be programmed to:

1. Coordinate more complex event sequences
2. Synchronize the local time at different locations with high precision (greater than 10 ns)
3. Timestamp events at different locations to analyze what happened first

Appendix A - Glossary

	API	Application Programming Interface
	ATF	Armstrong Test Facility
	COTS	Commercial Off-The-Shelf
	DAC	Data Acquisition and Control
	ETH	Ethernet
	EPICS	Experimental Physics and Industrial Control System
	FOSS	Free and Open-Source Software
	GOTS	Government Off-The-Shelf
	GPS	Global Positioning System

	GRC	Glenn Research Center
	I/O	Input/Output
	IRIG	Inter-range instrumentation group
	MRF	Micro-Research Finland
	NAS	Network Addressable Storage
	NASA	National Aeronautics and Space Administration
	NTP	Network Time Protocol
	PC	Personal Computer
	PTP	Precision Time Protocol
	SR	Sample Rate
	SW	Software
	UTC	Coordinated Universal Time

National Aeronautics and Space Administration

Page: **Neil A. Armstrong Test Facility - Common DAC System - System Architecture Specification**

Link: https://wiki.grc.nasa.gov/pbgeneral/Neil_A._Armstrong_Test_Facility_-_Common_DAC_System_-_System_Architecture_Specification

Last Updated: Thu. Nov. 2, 2023 (11:13 PM)

Contributors: R.Evans

Retrieved from "https://wiki.grc.nasa.gov/pbgeneral/index.php?title=Neil_A._Armstrong_Test_Facility_-_Common_DAC_System_-_System_Architecture_Specification&oldid=37113"